

Article

A GtoG Direct Coding Mapping Method for Multi-Type Global Discrete Grids Based on Space Filling Curves

Yalu Li ¹, Xuesheng Zhao ^{1,*} , Wenbin Sun ¹, Guangsong Wang ², Fuli Luo ³, Zheng Wang ⁴  and Yuanzheng Duan ¹

¹ School of Geosciences and Surveying Engineering, China University of Mining and Technology-Beijing, Beijing 100083, China

² Beijing SunWise Space Technology Ltd., Beijing 100191, China

³ College of Civil Engineering, Southwest Forestry University, Kunming 650224, China

⁴ College of Civil and Architectural Engineering, North China University of Science and Technology, Tangshan 063210, China

* Correspondence: zxs@cumt.edu.cn; Tel.: +86-135-2056-6340

Abstract: DGGs (Discrete Global Grid System) has many subdivision models and coding methods. Due to the lack of underlying consistency of different DGGs codes, most of them are converted through longitude–latitude, which greatly reduces the interoperability efficiency of different DGGs data and has become one of the bottlenecks in efficient integration of multi-source DGGs data. Therefore, a direct mapping method from one grid code to another (Grid to Grid, GtoG) for multi-type DGGs is proposed based on three classical DGGs (triangular, diamond and hexagonal grids) and two commonly used filling curves (*Hilbert curve* and *Z-curve*). The mutual conversion rules of different grids expressing spatial point, line and surface data are constructed. Then, the above method is extended to the spherical icosahedral grid framework, and three different region coding mapping rule tables of the basic inside cells, boundary cells and vertex cells are designed. Finally, the experimental results show that, compared with the longitude–latitude conversion method, the average conversion efficiency of spatial point, line and surface data is increased by 2–4 orders of magnitude. This new method greatly improves the interoperability efficiency and provides a feasible solution for the efficient integration of multi-source DGGs data.

Keywords: Discrete Global Grid System; filling curve; code interoperability; data fusion



Citation: Li, Y.; Zhao, X.; Sun, W.; Wang, G.; Luo, F.; Wang, Z.; Duan, Y. A GtoG Direct Coding Mapping Method for Multi-Type Global Discrete Grids Based on Space Filling Curves. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 595. <https://doi.org/10.3390/ijgi11120595>

Academic Editors: Wolfgang Kainz and Sisi Zlatanova

Received: 21 August 2022

Accepted: 24 November 2022

Published: 27 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Discrete Global Grid System divides the earth evenly to form multi-resolution, hierarchically indexed location systems that seamlessly and non-overlappingly tile the globe. It uses cell address codes instead of traditional geographic coordinates to participate in data operations [1], and it supports the effective integration, organization and management of global multi-source and multi-scale satellite remote sensing images, terrain data, vector data, urban 3D models and other basic data [2]. In the past two decades, the importance of discrete global grids for digital earth has been known [3] and a series of research results have been achieved [4–12]. In 2015, DGGs was identified as the foundation of a modern digital earth framework [13].

According to the different subdivision types, the commonly used global discrete grid cells mainly include spherical triangle, diamond (quadrilateral) and hexagon (as shown in Table 1). Among them, as the basic cell of visualization, triangle has a high rendering efficiency and is conducive to terrain modeling. The diamond (quadrilateral) has the same shape characteristics as the flat square cell and can well inherit the rich and mature algorithms on the square cell. Therefore, the global diamond (quadrilateral) grid system is widely used in the storage and management of massive remote sensing images and terrain data [14,15]. Hexagonal cells have uniform adjacency, and their proximity query efficiency

is the highest. In addition, the hexagonal mesh has a smaller quantization error and higher angular resolution [16] and sampling density [17], and the topological distance between nodes in the hexagonal mesh matches the pattern of the Euclidean space linear distance best, which is helpful in the establishment of a grid metric space [18] and planning for analysis. In order to meet the application requirements in different fields and realize the optimal expression, processing and analysis of spatial data, scholars have designed and developed several different grid model and coding systems according to the characteristics of triangular, diamond and hexagonal cells.

Table 1. Discrete global grid subdivision and index structure.

Author	Subdivision Method	Index Structure
Dutton, G.H. [19]	Triangular grid of regular octahedral	Hierarchy code
White, D. [20]	Diamond grids of regular icosahedral and octahedral	Morton code
Bartholdi, J.J. [21]	Triangular grid of polyhedron	Space filling curve
Górski, K.M., et al. [22]	1:4 subdivided HEALPix grid of diamond dodecahedron	Hierarchy code
Sahr, K. [23]	Aperture 3 hexagonal grid of regular icosahedral	MGBT hierarchy code
Xuesheng Zhao et al. [24]	Variable longitude–latitude DQG grid	Degenerate quadtree code
Vince, A. et al. [25]	Aperture 3 hexagonal grid of regular icosahedral	PYXIS hierarchy code
Ali Mahdavi Amiri [26]	1-to-2 refinement squares grid of cube	Hierarchy code and coordinate code
Gibb, R.G. [27]	1:9 subdivided rHEALPix grid of cube	<i>Z-curve</i>
Veach, E. et al. [28]	Aperture 4 square grid of cube	<i>Hilbert curve</i>
Uber Technologies [29]	Aperture 7 hexagonal tessellation of the icosahedron	Hierarchy code
Bingxian Lin et al. [30]	Regular icosahedral triangular grid	<i>Z-curve</i>
Jin Ben et al. [31]	Regular icosahedral aperture 3 hexagonal grid	Hierarchy code
Sahr, K. [32]	Regular icosahedral mixed aperture hexagonal grid	Hierarchical linear index code
Hall, J. et al. [33]	Aperture 4 triangular grid of triacontahedron	<i>Z-curve</i>
Rui Wang et al. [34]	Regular icosahedral mixed aperture 3 and 4 hexagonal grid	Hierarchy code

However, due to the lack of a unified description of the underlying grid, different grid data are mostly converted using longitude–latitude as middleware. Longitude–latitude is a floating-point format and lacks hierarchical features, which leads to a huge computing cost and low efficiency in the integrated analysis and interoperability of multi-source data in DGGS. Therefore, the efficient conversion of different global discrete grid data has become one of the urgent problems to be solved in DGGS multi-source information integration and sharing. In 2014, the OGC established a global discrete grid standardization working group to draft DGGS standards to promote the progress of DGGS interoperability in cross fields. Since the OGC issued the DGGS standard in 2017, DGGS interoperability has been listed increasingly often as one of the important development directions of DGGS [35,36]. Several scholars and experts have discussed different discrete grid interoperability methods, which can be divided into two types.

The first type is commonly used to realize the conversion of any DGGS through the longitude–latitude middleware. For example, Amiri et al. [37] proposed a general model of DGGS index conversion, meaning that different grid systems are converted to each other through longitude–latitude middleware. Gibb [27] implemented the data transfer between the rHEALPix grid and the HEALPix grid with the PROJ.4 projection library. Lingyu Du [38] used the weak duality method to transform the discrete line from a hexagonal grid expression to a triangular or diamond grid expression. This method can only ensure that some center points are coincident. When the cell center points do not coincide, it is also necessary to judge the position relationship between the cell center point and the vector line according to the spherical geometric algorithm, which increases the complexity of transformation. Yihang Chen et al. [39] used the longitude–latitude method to realize the conversion of different codes of the QTM grid.

The second type consists of methods that realize the conversion of different DGG codes through a row–column index. The common plane code is a space filling curve, which includes

the *Hilbert* curve, *Z-curve* and *Gray* curve, etc. Different space filling curves interoperate with row–column coordinates as middleware. For example, Amiri et al. [37] studied a grid conversion method in which only the spherical grid index structures are different and the other elements are the same; that is, the code is converted into a vector in the same coordinate system, and then converted into another code. Zhipeng Zhao [40] and Kun Liu [41] both proposed a composite grid data model integrating diamonds, triangles and hexagons based on the row–column index scheme, and established the conversion relationship between diamond, triangular and hexagonal grids by using the four corners of the diamond. Yihang Chen et al. [39] used the hierarchical transformation method and the row–column coordinate method to realize the transformation of different codes of the QTM grid.

In addition to using the row–column coordinate middleware to convert space filling curves, other space filling curves can be converted through bit operations. Hongbo Xu realized the mutual conversion of the *Z-curve*, *Hilbert* curve and *Gray* curve through bit operation, and used the *Z-curve* to speed up the neighbor search of the *Hilbert* curve and *Gray* curve [42].

In summary, using longitude–latitude as the middleware can realize the conversion between any DGGS systems, but this conversion needs to judge whether the spherical point is in the spherical polygon and whether it intersects between the spherical polygons, involving a large number of complex floating-point calculations, which greatly affects the interoperability efficiency of DGGS. In addition, because the spherical grid involves polyhedral vertex and cross surface problems, the row–column mapping method and bit operation mapping method of a plane space filling curve cannot be directly used for the code conversion of the spherical grid. At present, the existing row–column coordinate mapping methods for spherical grids are limited to conceptual models only and lack operable transformation methods for different grids.

Targeting the above problems, this paper proposes a direct mapping method using multi-type global discrete grid code based on a space filling curve on the basis of the quadtree partitioning of triangular, diamond and hexagonal spherical grids of the same polyhedron. That is, according to the coding characteristics of different cell types and different space filling curves, the mapping rule library of triangle, diamond and hexagon discrete grids and the mapping rule table of icosahedral cross face cells and vertex cells are established, and the direct coding conversion from one grid to another grid is realized. These are called GtoG rules for short.

The parts of this article are arranged as follows: The second part introduces the spherical grid subdivision structure and filling curve coding principle. The third part defines and designs a coding mapping rule library for diamond, triangular and hexagonal grids. In the fourth part, experiments verify the feasibility and efficiency of the coding mapping in this paper. The fifth and sixth parts are discussion and conclusion.

2. Spherical Polyhedron Grid Subdivision and Spatial Index

2.1. Spherical Polyhedron Grid Subdivision

It can be seen from Table 1 that most of the current studies are based on spherical triangle, diamond and hexagonal discrete grids systems of regular octahedrons and icosahedrons. The index structures include a row–column coordinate index, spatial filling curve and hierarchical structure code. Among them, the icosahedron is closer to the sphere than the octahedron, and it can be refined to obtain a truncated icosahedron, which has a smaller cell area deformation under the equal-area projection. The hierarchical structure code and row–column coordinates only consider the distribution characteristics of cell space, without considering the requirements of cell data storage and fast access. The space filling curve code takes into account the adjacent relationship of cells and is more efficient when performing neighborhood operations. The cell data can be physically stored according to the filling curve, which greatly improves the efficiency of data access. Therefore, this paper selects the icosahedron as the basic polyhedron of the spherical grid partitioning and constructs the filling curve code of the spherical triangle, diamond and hexagonal grid systems.

First, the regular polyhedron is expanded on the plane, and the regular icosahedron plane is expanded into 20 triangles or 10 diamonds, as shown in Figure 1. The subdivision rule of diamond, triangle and hexagon grids in this paper is a recursive quadtree partition in which a parent cell generates four sub cells. The specific partition method is as follows.

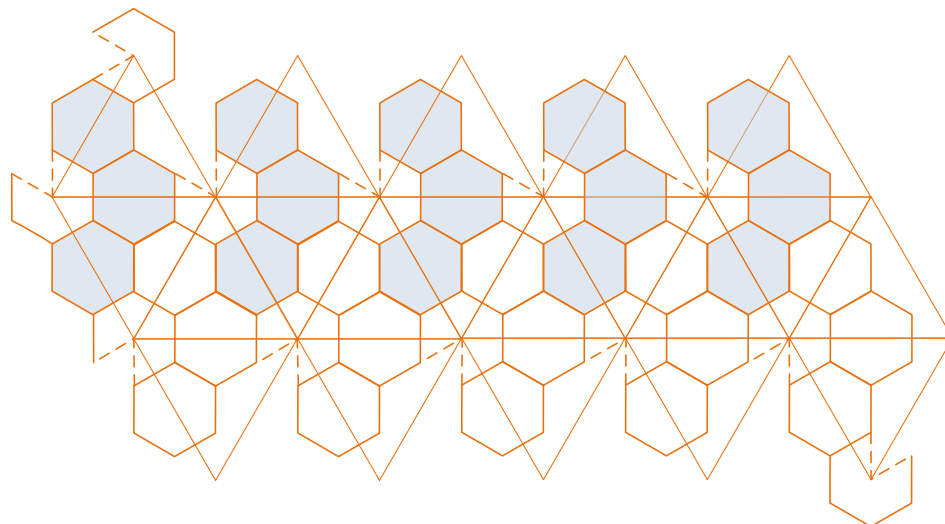


Figure 1. The first level hexagonal and pentagonal cells on the icosahedron.

On each triangular surface of the icosahedron, the midpoint of each triangular edge is connected in turn to obtain four triangular child cells (Figure 2b) and recursively subdivide until a triangle grid of the desired resolution is generated.

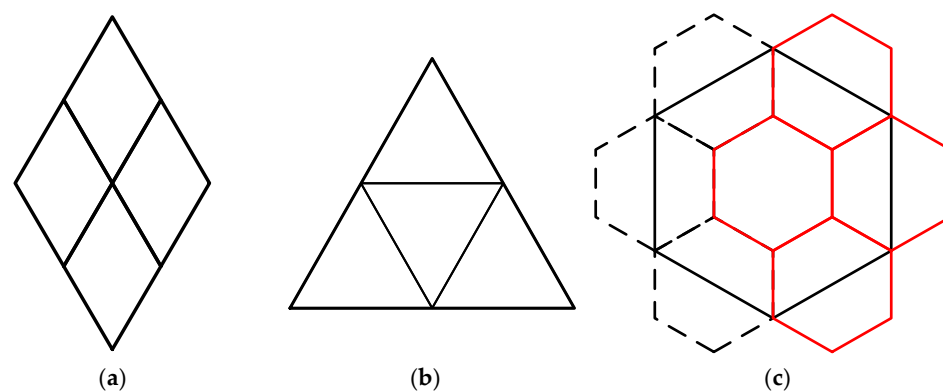


Figure 2. Different grid division rules. (a) Diamond hierarchical subdivision; (b) Triangular hierarchical subdivision; (c) Hexagonal hierarchical subdivision.

The midpoints of the ten initial diamond edges expanded by the icosahedron are taken. The midpoints of the opposite edges are connected, respectively, to form four child diamonds (Figure 2a) and recursively subdivided until a diamond grid of the desired resolution is generated.

The grid subdivision of diamond and triangle on an icosahedron is simple, and the coverage areas of the parent cells and the four child cells of diamonds and triangles completely coincide. However, the hexagonal grid subdivision is more complex, because it is necessary to determine the initial diamond of the cross-border hexagon. For example, when constructing an aperture 4 hexagonal grid in this paper, we firstly stipulate that there are four hexagonal cells on an initial diamond, and that there are two separate hexagons at the north and south poles, as shown in Figure 1. When folding the expanded icosahedron together, the hexagon at the twelve vertices of the icosahedron needs to be clipped into a pentagon (Figure 1). In addition, the hexagonal parent cell cannot completely cover

the child cell, and the affiliation between the parent cell and the child cell needs to be determined to ensure the uniqueness of the code. In this paper, in order to satisfy the quadtree structure, a hexagon parent cell generates the four red hexagon sub cells on the right in Figure 2c during hierarchical division, which ensures the uniqueness of affiliation, that is, that one sub cell has one parent cell. Divide the first level hexagonal cells recursively according to the above hexagonal quadtree subdivision rules until the desired resolution of the hexagonal grid is reached, and the generated child cells and the parent cell belong to the same initial diamond. The two pentagons of the north and south poles do not generate child cells, but the edge length of the pentagons decreases in equal proportion with the recursive subdivision.

This paper uses the Snyder equal-area projection to realize the mapping from the icosahedron to the sphere. The Snyder equal-area polyhedron projection takes the center of each face of the polyhedron as the projection center, uses azimuth projection to determine the boundary of the corresponding area of each face on the spherical surface, and then further obtains the corresponding relationship within the area according to the projection conditions and finally obtains the equal-area global discrete grid, as shown in Figure 3.

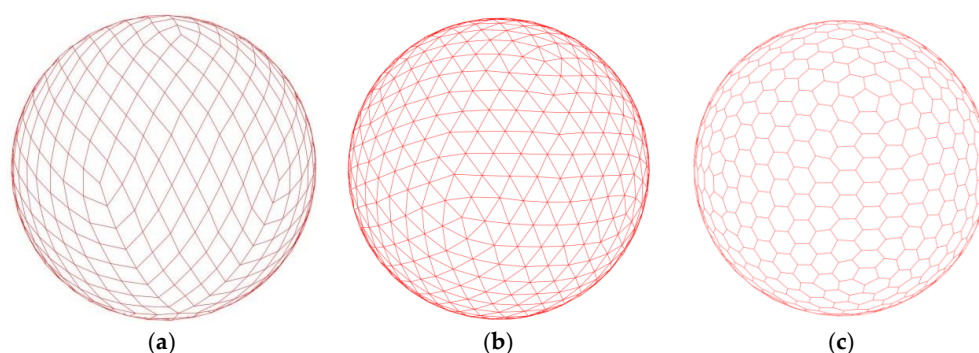


Figure 3. Different spherical grids. (a) diamond grid; (b) triangular grid; and (c) hexagonal grid.

2.2. Filling Curve of Spherical Grid

On the basis of quadtree subdivision, the *Hilbert* curve and *Z-curve* space indexes of the spherical diamond, triangular and hexagonal grids are established, respectively. The core coding rule is the combination of the basic diamond code and spatial filling curve code.

First, four-bit binary numbers are used to represent the code $D(0001-1010)$ of the initial ten diamonds, as shown in Figure 4. Combined with the partition structure of the diamond grid and the definition of the space filling curve, the *Hilbert* curve and *Z-curve* are established for the diamond grid, using an integer of m bits (on the x64 computer, $m = 64$, which can be adjusted according to the number of bits in the system; this paper uses $m = 64$ to illustrate the encoding method) for encoding. The upper four bits of each diamond cell code DC are the initial diamond code D , and the last 60 bits are the space filling curve code S of the diamond cell. Figure 5b shows the *Z-curve* of the second level diamond grid of the initial diamond 0001. In order to fully display the code in the cell, the code is expressed in the form of hexadecimal abbreviation, where N represents fourteen zeros.

$$DC = (D \ll 60) | S \quad (1)$$

Triangles distinguish upper and lower triangles, and the upper and lower triangles form a diamond (Figure 5c), which can be encoded by diamond (Figure 5c). The triangle cell code TC is obtained by adding the direction code to the code of the diamond where the triangle cell is located. The last bit of TC represents the direction of the cell, 0 represents the lower triangle and 1 represents the upper triangle. Figure 5c shows the *Z-curve* of

the second level triangular grid of the initial diamond 0001. The code is expressed in hexadecimal abbreviation, where M represents thirteen zeros.

$$TC = \begin{cases} (S \ll 1)|(D \ll 60) & \text{lower triangle} \\ (S \ll 1|1)|(D \ll 60) & \text{upper triangle} \end{cases} \quad (2)$$

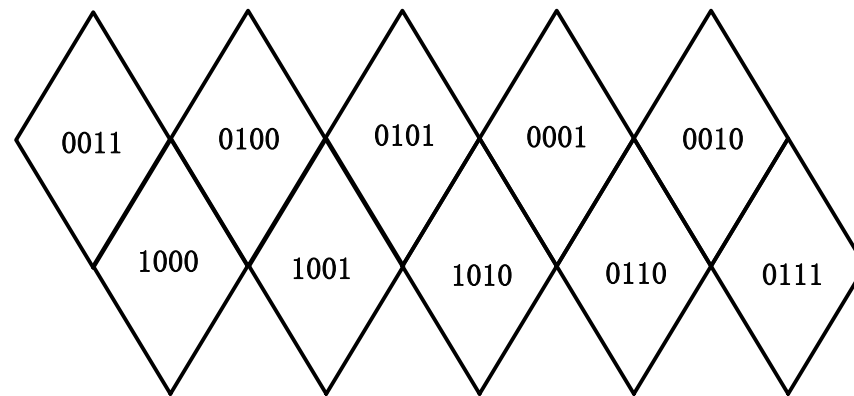


Figure 4. Basic diamond code on icosahedron.

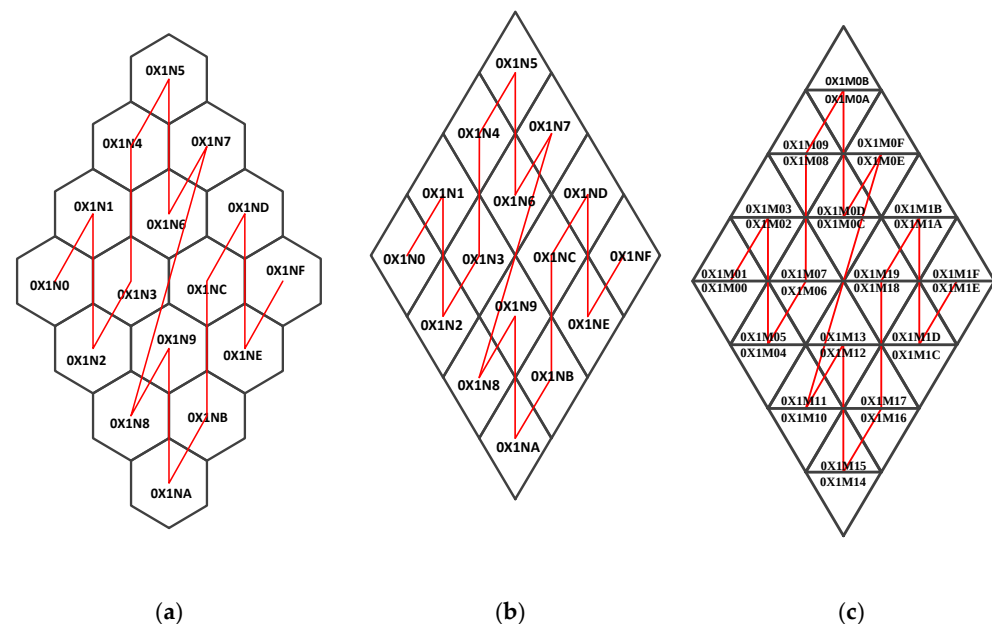


Figure 5. Second level Z-curve of different grids in a basic diamond plane. (a) Hexagon Z-curve; (b) diamond Z-curve; (c) triangle Z-curve.

Since the hexagonal cells cannot completely cover the sphere, there are twelve pentagons at the vertices of the icosahedron, of which 10 pentagons have subordinate initial diamonds and two pentagons do not. In order to uniformly manage the code of the hexagonal grid, according to the initial diamond coding rules, the other two pentagon codes are 0000 and 1011, respectively. The upper four bits of hexagonal cell code HC are the initial diamond code, and the last 60 bits are the hexagonal space filling curve code S . Figure 5a shows the Z-curve of the second level hexagonal grid of the initial diamond 0001. The code is expressed in hexadecimal abbreviation, where N represents fourteen zeros.

$$HC = (D \ll 60)|S \quad (3)$$

3. Direct Mapping Method for Diamond, Triangular and Hexagonal Grid Code

In this section, according to the geometric correspondences of the diamond, triangle and hexagon on the basic diamond, and the adjacent characteristics of different grid cells expressing spatial point, line and surface data, the coding mapping rule library of different spherical grid systems is designed. The direct mapping operation between the *Hilbert* curve and *Z-curve* coding of spherical aperture 4 diamond, triangle and hexagon grids can be realized. It mainly includes the following two steps.

1. Different filling curves transformations of the same cell type: The conversion between the *Hilbert* curve and *Z-curve* is realized by bit operation to obtain a spherical discrete grid system with the same spatial index structure and different element types;
2. Spherical grid coding transformation of different cell types: According to the adjacent characteristics of different grid cells expressing spatial point, line and surface data and the subdivision structure characteristics of different grids, this paper respectively designs the mutual conversion rules of the initial diamond in-plane and cross boundary cells.

3.1. Different Space Filling Curve Coding Mapping of Spherical Grid

In order to realize fast mapping between spherical grids with different spatial indexes and cell types, the grids must first be converted into the same spatial index. In this section, the conversion methods of the *Hilbert* curve and *Z-curve* spatial indexes of diamond, triangle and hexagon are given, respectively. First, set the triangular *Hilbert* curve and *Z-curve* as *THC* and *TZC*, the diamond *Hilbert* curve and *Z-curve* as *DHC* and *DZC*, and the hexagonal *Hilbert* curve and *Z-curve* as *HHC* and *HZC* for short, respectively. *DHC* and *DZC*, *THC* and *TZC*, and *HHC* and *HZC* are mapped to each other directly through bit operation. The coding mapping rules of hexagon and diamond are the same; therefore, only the mutual mapping rules of *DHC* and *DZC* are given here. *DHC* and *DZC* can obtain the traditional *Hilbert* curve and *Z-curve* by removing the initial diamond code (Figure 6). *THC* and *TZC* can obtain the traditional *Hilbert* curve and *Z-curve* by removing the initial diamond code and the direction code. Then, the rotation algorithm is used to realize the mutual conversion of the *Hilbert* curve and *Z-curve*.

$$\text{Hilbert} = \text{DHC} \& 0 \times 0\text{FFFFFFFFFFFFFFF} \quad (4)$$

$$\text{Z} = \text{DZC} \& 0 \times 0\text{FFFFFFFFFFFFFFF} \quad (5)$$

$$\text{Hilbert} = (\text{THC} \& 0 \times 0\text{FFFFFFFFFFFFFFF}) \gg 1 \quad (6)$$

$$\text{Z} = (\text{TZC} \& 0 \times 0\text{FFFFFFFFFFFFFFF}) \gg 1 \quad (7)$$

When turning a *Hilbert* curve into a *Z-curve*, first the rotation algorithm is used to turn the *m*-order *Hilbert* curve into a 1-order *Hilbert* recursive curve (Figure 6c). The *Hilbert* curve is represented by the quaternary number *A*, whose total bits are *m*, and the *i*th bit is defined as the number of bits counted from the low bit. The specific implementation process of the rotation algorithm is to process *A* bit by bit from the highest bit. If the *i*th bit is 0, the number 1 between the (*i* − 1)th and the 1st bit becomes 3, and the number 3 becomes 1. If the *i*th bit is 3, the number 0 between the (*i* − 1)th and 1st bit becomes 2, and the number 2 becomes 0. Then, the quaternary number *B* of the first-order *Hilbert* recursive curve is converted into a *Z-curve*. The specific method is to change the number 2 between the *m*th and 1st bits of the quaternary number *B* into 3, and to change the number 3 into 2, so that the *Z-curve* corresponding to the first-order *Hilbert* recursive curve can be obtained, which is also the *Z-curve* corresponding to the *m*-order *Hilbert* curve. In the same way, to convert the *Z-curve* to the *Hilbert* curve, first convert the *Z-curve* to a first-order *Hilbert* recursive curve according to the above inverse operation of converting the first-order *Hilbert* recursive curve (Figure 6c) to the *Z-curve*. Then, the first-order *Hilbert* recursive curve is rotated into an *m*-order *Hilbert* curve coding according to the inverse operation of the above rotation algorithm.

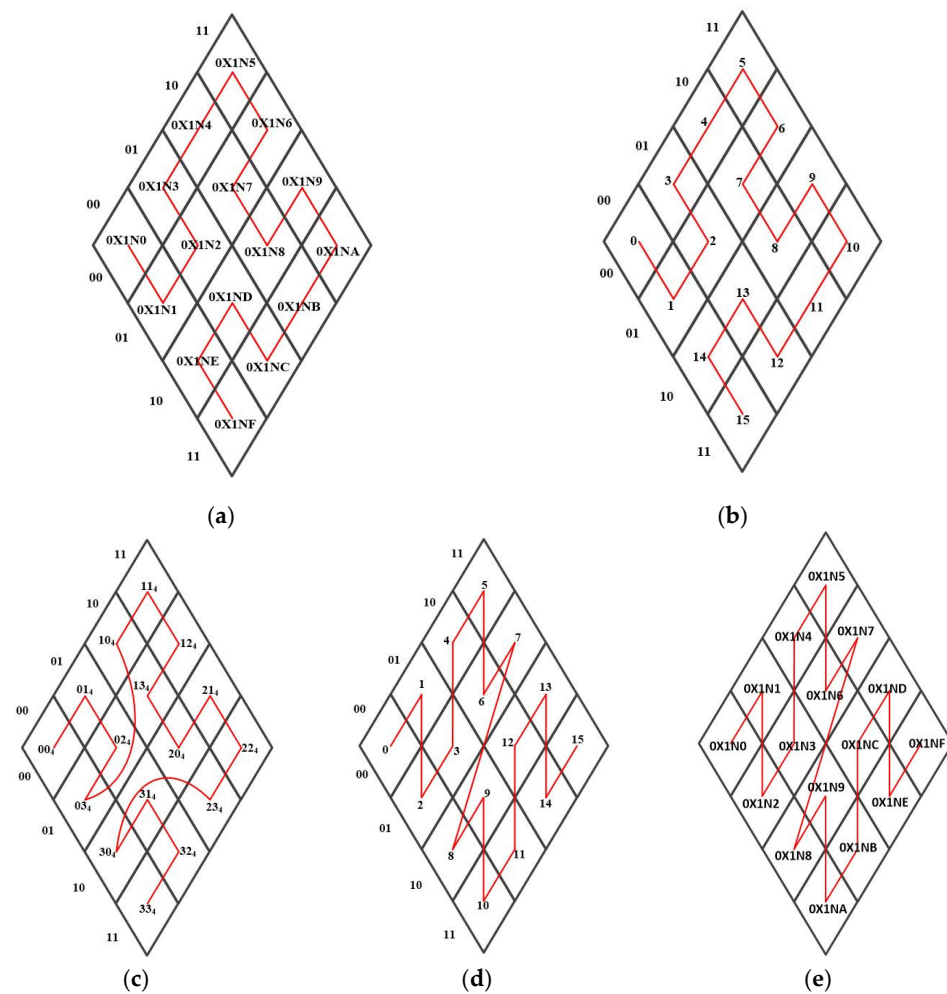


Figure 6. The process of converting DHC code to DZC code of the second level. (a) Second level DHC code; (b) Second level Hilbert curve; (c) 1-order Hilbert recursive curve; (d) Second level Z-curve; (e) Second level DZC code.

3.2. The Mapping Rules of Spherical Grids with Different Cell Types

After converting different space filling curve codes into the same space filling curve code, the next step is to convert the different cell type grids. According to the above-defined spherical aperture 4 diamond, triangle and hexagon grid subdivisions and coding method, the mutual conversion rules of diamond, triangular and hexagonal grids with the same space filling curve code are designed, yielding the following six types of conversion rules: Triangle code to Hexagon code (*TtoHcode*), Hexagon code to Triangle code (*HtoTcode*), Hexagon code to Diamond code (*HtoDcode*), Diamond code to Hexagon code (*DtoHcode*), Diamond code to Triangle code (*DtoTcode*), and Triangle code to Diamond code (*TtoDcode*). The specific methods of the six mapping rules are given below.

3.2.1. Mapping Rules of Triangular and Hexagonal Code

When a triangular grid is transformed into a hexagonal grid, the transformation from triangle to hexagon is realized with the help of the diamond where the triangle is located. Removing the direction code of the triangle code yields the diamond code. There is a one-to-one correspondence between the diamond and the hexagon on the n th level. Therefore, the hexagon code and diamond code corresponding to the triangle are the same.

$$HC = ((TC \& 0 \times 0FFFFFFF) \gg 1) | (TC \& 0 \times F0000000) \quad (8)$$

When transforming a hexagonal grid code to a triangular grid code, the corresponding conversion rules are defined according to the characteristics of the spatial data. Spatial data includes spatial point data, spatial line data and spatial surface data. The *HtoTcode* rules of spatial point data are different from those of spatial discrete line and surface data. Their corresponding conversion rules are given below.

Discrete spatial points are represented by grid cell codes on hexagonal grids and triangular grids, respectively. Therefore, the conversion of spatial point data is the conversion of discrete hexagonal cell codes and triangular cell codes. Regardless of whether the hexagon belongs to the upper triangle or the lower triangle of the corresponding diamond, the maximum offset error between the center point of the hexagonal cell and the corresponding triangular cell is half of the diameter of the hexagonal grid cell. It is specified herein that the hexagon of the n th layer corresponds to the upper triangle of the diamond of the n th layer. In Figure 7, the hexagonal cells 0, 1, 2 and 3 correspond to the upper triangles of diamonds A, B, C and D, respectively. Therefore, the corresponding triangular code TC can be obtained by adding the hexagonal cell code HC and the direction code of the corresponding triangular cell.

$$TC = (((HC \& 0 \times 0FFFFFFF) \ll 1) | 1) | (HC \& 0 \times F000000000000000) \quad (9)$$

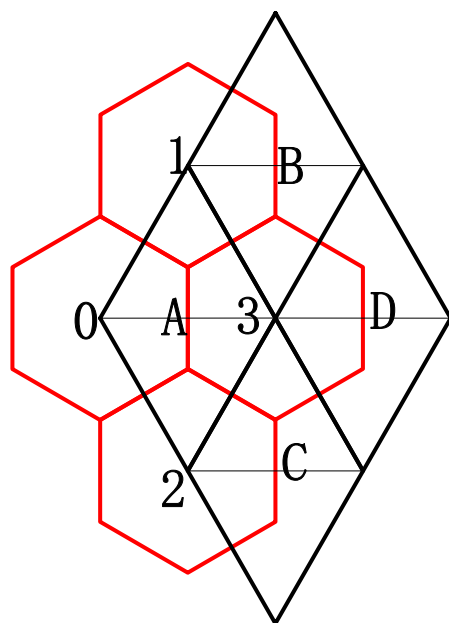


Figure 7. Corresponding relationship between hexagon grid and triangle grid.

The *HtoTcode* rule of spatial discrete line and surface data involves the transformation problem of the boundary and the vertex hexagonal cell of the basic diamond. According to the position of the hexagonal grid cell on the basic diamond, the hexagonal cells are divided into the following 4 categories and 11 subcategories (as shown in Figure 8): A is the cell in the basic diamond area, and its six adjacent cells are all in a basic diamond; B_1 , B_2 , B_3 and B_4 are the basic diamond boundary cells; C_1 , C_2 and C_3 are the basic diamond corner cells; E_1 , E_2 and E_3 are the vertex cells of icosahedron, which need to be cut into a pentagon. Among them, E_1 and E_2 are different from the basic diamond to which other cell types belong. The basic diamond of E_1 is the virtual basic diamond 0, and the basic diamond of E_2 is the virtual basic diamond 11. In addition, the adjacency relations of the applied cells during the transformation ensure the continuity of spatial discrete lines and surfaces.

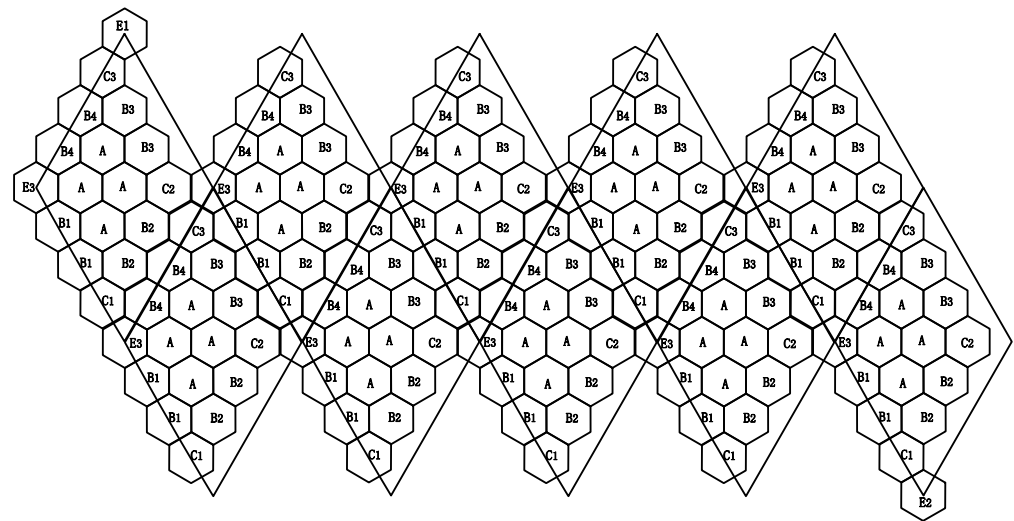


Figure 8. Icosahedral hexagonal cell type.

In order to efficiently realize the neighborhood search of grid cells, the hexagonal cells are represented by row–column coordinates, wherein the hexagonal code $Hcode$ is represented as $(D, (x_n, y_n))$, D represents the basic diamond number and (x_n, y_n) represents the coordinate position of the hexagonal cell in the basic diamond D . The triangle code $Tcode$ is represented as $(D, (x_n, y_n), 0)$ or $(D, (x_n, y_n), 1)$, wherein D represents the basic diamond number, (x_n, y_n) represents the coordinate position of the diamond where the triangle is located in the basic diamond D , 0 represents the lower triangle of the diamond $(D, (x_n, y_n))$ and 1 represents the upper triangle of the diamond $(D, (x_n, y_n))$. When selecting a triangle, it is necessary to judge whether the two adjacent diamond cells are edge adjacent cells or corner adjacent cells. If two diamond cells are edge adjacent, take two triangles with a common edge (Figure 9a); if the two diamond cells are corner adjacent, it is necessary to determine whether the $(n - 1)$ th triangle is an upper triangle or a lower triangle, and to determine the following triangle according to the direction of the $(n - 1)$ th triangle (Figure 9b,c).

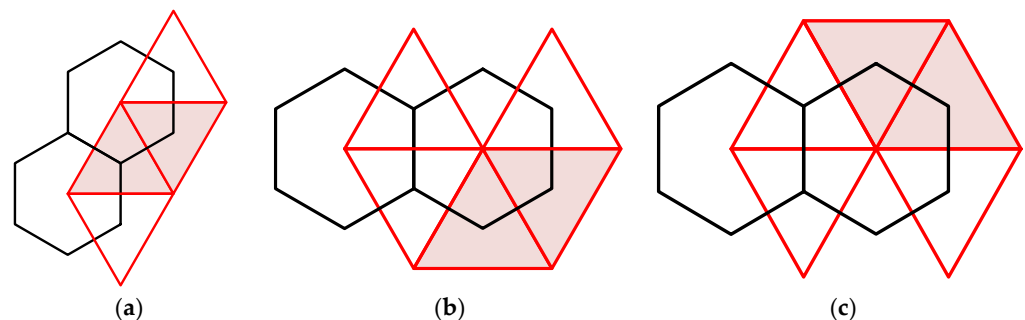


Figure 9. Triangle selection rules of $HtoTcode$ for spatial discrete line and surface data. (a) Edge adjacency; (b) Corner adjacency and the $(n - 1)$ th triangle is a lower triangle; and (c) Corner adjacency and the $(n - 1)$ th triangle is an upper triangle.

Table 2 shows the GtoG mapping rule index table of 4 categories and 11 subcategories corresponding to different hexagonal cells. Table 3 shows the $HtoTcode$ mapping rule of class A in the basic diamond plane; Tables 4–7 are the $HtoTcode$ mapping rule for class B of basic diamond boundary cells; Tables 8–12 are the $HtoTcode$ mapping rule for class C of basic diamond corner cells; and Tables 13–15 are the $HtoTcode$ mapping rule for class E of basic diamond vertex cells. In Tables 4–15, HD represents the basic diamond of the hexagon and H represents the hexagonal cell.

Table 2. GtoG mapping rule index table corresponding to hexagonal cell type.

	Hexagonal Cell Type	Corresponding Rule Table
Cells in the basic diamond	A	Table 3
Basic diamond boundary cells	B_1	Table 4
	B_2	Table 5
	B_3	Table 6
	B_4	Table 7
Basic diamond corner cells	C_1	Tables 8 and 9
	C_2	Table 10
	C_3	Tables 11 and 12
Icosahedral vertex cells	E_1	Table 13
	E_2	Table 14
	E_3	Table 15

Table 3. Htotcode rule of the $(n - 1)$ th hexagon cell A.

H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$x_{n-1} = x_n$ and $y_{n-1} < y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D, (x_n, y_n), 0)$
$x_{n-1} = x_n$ and $y_{n-1} > y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D, (x_n, y_n), 1)$
$y_{n-1} = y_n$ and $x_{n-1} < x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D, (x_n, y_n), 1)$
$y_{n-1} = y_n$ and $x_{n-1} > x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D, (x_n, y_n), 0)$
$x_{n-1} < x_n$ and $y_{n-1} < y_n$	0	$(D, (x_{n-1} + 1, y_{n-1}), 1)$ and $(D, (x_n, y_n), 0)$
$x_{n-1} < x_n$ and $y_{n-1} > y_n$	1	$(D, (x_{n-1}, y_{n-1} + 1), 0)$ and $(D, (x_n, y_n), 1)$
$x_{n-1} > x_n$ and $y_{n-1} < y_n$	0	$(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D, (x_n, y_n), 0)$
$x_{n-1} > x_n$ and $y_{n-1} > y_n$	1	$(D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D, (x_n, y_n), 1)$

Table 4. Htotcode rule of the $(n - 1)$ th hexagon cell B_1 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$D = 1$	$D + 9$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 9, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D + 9, (x_n + 1, y_n), 1)$ and $(D + 9, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 9, (x_n, y_n), 1)$
$1 < D \leq 5$	$D + 4$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D + 4, (x_n + 1, y_n), 1)$ and $(D + 4, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 1)$
$D = 6$	$D + 4$	$2^l - x_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 0)$
		$2^l - x_{n-1} \neq y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 0)$
$6 < D \leq 10$	$D - 1$	$2^l - x_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$
		$2^l - x_{n-1} \neq y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$

Table 5. Htotcode rule of the $(n - 1)$ th hexagon cell B_2 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$1 \leq D \leq 5$	$D + 5$	$y_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 5, (x_n, y_n), 1)$
		$y_{n-1} \neq y_n$	1	$(D, (x_{n-1}, y_{n-1} + 1), 0)$ and $(D + 5, (x_n, y_n), 1)$
		$y_{n-1} \neq y_n$	0	$(D + 5, (x_n, y_n - 1), 1)$ and $(D + 5, (x_n, y_n), 0)$
$6 \leq D < 10$	$D + 1$	$2^l - x_n \neq y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_n, y_n), 0)$
		$2^l - x_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0), (D + 1, (x_n - 1, y_n), 0)$ and $(D + 1, (x_n, y_n), 1)$
$D = 10$	$D - 4$	$2^l - x_n \neq y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 4, (x_n, y_n), 0)$
		$2^l - x_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0), (D - 4, (x_n - 1, y_n), 0)$ and $(D - 4, (x_n, y_n), 1)$

Table 6. Htotcode rule of the $(n - 1)$ th hexagon cell B_3 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$1 \leq D < 5$	$D + 1$	$2^l - y_n = x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D + 1, (x_n, y_n - 1), 1)$ and $(D + 1, (x_n, y_n), 0)$
		$2^l - y_n \neq x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_n, y_n), 1)$
$D = 5$	$D - 4$	$2^l - y_n = x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D - 4, (x_n, y_n - 1), 1)$ and $(D - 4, (x_n, y_n), 0)$
		$2^l - y_n \neq x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 1)$
$6 \leq D < 10$	$D - 4$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D - 4, (x_n - 1, y_n), 0)$ and $(D - 4, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D, (x_{n-1} + 1, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 0)$
$D = 10$	$D - 9$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 9, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D - 9, (x_n - 1, y_n), 0)$ and $(D - 9, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D, (x_{n-1} + 1, y_{n-1}), 1)$ and $(D - 9, (x_n, y_n), 0)$

Table 7. Htotcode rule of the $(n - 1)$ th hexagon cell B_4 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$D = 1$	$D + 4$	$2^l - y_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0), (D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D + 4, (x_n, y_n), 1)$
		$2^l - y_{n-1} \neq x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 4, (x_n, y_n), 1)$
$1 < D \leq 5$	$D - 1$	$2^l - y_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0), (D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 1, (x_n, y_n), 1)$
		$2^l - y_{n-1} \neq x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_n, y_n), 1)$
$6 \leq D \leq 10$	$D - 5$	$y_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 5, (x_n, y_n), 0)$
		$y_{n-1} \neq y_n$	1	$(D - 5, (x_n, y_n + 1), 0)$ and $(D - 5, (x_n, y_n), 1)$
		$y_{n-1} \neq y_n$	0	$(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 5, (x_n, y_n), 0)$

Table 8. Htotcode rule of the $(n - 1)$ th hexagon cell C_1 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$D = 1$	$D + 5$	$y_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 5, (x_n, y_n), 1)$
		$y_n \neq y_{n-1}$	0	$(D + 5, (x_n, y_n - 1), 1)$ and $(D + 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	1	$(D, (x_{n-1}, y_{n-1} + 1), 0)$ and $(D + 5, (x_n, y_n), 1)$
	$D + 9$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 9, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D + 9, (x_n + 1, y_n), 1)$ and $(D + 9, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 9, (x_n, y_n), 1)$
$1 < D \leq 5$	$D + 4$	$x_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 1)$
		$x_{n-1} \neq x_n$	0	$(D + 4, (x_n + 1, y_n), 1)$ and $(D + 4, (x_n, y_n), 0)$
		$x_{n-1} \neq x_n$	1	$(D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 1)$
	$D + 5$	$y_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 5, (x_n, y_n), 1)$
		$y_n \neq y_{n-1}$	0	$(D + 5, (x_n, y_n - 1), 1)$ and $(D + 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	1	$(D, (x_{n-1}, y_{n-1} + 1), 0)$ and $(D + 5, (x_n, y_n), 1)$
$D = 6$	$D + 1$	–	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_n, y_n), 0)$
	$D + 4$	$2^l - x_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 0)$
		$2^l - x_{n-1} \neq y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 0)$
	$D + 1$	–	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_n, y_n), 0)$
$6 < D < 10$	$D - 1$	$2^l - x_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$
		$2^l - x_{n-1} \neq y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$
$D = 10$	$D - 4$	–	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 4, (x_n, y_n), 0)$
	$D - 1$	$2^l - x_{n-1} = y_n$	–	$(D, (x_{n-1}, y_{n-1}), 1), (D, (x_{n-1} - 1, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$
		$2^l - x_{n-1} \neq y_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$

Table 9. Htotcode rule of the $(n - 1)$ th hexagon cell C_1 and the n th hexagon cell E_2 .

HD_{n-1}	HD_n	HD_{n+1}	Triangle Cells Selected
$D = 6$	$D + 5$	$D + 4$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_{n+1}, y_{n+1}), 0)$
		$D + 3$	$(D + 3, (x_{n+1}, y_{n+1}), 0)$, $(D + 4, (2^l - 1, 0), 0)$ and $(D, (x_{n-1}, y_{n-1}), 0)$
		$D + 2$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D + 1, (2^l - 1, 0), 0)$ and $(D + 2, (x_{n+1}, y_{n+1}), 0)$
		$D + 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_{n+1}, y_{n+1}), 0)$
$D = 7$	$D + 4$	$D + 3$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 1, (2^l - 1, 0), 0)$ and $(D + 3, (x_{n+1}, y_{n+1}), 0)$
		$D + 2$	$(D + 2, (x_{n+1}, y_{n+1}), 0)$, $(D + 1, (2^l - 1, 0), 0)$ and $(D, (x_{n-1}, y_{n-1}), 0)$
		$D + 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_{n+1}, y_{n+1}), 0)$
		$D - 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_{n+1}, y_{n+1}), 0)$
$D = 8$	$D + 3$	$D + 2$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D + 1, (2^l - 1, 0), 0)$ and $(D + 2, (x_{n+1}, y_{n+1}), 0)$
		$D + 1$	$(D + 1, (x_{n+1}, y_{n+1}), 0)$ and $(D, (x_{n-1}, y_{n-1}), 0)$
		$D - 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_{n+1}, y_{n+1}), 0)$
		$D - 2$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 1, (2^l - 1, 0), 0)$ and $(D - 2, (x_{n+1}, y_{n+1}), 0)$
$D = 9$	$D + 2$	$D + 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_{n+1}, y_{n+1}), 0)$
		$D - 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_{n+1}, y_{n+1}), 0)$
		$D - 2$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 1, (2^l - 1, 0), 0)$ and $(D - 2, (x_{n+1}, y_{n+1}), 0)$
		$D - 3$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D + 1, (2^l - 1, 0), 0)$ and $(D - 3, (x_{n+1}, y_{n+1}), 0)$
$D = 10$	$D + 1$	$D - 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_{n+1}, y_{n+1}), 0)$
		$D - 2$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 1, (2^l - 1, 0), 0)$ and $(D - 2, (x_{n+1}, y_{n+1}), 0)$
		$D - 3$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 4, (2^l - 1, 0), 0)$ and $(D - 3, (x_{n+1}, y_{n+1}), 0)$
		$D - 4$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 4, (x_{n+1}, y_{n+1}), 0)$

Table 10. Htotcode rule of the $(n - 1)$ th hexagon cell C_2 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Triangle cells selected
$1 \leq D < 5$	$D + 5$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 5, (x_n, y_n), 1)$
	$D + 1$	$2^l - y_n = x_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 1)$, $(D + 1, (0, 0), 1)$ and $(D + 1, (x_n, y_n), 0)$
		$2^l - y_n \neq x_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_n, y_n), 1)$
$D = 5$	$D - 4$	$2^l - y_n = x_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 1)$, $(D - 4, (0, 0), 1)$ and $(D - 4, (x_n, y_n), 0)$
		$2^l - y_n \neq x_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 1)$
	$D + 5$	–	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 5, (x_n, y_n), 1)$
$6 \leq D < 10$	$D + 1$	$2^l - x_n = y_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D + 1, (0, 0), 0)$ and $(D + 1, (x_n, y_n), 1)$
		$2^l - x_n \neq y_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_n, y_n), 0)$
	$D - 4$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 0)$
$D = 10$	$D - 4$	$2^l - x_n = y_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D - 4, (0, 0), 0)$ and $(D - 4, (x_n, y_n), 1)$
		$2^l - x_n \neq y_{n-1}$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 4, (x_n, y_n), 0)$
	$D - 9$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 9, (x_n, y_n), 0)$

Table 11. Htotcode rule of the $(n - 1)$ th hexagon cell C_3 .

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$D = 1$	$D + 4$	$2^l - y_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D + 4, (x_n, y_n), 1)$
		$2^l - y_{n-1} \neq x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 4, (x_n, y_n), 1)$
	$D + 1$	–	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_n, y_n), 1)$
$1 < D < 5$	$D + 1$	–	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_n, y_n), 1)$
	$D - 1$	$2^l - y_{n-1} = x_n$	–	$(D, (x_{n-1}, y_{n-1}), 0)$, $(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 1, (x_n, y_n), 1)$
		$2^l - y_{n-1} \neq x_n$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_n, y_n), 1)$

Table 11. Cont.

HD_{n-1}	HD_n	H_{n-1} and H_n Coordinate Relationship	Td_{n-1}	Triangle Cells Selected
$D = 5$	$D - 4$	–	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 1)$
	$D - 1$	$2^l - y_{n-1} = x_n$ $2^l - y_{n-1} \neq x_n$	– –	$(D, (x_{n-1}, y_{n-1}), 0), (D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 1, (x_n, y_n), 1)$ $(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_n, y_n), 1)$
$6 \leq D < 10$	$D - 5$	$y_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	0	$(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	1	$(D - 5, (x_n, y_n + 1), 0)$ and $(D - 5, (x_n, y_n), 1)$
	$D - 4$	$x_n = x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 0)$
		$x_n \neq x_{n-1}$	1	$(D - 4, (x_n - 1, y_n), 0)$ and $(D - 4, (x_n, y_n), 1)$
		$x_n \neq x_{n-1}$	0	$(D, (x_{n-1} + 1, y_{n-1}), 1)$ and $(D - 4, (x_n, y_n), 0)$
$D = 10$	$D - 5$	$y_n = y_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	0	$(D, (x_{n-1}, y_{n-1} - 1), 1)$ and $(D - 5, (x_n, y_n), 0)$
		$y_n \neq y_{n-1}$	1	$(D - 5, (x_n, y_n + 1), 0)$ and $(D - 5, (x_n, y_n), 1)$
	$D - 9$	$x_n = x_{n-1}$	–	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 9, (x_n, y_n), 0)$
		$x_n \neq x_{n-1}$	1	$(D - 9, (x_n - 1, y_n), 0)$ and $(D - 9, (x_n, y_n), 1)$
		$x_n \neq x_{n-1}$	0	$(D, (x_{n-1} + 1, y_{n-1}), 1)$ and $(D - 9, (x_n, y_n), 0)$

Table 12. Htotcode rule of the $(n - 1)$ th hexagon cell C_3 and the n th hexagon cell E_1 .

HD_{n-1}	HD_n	HD_{n+1}	Triangle Cells Selected
$D = 1$	$D - 1$	$D + 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_{n+1}, y_{n+1}), 1)$
		$D + 2$	$(D, (x_{n-1}, y_{n-1}), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D + 2, (x_{n+1}, y_{n+1}), 1)$
		$D + 3$	$(D, (x_{n-1}, y_{n-1}), 1), (D + 4, (0, 2^l - 1), 1)$ and $(D + 3, (x_{n+1}, y_{n+1}), 1)$
		$D + 4$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 4, (x_{n+1}, y_{n+1}), 1)$
$D = 2$	$D - 2$	$D - 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_{n+1}, y_{n+1}), 1)$
		$D + 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_{n+1}, y_{n+1}), 1)$
		$D + 2$	$(D, (x_{n-1}, y_{n-1}), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D + 2, (x_{n+1}, y_{n+1}), 1)$
		$D + 3$	$(D, (x_{n-1}, y_{n-1}), 1), (D - 1, (0, 2^l - 1), 1)$ and $(D + 3, (x_{n+1}, y_{n+1}), 1)$
$D = 3$	$D - 3$	$D - 2$	$(D, (x_{n-1}, y_{n-1}), 1), (D - 1, (0, 2^l - 1), 1)$ and $(D - 2, (x_{n+1}, y_{n+1}), 1)$
		$D - 1$	$(D - 1, (x_{n+1}, y_{n+1}), 1)$ and $(D, (x_{n-1}, y_{n-1}), 1)$
		$D + 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_{n+1}, y_{n+1}), 1)$
		$D + 2$	$(D, (x_{n-1}, y_{n-1}), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D + 2, (x_{n+1}, y_{n+1}), 1)$
$D = 4$	$D - 4$	$D - 3$	$(D, (x_{n-1}, y_{n-1}), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D - 3, (x_{n+1}, y_{n+1}), 1)$
		$D - 2$	$(D - 2, (x_{n+1}, y_{n+1}), 1), (D - 1, (0, 2^l - 1), 1)$ and $(D, (x_{n-1}, y_{n-1}), 1)$
		$D - 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_{n+1}, y_{n+1}), 1)$
		$D + 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 1, (x_{n+1}, y_{n+1}), 1)$
$D = 5$	$D - 5$	$D - 4$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 4, (x_{n+1}, y_{n+1}), 1)$
		$D - 3$	$(D - 3, (x_{n+1}, y_{n+1}), 1), (D - 4, (0, 2^l - 1), 1)$ and $(D, (x_{n-1}, y_{n-1}), 1)$
		$D - 2$	$(D, (x_{n-1}, y_{n-1}), 1), (D - 1, (0, 2^l - 1), 1)$ and $(D - 2, (x_{n+1}, y_{n+1}), 1)$
		$D - 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_{n+1}, y_{n+1}), 1)$

Table 13. Htotcode rule of the $(n - 1)$ th hexagon cell E_1 .

HD_{n-1}	HD_n	HD_{n-2}	Triangle Cells Selected
$D = 0$	$D + 1$	$D + 2$	$(D + 1, (x_n, y_n), 1)$ and $(D + 2, (x_{n-2}, y_{n-2}), 1)$
		$D + 3$	$(D + 1, (x_n, y_n), 1), (D + 2, (0, 2^l - 1), 1)$ and $(D + 3, (x_{n-2}, y_{n-2}), 1)$
		$D + 4$	$(D + 1, (x_n, y_n), 1), (D + 5, (0, 2^l - 1), 1)$ and $(D + 4, (x_{n-2}, y_{n-2}), 1)$
		$D + 5$	$(D + 1, (x_n, y_n), 1)$ and $(D + 5, (x_{n-2}, y_{n-2}), 1)$
	$D + 2$	$D + 1$	$(D + 2, (x_n, y_n), 1)$ and $(D + 1, (x_{n-2}, y_{n-2}), 1)$
		$D + 3$	$(D + 2, (x_n, y_n), 1)$ and $(D + 3, (x_{n-2}, y_{n-2}), 1)$
		$D + 4$	$(D + 2, (x_n, y_n), 1), (D + 3, (0, 2^l - 1), 1)$ and $(D + 4, (x_{n-2}, y_{n-2}), 1)$
		$D + 5$	$(D + 2, (x_n, y_n), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D + 5, (x_{n-2}, y_{n-2}), 1)$
	$D + 3$	$D + 1$	$(D + 3, (x_n, y_n), 1), (D + 2, (0, 2^l - 1), 1)$ and $(D + 1, (x_{n-2}, y_{n-2}), 1)$
		$D + 2$	$(D + 2, (x_{n-2}, y_{n-2}), 1)$ and $(D + 3, (x_n, y_n), 1)$
		$D + 4$	$(D + 3, (x_n, y_n), 1)$ and $(D + 4, (x_{n-2}, y_{n-2}), 1)$
		$D + 5$	$(D + 3, (x_n, y_n), 1), (D + 4, (0, 2^l - 1), 1)$ and $(D + 5, (x_{n-2}, y_{n-2}), 1)$
	$D + 4$	$D + 1$	$(D + 4, (x_n, y_n), 1), (D + 5, (0, 2^l - 1), 1)$ and $(D + 1, (x_{n-2}, y_{n-2}), 1)$
		$D + 2$	$(D + 2, (x_{n-2}, y_{n-2}), 1), (D + 3, (0, 2^l - 1), 1)$ and $(D + 4, (x_n, y_n), 1)$
		$D + 3$	$(D + 4, (x_n, y_n), 1)$ and $(D + 3, (x_{n-2}, y_{n-2}), 1)$
		$D + 5$	$(D + 4, (x_n, y_n), 1)$ and $(D + 5, (x_{n-2}, y_{n-2}), 1)$
	$D + 5$	$D + 1$	$(D + 5, (x_n, y_n), 1)$ and $(D + 1, (x_{n-2}, y_{n-2}), 1)$
		$D + 2$	$(D + 2, (x_{n-2}, y_{n-2}), 1), (D + 1, (0, 2^l - 1), 1)$ and $(D + 5, (x_n, y_n), 1)$
		$D + 3$	$(D + 5, (x_n, y_n), 1), (D + 4, (0, 2^l - 1), 1)$ and $(D + 3, (x_{n-2}, y_{n-2}), 1)$
		$D + 4$	$(D + 5, (x_n, y_n), 1)$ and $(D + 4, (x_{n-2}, y_{n-2}), 1)$

Table 14. Htotcode rule of the $(n - 1)$ th hexagon cell E_2 .

HD_{n-1}	HD_n	HD_{n-2}	Triangle Cells Selected
$D = 11$	$D - 1$	$D - 2$	$(D - 1, (x_n, y_n), 0)$ and $(D - 2, (x_{n-2}, y_{n-2}), 0)$
		$D - 3$	$(D - 1, (x_n, y_n), 0), (D - 2, (2^l - 1, 0), 0)$ and $(D - 3, (x_{n-2}, y_{n-2}), 0)$
		$D - 4$	$(D - 1, (x_n, y_n), 0), (D - 5, (2^l - 1, 0), 0)$ and $(D - 4, (x_{n-2}, y_{n-2}), 0)$
		$D - 5$	$(D - 1, (x_n, y_n), 0)$ and $(D - 5, (x_{n-2}, y_{n-2}), 0)$
	$D - 2$	$D - 1$	$(D - 2, (x_n, y_n), 0)$ and $(D - 1, (x_{n-2}, y_{n-2}), 0)$
		$D - 3$	$(D - 2, (x_n, y_n), 0)$ and $(D - 3, (x_{n-2}, y_{n-2}), 0)$
		$D - 4$	$(D - 2, (x_n, y_n), 0), (D - 3, (2^l - 1, 0), 0)$ and $(D - 4, (x_{n-2}, y_{n-2}), 0)$
		$D - 5$	$(D - 2, (x_n, y_n), 0), (D - 1, (2^l - 1, 0), 0)$ and $(D - 5, (x_{n-2}, y_{n-2}), 0)$
	$D - 3$	$D - 1$	$(D - 3, (x_n, y_n), 0), (D - 2, (2^l - 1, 0), 0)$ and $(D - 1, (x_{n-2}, y_{n-2}), 0)$
		$D - 2$	$(D - 2, (x_{n-2}, y_{n-2}), 0)$ and $(D - 3, (x_n, y_n), 0)$
		$D - 4$	$(D - 3, (x_n, y_n), 0)$ and $(D - 4, (x_{n-2}, y_{n-2}), 0)$
		$D - 5$	$(D - 3, (x_n, y_n), 0), (D - 4, (2^l - 1, 0), 0)$ and $(D - 5, (x_{n-2}, y_{n-2}), 0)$
	$D - 4$	$D - 1$	$(D - 4, (x_n, y_n), 0), (D - 5, (2^l - 1, 0), 0)$ and $(D - 1, (x_{n-2}, y_{n-2}), 0)$
		$D - 2$	$(D - 2, (x_{n-2}, y_{n-2}), 0), (D - 3, (2^l - 1, 0), 0)$ and $(D - 4, (x_n, y_n), 0)$
		$D - 3$	$(D - 4, (x_n, y_n), 0)$ and $(D - 3, (x_{n-2}, y_{n-2}), 0)$
		$D - 5$	$(D - 4, (x_n, y_n), 0)$ and $(D - 5, (x_{n-2}, y_{n-2}), 0)$
	$D - 5$	$D - 1$	$(D - 5, (x_n, y_n), 0)$ and $(D - 1, (x_{n-2}, y_{n-2}), 0)$
		$D - 2$	$(D - 2, (x_{n-2}, y_{n-2}), 0), (D - 1, (2^l - 1, 0), 0)$ and $(D - 5, (x_n, y_n), 0)$
		$D - 3$	$(D - 5, (x_n, y_n), 0), (D - 4, (2^l - 1, 0), 0)$ and $(D - 3, (x_{n-2}, y_{n-2}), 0)$
		$D - 4$	$(D - 5, (x_n, y_n), 0)$ and $(D - 4, (x_{n-2}, y_{n-2}), 0)$

Table 15. *Htotcode* rule of the $(n - 1)$ th hexagon cell E_3 .

HD_{n-1}	HD_n	Triangle Cells Selected
$D = 1$	$D + 4$ $D + 9$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D + 4, (x_n, y_n), 1)$ $(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 9, (x_n, y_n), 1)$
$1 < D \leq 5$	$D - 1$ $D + 4$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 1, (x_n, y_n), 1)$ $(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 1)$
$D = 6$	$D - 5$ $D + 4$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 5, (x_n, y_n), 0)$ $(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 4, (x_n, y_n), 0)$
$6 < D \leq 10$	$D - 5$ $D - 1$	$(D, (x_{n-1}, y_{n-1}), 1)$ and $(D - 5, (x_n, y_n), 0)$ $(D, (x_{n-1}, y_{n-1}), 0)$ and $(D - 1, (x_n, y_n), 0)$

3.2.2. Mapping Rules of Diamond and Hexagonal Code

Mapping from hexagonal grid to diamond grid (*HtoDcode*): Since both the hexagonal and diamond grid adopt a quadtree partition structure, there is a one-to-one correspondence between the hexagonal cells and diamond cells at the same level in the basic diamond plane. In Figure 10a, the hexagonal cells 0, 1, 2 and 3 correspond to the diamond cells A, B, C and D, respectively. The space filling curve codes of hexagonal cells A, B₁, B₂, B₃, B₄, C₁, C₂, C₃ and E₃ are the same as the corresponding diamond codes, i.e., $DC = HC$. However, the hexagonal cells E₁ and E₂ have no corresponding diamond cells on the icosahedral diamond grid. This paper stipulates that, for the spatial point data, the diamond corresponding to the hexagonal cell E₁ is $(1, (0, y_{\max}))$, and the diamond corresponding to E₂ is $(10, (x_{\max}, 0))$. For the conversion of spatial line and surface data, the conversion process skips when encountering the hexagonal cells E₁ and E₂ and then processes the following cells.

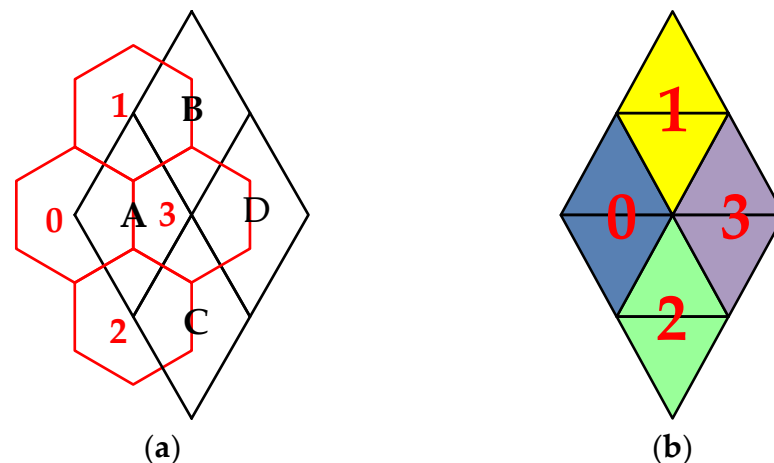


Figure 10. Corresponding relationship of different cells. (a) Corresponding relationship between hexagon grid and diamond grid; (b) Corresponding relationship between triangle grid and diamond grid.

Mapping from diamond grid to hexagonal grid (*DtoHcode*): For the aperture 4 diamond and hexagonal grids given above, there is a one-to-one correspondence between the diamond and the hexagon, and each diamond has its corresponding hexagon. Therefore, the hexagonal cell code is the same as the diamond code, i.e., $DC = HC$.

3.2.3. Mapping Rules of Diamond and Triangular Code

During the *DtoTcode* conversion of spatial point data, the diamond cell includes upper and lower triangles (Figure 10b). Whether the diamond cell belongs to the upper triangle or the lower triangle, the offset error of the center point of the diamond cell and the triangle

cell are the same. This paper specifies that the diamond cell of the n th level corresponds to the upper triangle of the diamond cell of the n th level.

$$TC = (((DC \& 0 \times 0FFFFFFFFFFFFFFF) \ll 1) | 1) | ((DC \& 0 \times F000000000000000) \quad (10)$$

During the *DtoTcode* conversion of spatial discrete line and surface data, since there is a one-to-one correspondence between the hexagonal cells and diamond cells on the icosahedron, the *DtoTcode* conversion rules are the same as the *HtoTcode* conversion rules for the spatial discrete line and surface data of hexagonal cells A , B_1 , B_2 , B_3 , B_4 , C_1 , C_2 , C_3 and E_3 . The mapping from diamond code to triangle code can be realized according to Tables 3–15.

During the *TtoDcode* conversion, the attribution of the triangular cell is clear, and the diamond where the triangle is located is the corresponding diamond cell. The diamond code DC can be obtained by removing the direction code of triangle code TC .

$$DC = ((TC \& 0 \times 0FFFFFFFFFFFFFFF) \gg 1) | (TC \& 0 \times F000000000000000) \quad (11)$$

4. Experimental Comparison and Analysis

This paper selects spatial point, line and surface entity data to analyze the efficiency and accuracy of the GtoG mapping method and the conventional longitude–latitude middleware method. *HtoDcode* and *DtoHcode* consume almost no time, since the hexagonal code is the same as the corresponding diamond code. The *HtoTcode* and *DtoTcode*, and *TtoDcode* and *TtoHcode* mapping rules are the same, so this experiment only calculates the conversion efficiency of *HtoTcode* and *TtoDcode*.

The development test environment utilizes the following: Windows10, intel(R) Core (TM)i5-8250U CPU @1.60 GHz 1.80 GHz, 64 bit, 8 GB, Visual Studio 2012, C++. Compile all programs into the release version.

4.1. Efficiency and Accuracy Comparison of Spatial Point GtoG Mapping

First, randomly select 10,000, 100,000, 1 million and 10 million spherical discrete points. Then, take the 20-level hexagonal (diamond or triangle) *Z-curve* corresponding to the random spatial points as the basic data, and calculate the conversion efficiency of the *HtoTcode* and *TtoDcode* respectively. From the results (as shown in Tables 16 and 17), it can be seen that the efficiency of *HtoTcode* and *TtoDcode* mapping is more than 3000 times (more than 3 orders of magnitude) higher than that of the longitude–latitude middleware method, and the efficiency of the mapping method grows higher and higher as the amount of data increases.

In addition, in the grid conversion, the centers of different grid cells do not completely coincide, so there is a certain deviation. Table 16 shows that the maximum center distance deviation of the hexagon to triangle (*HtoTcode*) mapping is 0.639 times the average diameter of the corresponding level hexagon cells. Table 17 shows that the maximum center distance deviation of the triangle to diamond (*TtoDcode*) mapping is 0.452 times of the average diameter of the corresponding level triangle cells. Therefore, in different grid conversion, the maximum offset error of the cell center point is limited within the corresponding level cell diameter. In practical application, the appropriate hierarchical grid can be selected for conversion according to the accuracy requirements.

Table 16. Comparison of efficiency and accuracy of *HtoTcode* and longitude–latitude expressing discrete points.

Points	<i>HtoTcode</i> Efficiency (ms)	Longitude- Latitude Efficiency (ms)	Efficiency Ratio	Maximum Distance Difference Between Center Points (m)	Average Diameter of 20 Level Hexagonal Cells (m)	Length Ratio
10^4	0.041	153	3731	4.90734	7.68543	0.639
10^5	0.408	1561	3825	4.90734	7.68543	0.639
10^6	4	17,518	4379	4.90734	7.68543	0.639
10^7	41	497,062	12,123	4.90734	7.68543	0.639

Table 17. Comparison of efficiency and accuracy of *TtoDcode* and longitude–latitude expressing discrete points.

Points	<i>TtoDcode</i> Efficiency (ms)	Longitude– latitude Efficiency (ms)	Efficiency Ratio	Maximum Distance difference Between Center Points (m)	Average Diameter of 20 Level Hexagonal Cells (m)	Length Ratio
10 ⁴	0.04	153	3825	2.45551	5.43442	0.452
10 ⁵	0.4	1544	3860	2.45551	5.43442	0.452
10 ⁶	4	17,197	4299	2.45551	5.43442	0.452
10 ⁷	43	492,169	11,445	2.45551	5.43442	0.452

4.2. Efficiency and Accuracy Comparison of Spatial Line GtoG Mapping

The Henan Province administrative boundary data in China are selected as the experimental data of the vector line (from Resources and Environment Science and Data Center, <https://www.resdc.cn/data.aspx?DATAID=200>, (accessed on 2 March 2022)).

The method of vector line discretization is the dimension reduction method [38]. The idea of dimension reduction is that the two-dimensional discrete line model is equivalently transformed into a one-dimensional closed path solution. According to the grid code mapping rules of spatial line and area data, the *HtoTcode* and *TtoDcode* mapping efficiency of Henan administrative boundary is calculated. From the results (as shown in Tables 18 and 19 and Figure 11), it can be seen that the efficiency of *HtoTcode* and *TtoDcode* mapping is 100 times and 1000 times higher, respectively, than that of the longitude–latitude middleware (dimension reduction method), and that the efficiency of the mapping method is increases with the increase in subdivision levels. Because the *HtoTcode* and *TtoDcode* mapping rules avoid the complex spherical geometric calculation of the optimal direction and shortest distance of the longitude–latitude method, the coding mapping of different grids can be realized only according to the adjacency and geometric correspondence of the cells.

Table 18. Comparison of efficiency and accuracy of *HtoTcode* and longitude–latitude expressing discrete lines.

Level	<i>HtoTcode</i> Efficiency (ms)	Longitude–Latitude (ms)	Efficiency Ratio	Maximum Distance Difference between Center Points (m)	Average Diameter of Hexagonal Cells (m)	Length Ratio
17	69	10,769	156	38.9844	61.48344	0.634
18	121	14,544	120	19.4925	30.74172	0.634
19	241	23,058	95	9.7465	15.37086	0.634
20	476	43,368	91	4.87417	7.68543	0.634
21	900	105,699	117	2.43893	3.842715	0.635
22	2312	306,969	132	1.22316	1.921358	0.637

Table 19. Comparison of efficiency and accuracy of *TtoDcode* and longitude–latitude expressing discrete lines.

Level	<i>TtoDcode</i> Efficiency (ms)	Longitude–Latitude (ms)	Efficiency Ratio	Maximum Distance Difference between Center Points (m)	Average Diameter of Triangular Cells (m)	Length Ratio
17	6	11,202	1867	31.9519	43.4754	0.735
18	12	15,065	1255	15.9762	21.7377	0.735
19	22	24,251	1102	7.98812	10.8688	0.735
20	43	46,911	1090	3.99632	5.43442	0.735
21	80	114,516	1431	2.00041	2.71721	0.736
22	168	339,406	2020	1.0047	1.3586	0.740

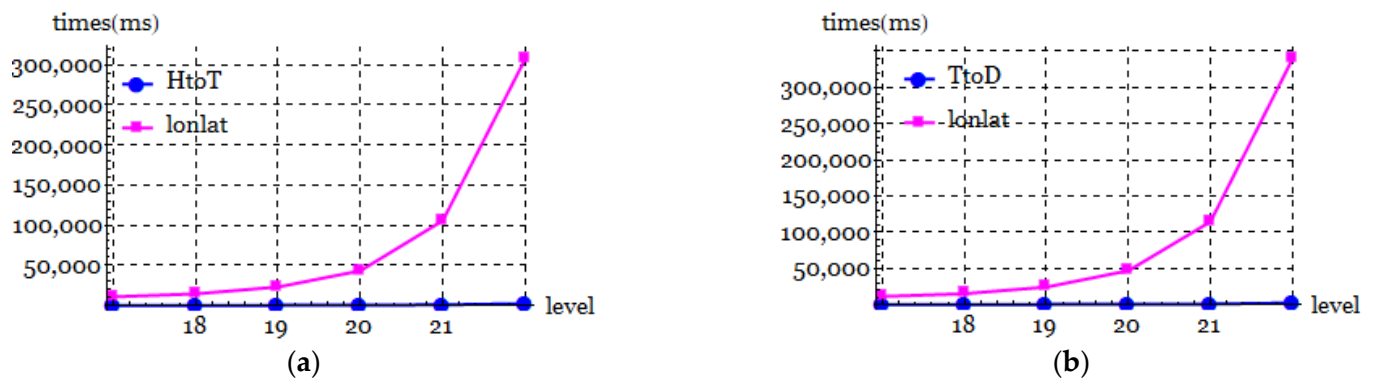


Figure 11. Efficiency comparison diagram between GtoG and longitude–latitude method expressing discrete lines. (a) Efficiency comparison curve between *HtoTcode* and longitude–latitude method, and (b) Efficiency comparison curve between *TtoDcode* and longitude–latitude method.

Figure 12 is the superposition diagram of the ninth level *HtoTcode* triangle discrete line and the basic data hexagon discrete line. Figure 13 is the superposition diagram of the ninth level *TtoDcode* diamond discrete line and the basic data triangle discrete line. It can be seen that there are certain deviations in the discrete lines expressed in different grids. It can be seen from Tables 18 and 19 that the maximum deviation value is less than the average diameter of the grid cells at corresponding levels. In practical application, the appropriate hierarchical grid can be selected for conversion according to the accuracy requirements.

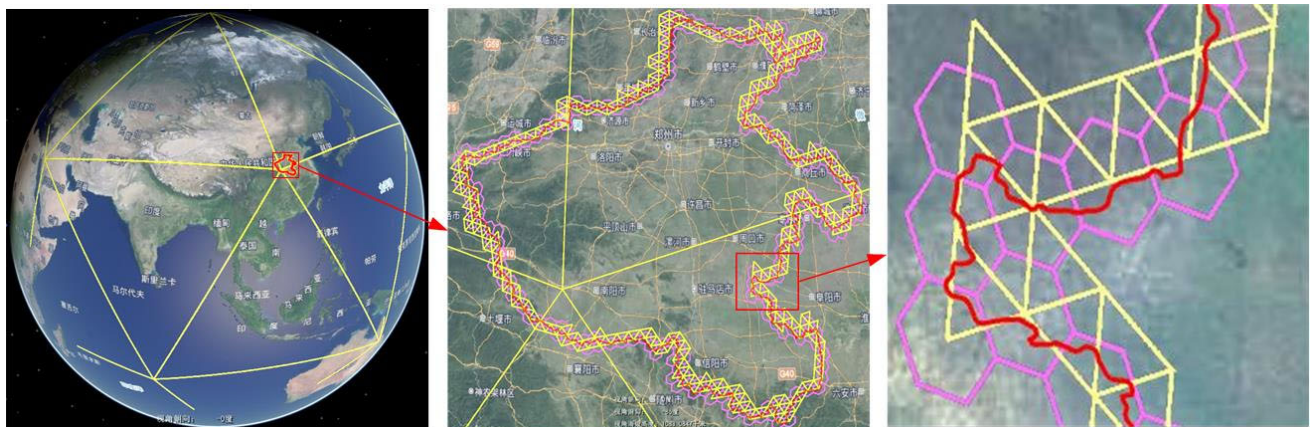


Figure 12. The ninth level *HtoTcode* discrete line.

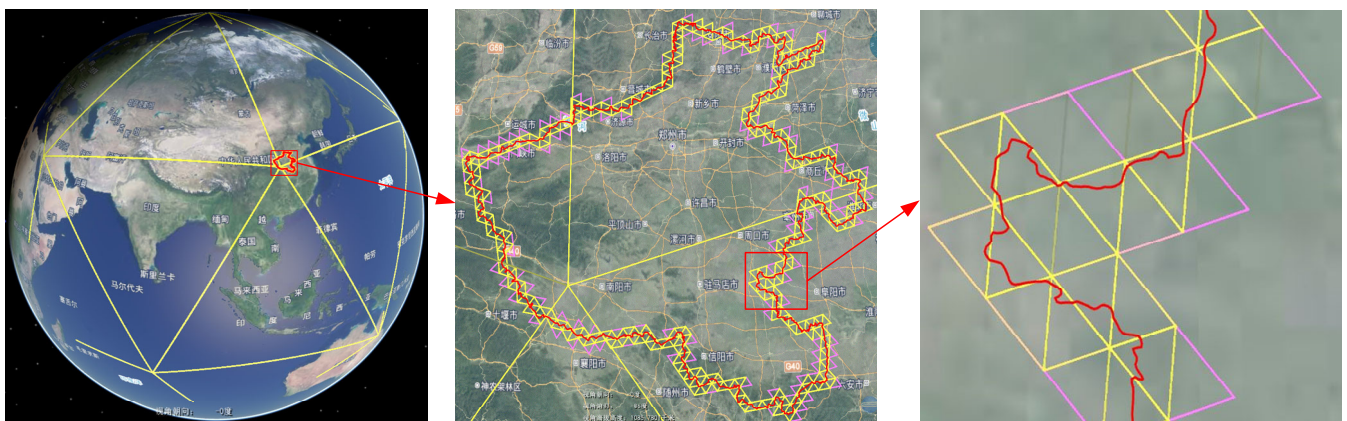


Figure 13. The ninth level *TtoDcode* discrete line.

4.3. Efficiency and Accuracy Comparison of Spatial Area GtoG Mapping

The GlobeLand30 land cover data of Henan Province in China is selected as the experimental data of vector surface (<https://www.webmap.cn/commres.do?method=globeIndex> (accessed on 6 March 2022)).

The method of vector area discretization is the local grid generation method for the DGGRID library [43]. Firstly, the intersection points of twenty basic diamonds and irregular polygons are calculated, and the irregular polygons are decomposed into several diamond surfaces to create “subregions”. According to the boundaries of the subregions, the traversal algorithm of “minimum diamond” is designed on the diamond structure to eliminate the grid cells unrelated to the target area. According to the grid code mapping rules of the spatial line and surface data, the *HtoTcode* and *TtoDcode* conversion efficiency of Henan globeland30 data is calculated. From the results (as shown in Tables 20 and 21), it can be seen that the mapping efficiency of *HtoTcode* and *TtoDcode* is 20,000 times and 300,000 times higher, respectively, than that of longitude–latitude middleware (local grid generation method for DGGRID library). This is because the *HtoTcode* and *TtoDcode* mapping rules avoid the complex process of determining the minimum circumscribed diamond of the local area used by the longitude–latitude method, and then determining one by one whether the triangle cell (diamond cell) in the minimum circumscribed diamond intersects with the local polygon.

Table 20. Comparison of efficiency and accuracy of *HtoTcode* and longitude–latitude expressing discrete area.

Level	<i>HtoTcode</i> Efficiency (ms)	Longitude–Latitude Efficiency (ms)	Efficiency Ratio	Number of Hexagonal Cells	Number of Triangle Cells	Relative Area Deviation
11	3	108,748	36,249	14,124	27,910	0.012305
12	17	418,611	24,624	55,507	110,340	0.006138
13	63	1,652,347	26,227	219,917	438,486	0.003066
14	297	6,632,788	22,332	875,486	1,748,282	0.001533

Table 21. Comparison of efficiency and accuracy of *TtoDcode* and longitude–latitude expressing discrete area.

Level	<i>TtoDcode</i> Efficiency (ms)	Longitude–Latitude Efficiency (ms)	Efficiency Ratio	Number of Triangle Cells	Number of Diamond Cells	Relative Area Deviation
10	0.09	27,313	303,478	7225	3677	0.017854
11	0.32	106,014	331,294	28,072	14,165	0.009191
12	1	419,343	419,343	110,719	55,632	0.004922
13	5	1,719,359	343,871	439,279	220,202	0.002561

Figures 14 and 15 show the different grid codes of the sixth, seventh and eighth level Henan Province GlobeLand30 data before and after *HtoTcode* and *TtoDcode* conversion. The codes are expressed in hexadecimal. In order to display all the codes in the unit, the code is represented in abbreviated form, where *N* represents the 11 zeros in the code.

Tables 20 and 21 show the area difference of Henan administrative regions expressed in different grids before and after *HtoTcode* and *TtoDcode* mapping, respectively. From Tables 20 and 21, it can be concluded that the area differences of the Henan administrative regions expressed in different grids decrease with the increase in levels. It can be seen from Figures 14 and 15 that, after *HtoTcode* and *TtoDcode* mapping, the land types of the land coverage data in the corresponding area may change. By comparing the conversion results of the sixth, seventh and eighth levels, it is concluded that the land type transformation can be avoided by improving the resolution, that is, increasing the level. Therefore, in practical applications, an appropriate grid level should be selected according to the area of the smallest land class, and the cell area of the selected level should be larger than the area of the smallest land class.

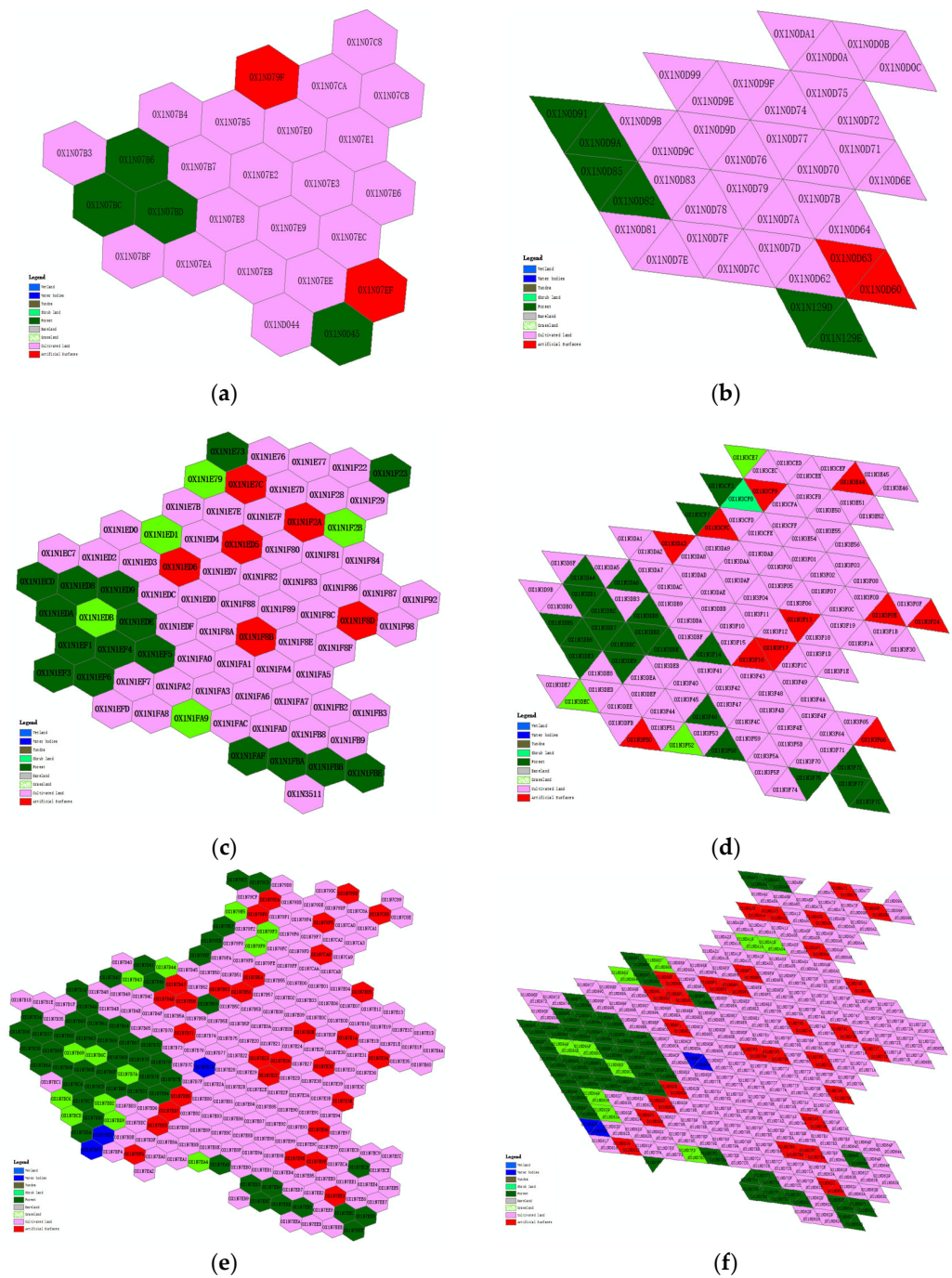


Figure 14. The different grid codes of the sixth, seventh and eighth levels of Globeland30 land coverage data in Henan Province before and after *HtoTcode* conversion. (a) The sixth level hexagonal grid code of land coverage data in Henan Province, (b) The sixth level *HtoTcode* triangle grid code of land coverage data in Henan Province, (c) The seventh level hexagonal grid code of land coverage data in Henan Province, (d) The seventh level *HtoTcode* triangle grid code of land coverage data in Henan Province, (e) The eighth level hexagonal grid code of land coverage data in Henan Province, (f) The eighth level *HtoTcode* triangle grid code of land coverage data in Henan Province.

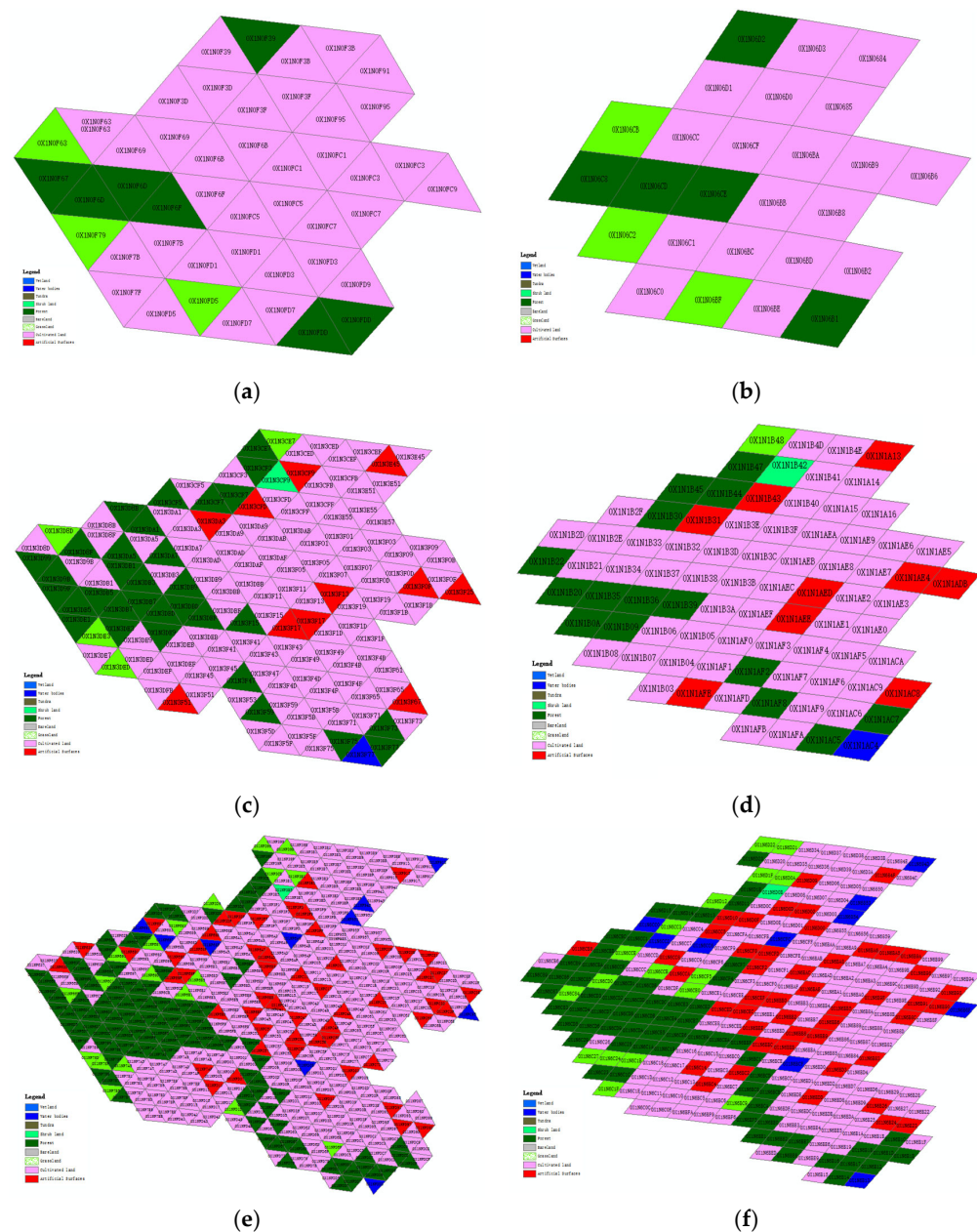


Figure 15. The different grid codes of the sixth, seventh and eighth levels of Globeland30 land coverage data in Henan Province before and after *TtoDcode* conversion. (a) The sixth level triangle grid code of land coverage data in Henan Province, (b) The sixth level *TtoDcode* diamond grid code of land coverage data in Henan Province, (c) The seventh level triangle grid code of land coverage data in Henan Province, (d) The seventh level *TtoDcode* diamond grid code of land coverage data in Henan Province, (e) The eighth level triangle grid code of land coverage data in Henan Province, (f) The eighth level *TtoDcode* diamond grid code of land coverage data in Henan Province.

5. Discussions

5.1. Uncertainty Generated by GtoG Mapping

The multi-source data of different grids after GtoG mapping will generate uncertainty (or error). The uncertainty includes three aspects: position, attribute and topological relationship. The followings will analyze the errors generated by GtoG mapping in relation to these three aspects.

5.1.1. Position Error

Point position error analysis is the basis for line and area position error analysis. The above experiment gives the maximum deviation of the center point position of different grid cells expressing point data before and after GtoG mapping. A point position error model can be established to quantitatively study the position deviation between the grid center point and the original vector point after the GtoG mapping of different grids, as well as the influence of grid levels on the position deviation in the future research.

The above experiment gives the maximum deviation of the center point of the corresponding cell on different grid discrete lines before and after GtoG mapping. From the results (Figures 16 and 17), it can be seen that the discrete lines of *HtoTcode* and *TtoDcode* cannot completely coincide with the discrete line of the longitude–latitude method. The discrete line of longitude–latitude conversion can better fit the vector line. Some cells of the *HtoTcode* and *TtoDcode* discrete lines do not pass through the vector line. However, it can be seen from Tables 18 and 19 that the maximum deviation value is smaller than the average diameter of grid cells at the corresponding level. The coincidence degree between the *TtoDcode* discrete line and the longitude–latitude discrete line is very high, and only two cells in the ninth layer do not coincide in Figure 17c. In future research, the overlap degree and the matching degree between the discrete lines of GtoG mapping and longitude–latitude conversion can be quantitatively analyzed.

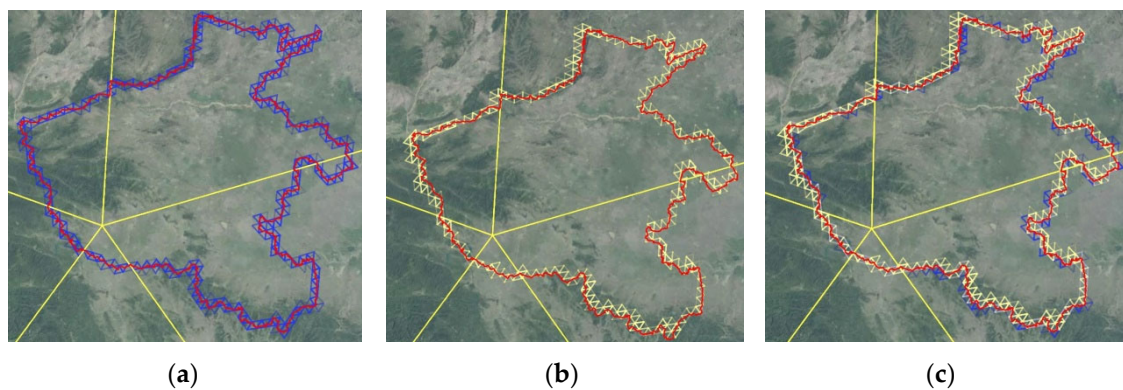


Figure 16. Comparison diagram of the ninth level triangle discrete lines of *HtoTcode* and longitude–latitude. (a) The ninth level triangle discrete lines of longitude–latitude; (b) the ninth level triangle discrete lines of longitude–latitude; (c) the ninth level discrete lines superposition of longitude–latitude and *HtoTcode*.

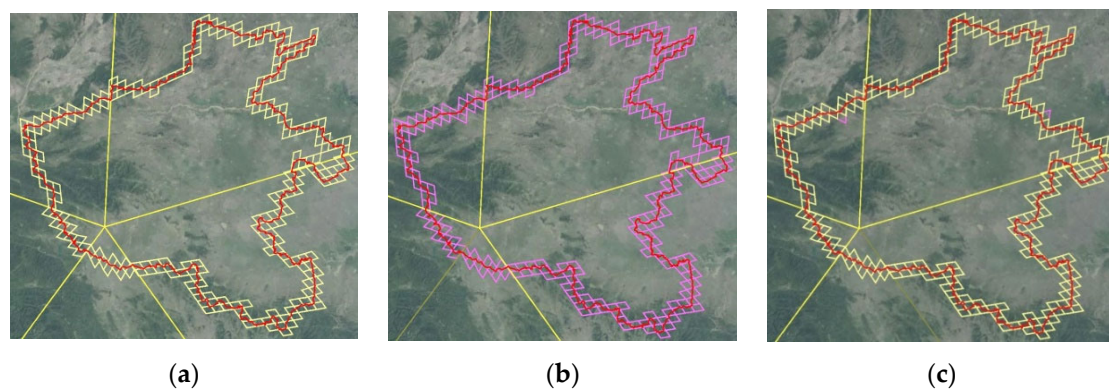


Figure 17. Comparison diagram of the ninth level diamond discrete lines of *TtoDcode* and longitude–latitude. (a) The ninth level diamond discrete lines of longitude–latitude; (b) the ninth level diamond discrete lines of *TtoDcode*; (c) the ninth level discrete lines superposition of longitude–latitude and *TtoDcode*.

The polygon position uncertainty is determined by the position error of the polygon boundary. GtoG mapping leads to a change in the polygon boundary position, which leads

to a change in polygon area, perimeter and barycenter. The deviation of the polygon area is given above, and the deviation of the polygon perimeter and barycenter caused by the change in polygon boundary position can also be quantitatively given in future research.

5.1.2. Attribute Uncertainty

Attribute uncertainty is one of the main indicators used to describe the quality of spatial data. Attribute data is used to further describe some qualitative and quantitative characteristics of spatial data. The attribute uncertainty of a discrete line is the length deviation of the line. In addition to the position offset of a discrete line, GtoG mapping may also lead to the length deviation of a discrete line. In future research, we can make up for the lack of accuracy evaluation in this paper and study the length and shape changes of discrete lines caused by the different mapping rules of GtoG, so as to obtain the fidelity of discrete lines. The influence of the scale factor on the length and shape of discrete lines of GtoG mapping can also be studied.

The position and attribute uncertainty synthesis of GtoG mapping of discrete surface data is analyzed. The difference between the internal region and the fuzzy boundary region results from the position uncertainty. In the internal region, the uncertainty mainly comes from attribute error. Within the fuzzy boundary, the spatial entity is affected by the position and attribute errors. The boundary uncertainty of planar objects is a typical example of the integration of location uncertainty and attribute uncertainty. For land cover data, GtoG mapping may lead to the change of land classes in the corresponding location, and it may also make small land classes disappear. From the results (Figure 14b,d,f and Figure 15a,c,e), it can be seen that the land type of the surface coverage data of *HtoTcode* and *TtoDcode* is different from that of the longitude and latitude method at some of the same positions of the corresponding layer.

5.1.3. Topological Relationship Uncertainty

The topological relationships of spatial data can be divided into six types: point to point, point to line, point to face, line to line, line to face and face to face. Because the types of topological relationship involving points (i.e., point to point, point to line and point to face) are relatively simple, the next step will mainly analyze the influence of GtoG mapping on the topological relationship of three spatial objects: line to line, line to face and face to face. The topological relationship between lines includes separation, connection and overlap. The topological relationship between lines and faces includes separation, connection, on the border, intersection, coverage and inclusion. The topological relationship between faces includes separation, tangency, intersection, inclusion, etc. After the GtoG mapping of multi-source data in different grids, the topological relationship of spatial data may change. For example, two polygons that are separated may intersect, and the two polygons that intersect may be separated. Therefore, it is necessary to study the uncertainty of the topological relationships of spatial data generated by GtoG mapping to make up for the deficiency of the accuracy evaluation above.

5.2. Different Aperture Extensions of GtoG Mapping

The method in this paper is a mapping rule library of aperture 4 hexagonal, triangular and diamond grids on the icosahedron, which can be extended to other apertures through modification. For example, on the basis of the hierarchical transformation method proposed by Amiri [44], the multi-level representation of the grid and the mutual mapping between different grids can be realized through decomposition and aggregation. In Figure 18b, the hexagon is decomposed into triangles, and in Figure 19b,c, the triangles are aggregated into hexagons. Selecting different starting points will result in different hexagonal cells, which can yield hexagonal grids with different partition structures, as in Figure 20, including aperture 4 hexagonal grids, aperture 3 hexagonal grids and aperture 7 hexagonal grids. A diamond can be decomposed into two triangles, and two triangles can be aggregated into a diamond. With different decomposition and aggregation methods, triangle, diamond

and hexagon cells have different corresponding relations, and the corresponding cells are obtained through the transformation of different grid coordinate systems. For the transformation of spatial point data, it is only necessary to obtain the corresponding cell according to the transformation of the different grid coordinate systems. For spatial line and area data, in order to prevent breaks and loopholes in spatial line data and area data, in addition to obtaining the one-to-one correspondence between different grid cells, it is also necessary to determine the connectivity of the converted cells according to the adjacent relationship of the cells. For example, in Figure 21, the hexagonal cell h_a corresponds to the triangular cell t_a , and the hexagonal cell h_b corresponds to the triangular cell t_b . For the conversion of spatial line and area data, on the basis of finding the triangle cell corresponding to hexagon cell h_a and h_b , it is also necessary to determine the connection cell t_c between triangle cell t_a and t_b .

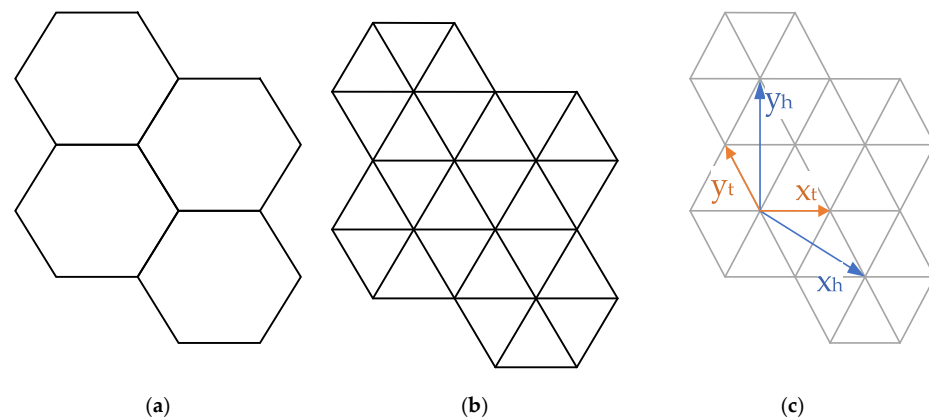


Figure 18. Hexagonal grid to triangular grid. (a) Hexagonal grid cells; (b) Triangles obtained by decomposing hexagons; (c) Coordinate systems for triangular and hexagonal grids, x_t and y_t are triangular grid coordinate systems, x_h and y_h are hexagonal coordinate systems.

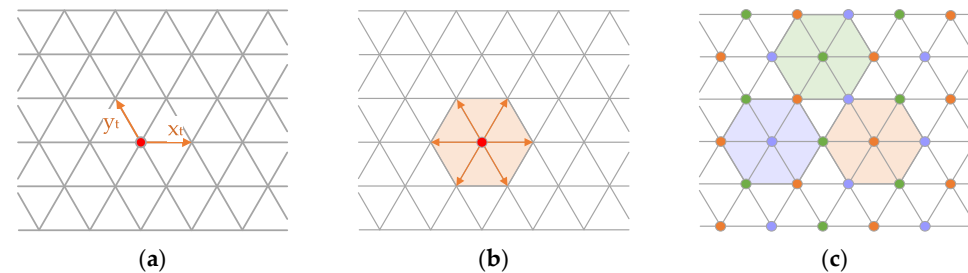


Figure 19. Triangular grid to hexagonal grid. (a) Triangular grid, its origin and coordinate system; (b) Vectors chosen to aggregate a hexagonal cell in triangular grid; (c) There are three possibilities to choose the origin of the aggregation conversion illustrated by salmon, green and blue points.

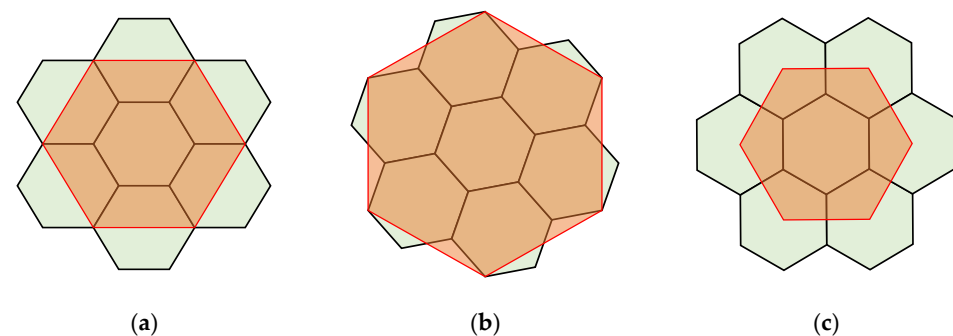


Figure 20. Hexagonal grids with different subdivision structures. (a) Aperture 4 hexagonal grid; (b) Aperture 7 hexagonal grid; (c) Aperture 3 hexagonal grid.

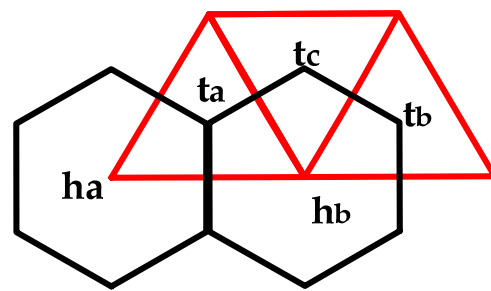


Figure 21. Correspondence between hexagonal grids and triangular grids.

5.3. Different Polyhedron Extensions of GtoG Mapping

It can be seen from Table 1 that the polyhedrons selected for the global discrete grid model include hexahedrons, diamonds, dodecahedrons, octahedrons, icosahedrons and triacontahedrons. The most commonly used polyhedrons include hexahedrons, octahedrons and icosahedrons. At present, only quadrilateral grids are established on the hexahedron, and there is no need to perform isomorphic multi-type grid mapping. The mapping rule library based on icosahedral multi-type grids (triangular, rhombic and hexagonal grids) constructed in this paper can be extended to the octahedron by modifying the mapping rules of the polyhedral vertex and boundary cells.

For the transformation of spatial point data, the mapping rules of the icosahedron and octahedron are the same. For the transformation of spatial line data and surface data, a problem arises involving the cells at the vertices and boundaries of the polyhedron, because the cells at the vertices of the octahedron are quadrilateral, which is different from the pentagons at the vertices of the icosahedron. The proximity of the cells has changed, and the planar expansion of the octahedron is four diamonds (Figure 22a), so that the faces crossed by the cells E_1 and E_2 will be reduced compared to the icosahedron. At the same time, the hexagonal cells on the octahedron are classified according to the above location classification method of hexagonal cells on the basic diamond (Figure 22b). Comparing Figures 8 and 22b, it can be concluded that the mapping rule bases of the octahedron and icosahedron are identical for A type cells. For the B type cells, it is only necessary to modify the cross face number of the B type cells, such as, for example, changing $D + 4$ in Table 7 to $D + 1$, and changing $D - 5$ in Table 7 to $D - 2$. For C and E type cells, the cross face number of mapping cells needs to be changed and the corresponding connection cells need to be reduced. For example, for Table 9, the HD_{n+1} column in Table 9 needs to be deleted and Table 9 needs to be changed to Table 22.

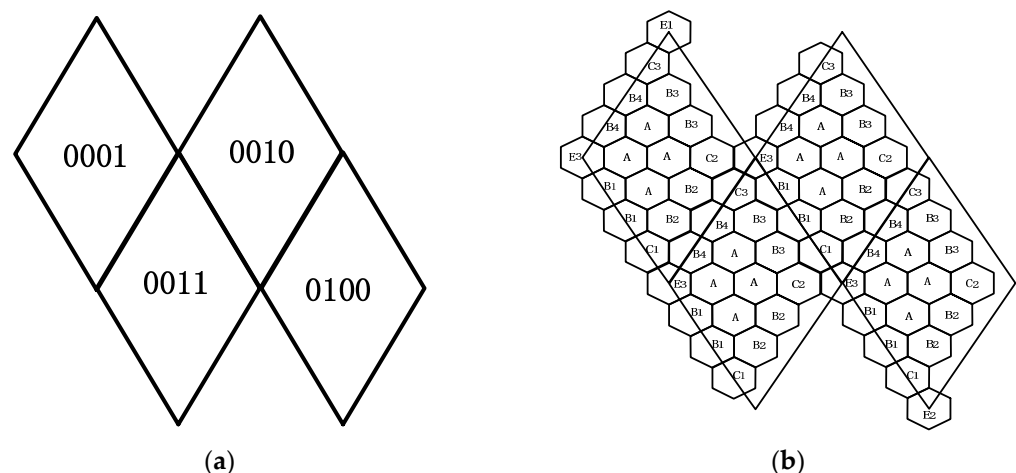


Figure 22. Octahedral planar expansion and hexagonal cell classification. (a) Basic diamond code of octahedron; (b) Octahedral hexagonal cell type.

Table 22. *HtoTcode* rule of the $(n - 1)$ th hexagon cell C_1 and the n th hexagon cell E_2 .

HD_{n-1}	HD_n	Triangle Cells Selected
$D = 1$	$D + 1$	$(D, (x_{n-1}, y_{n-1}), 0)$ and $(D + 1, (x_{n+1}, y_{n+1}), 0)$

6. Conclusions

In this paper, the GtoG direct mapping method of multi-type global discrete grid code based on a spatial filling curve is proposed, given the background of the high complexity of longitude–latitude transformation method in DGGS framework. The main contributions of this paper include:

1. Formulate GtoG mapping rule sets for multi-type spherical grids. This paper constructs the mutual conversion rule bases of different grids expressing spatial point, line and surface data, then extends the method to the spherical icosahedral grid framework model, and designs the GtoG coding mapping rule tables of three different regions of the basic in-plane cells, boundary cells and vertex cells.
2. Test GtoG mapping rule efficiency and error. Compared with the longitude–latitude conversion method, the GtoG average conversion efficiency of spatial point data, line data and area data are increased by 2–4 orders of magnitude. Since the center points of different grid cells do not overlap, there is a certain deviation in the spatial point and line-surface data before and after GtoG mapping, but with the increase of the level, the deviation will become smaller and smaller. Therefore, in practical application, the appropriate hierarchical grid can be selected for conversion according to the accuracy requirements.

GtoG mapping realizes the direct coding mapping of triangular, diamond and hexagonal grids expressing spatial point, line and surface data, which reduces the complexity of mutual conversion between multi-type DGGS. The method proposed in this paper lays a foundation for the efficient integration of multi-source spherical grid data, which is beneficial in promoting the development of DGGS interoperability.

Author Contributions: Conceptualization, Yalu Li and Xuesheng Zhao; methodology, Yalu Li; software, Yalu Li, Fuli Luo and Guangsong Wang; validation, Yalu Li, Xuesheng Zhao and Wenbin Sun; formal analysis, Yalu Li; investigation, Yalu Li; resources, Yalu Li; data curation, Zheng Wang and Yuanzheng Duan; writing—original draft preparation, Yalu Li; writing—review and editing, Yalu Li and Xuesheng Zhao; visualization, Yalu Li and Guangsong Wang; supervision, Xuesheng Zhao and Wenbin Sun; project administration, Yalu Li and Guangsong Wang; funding acquisition, Xuesheng Zhao. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [the National Key Research and Development Program of China] grant number [2018YFB0505301], [the general program of National Natural Science Foundation of China] grant number [42271435] and [the key program of National Natural Science Foundation of China] grant number [41930650].

Data Availability Statement: The Henan Province administrative boundary data in China are selected as the experimental data of the vector line (from Resources and Environment Science and Data Center, <https://www.resdc.cn/data.aspx?DATAID=200>, (accessed on 2 March 2022)). The GlobeLand30 land cover data of Henan Province in China is selected as the experimental data of vector surface (<https://www.webmap.cn/commres.do?method=globeIndex>, (accessed on 6 March 2022)).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Goodchild, M.F.; Shiren, Y. A Hierarchical Spatial Data Structure for Global Geographic Information Systems. *CVGIP: Graph. Model. Image Process.* **1992**, *54*, 31–44. [\[CrossRef\]](#)
- Chen, J.; Xiang, L.; Gong, J. Virtual globe-based integration and sharing service method of GeoSpatial Information. *Sci. China Earth Sci.* **2013**, *56*, 1780–1790. [\[CrossRef\]](#)
- Goodchild, M.F.; Guo, H.; Annoni, A.; Bian, L.; de Bie, K.; Campbell, F.; Craglia, M.; Ehlers, M.; van Genderen, J.; Jackson, D.; et al. Next-Generation Digital Earth. *Proc. Natl. Acad. Sci. USA* **2012**, *109*, 11088–11094. [\[CrossRef\]](#)
- Li, M.; Stefanakis, E. Geospatial Operations of Discrete Global Grid Systems—A Comparison with Traditional GIS. *J. Geovis. Spat. Anal.* **2020**, *4*, 26. [\[CrossRef\]](#)
- Lewis, A.; Oliver, S.; Lymburner, L.; Evans, B.; Wyborn, L.; Mueller, N.; Raevksi, G.; Hooke, J.; Woodcock, R.; Sixsmith, J.; et al. The Australian Geoscience Data Cube—Foundations and Lessons Learned. *Remote Sens. Environ.* **2017**, *202*, 276–292. [\[CrossRef\]](#)
- Kim, T.; Lee, J.; Kim, K.-S.; Matono, A.; Li, K.-J. Utilizing Extended Geocodes for Handling Massive Three-Dimensional Point Cloud Data. *World Wide Web* **2021**, *24*, 1321–1344. [\[CrossRef\]](#)
- Li, M.; McGrath, H.; Stefanakis, E. Integration of Heterogeneous Terrain Data into Discrete Global Grid Systems. *Cartogr. Geogr. Inf. Sci.* **2021**, *48*, 546–564. [\[CrossRef\]](#)
- Jendryke, M.; McClure, S.C. Spatial Prediction of Sparse Events Using a Discrete Global Grid System; a Case Study of Hate Crimes in the USA. *Int. J. Digit. Earth* **2021**, *14*, 789–805. [\[CrossRef\]](#)
- Sherlock, M.J.; Hasan, M.; Samavati, F.F. Interactive Data Styling and Multifocal Visualization for a Multigrid Web-Based Digital Earth. *Int. J. Digit. Earth* **2021**, *14*, 288–310. [\[CrossRef\]](#)
- Bowater, D.; Stefanakis, E. An Open-Source Web Service for Creating Quadrilateral Grids Based on the RHEALPix Discrete Global Grid System. *Int. J. Digit. Earth* **2020**, *13*, 1055–1071. [\[CrossRef\]](#)
- Zhou, J.; Ben, J.; Wang, R.; Zheng, M.; Yao, X.; Du, L. A Novel Method of Determining the Optimal Polyhedral Orientation for Discrete Global Grid Systems Applicable to Regional-Scale Areas of Interest. *Int. J. Digit. Earth* **2020**, *13*, 1553–1569. [\[CrossRef\]](#)
- Jendryke, M.; McClure, S.C. Mapping Crime—Hate Crimes and Hate Groups in the USA: A Spatial Analysis with Gridded Data. *Appl. Geogr.* **2019**, *111*, 102072. [\[CrossRef\]](#)
- Amiri, A.M.; Alderson, T.; Samavati, F. A Survey of Digital Earth. *Comput. Graph.* **2015**, *53*, 95–117. [\[CrossRef\]](#)
- Lü, X.; Cheng, C.; Gong, J.; Guan, L. Review of Data Storage and Management Technologies for Massive Remote Sensing Data. *Sci. China Technol. Sci.* **2011**, *54*, 3220–3232. [\[CrossRef\]](#)
- Zhao, X.; Bai, J.; Wang, Z. An Adaptive Visualized Model of the Global Terrain Based on QTM. *Acta Geod. Cartogr. Sin.* **2007**, *36*, 316–320.
- Golay, M.J.E. Hexagonal Parallel Pattern Transformations. *IEEE Trans. Comput.* **1969**, *100*, 733–740. [\[CrossRef\]](#)
- Mersereau, R.M. The Processing of Hexagonally Sampled Two-Dimensional Signals. *Proc. IEEE* **1979**, *67*, 930–949. [\[CrossRef\]](#)
- Birch, C.P.D.; Oom, S.P.; Beecham, J.A. Rectangular and Hexagonal Grids Used for Observation, Experiment and Simulation in Ecology. *Ecol. Model.* **2007**, *206*, 347–359. [\[CrossRef\]](#)
- Dutton, G. *A Hierarchical Coordinate System for Geoprocessing and Cartography*; Lecture notes in earth sciences; Springer: Berlin/Heidelberg, Germany; New York, NY, USA, 1999; ISBN 978-3-540-64980-9.
- White, D. Global Grids From Recursive Diamond Subdivisions of the Surface of an Octahedron or Icosahedron. In *Monitoring Ecological Condition in the Western United States*; Sandhu, S.S., Melzian, B.D., Long, E.R., Whitford, W.G., Walton, B.T., Eds.; Springer: Dordrecht, The Netherlands, 2000; pp. 93–103. ISBN 978-94-010-5870-4.
- Bartholdi, J.J.; Goldsman, P. Continuous Indexing of Hierarchical Subdivisions of the Globe. *Int. J. Geogr. Inf. Sci.* **2001**, *15*, 489–522. [\[CrossRef\]](#)
- Gorski, K.M.; Hivon, E.; Banday, A.J.; Wandelt, B.D.; Hansen, F.K.; Reinecke, M.; Bartelmann, M. HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere. *ApJ* **2005**, *622*, 759–771. [\[CrossRef\]](#)
- Sahr, K. Location Coding on Icosahedral Aperture 3 Hexagon Discrete Global Grids. *Comput. Environ. Urban Syst.* **2008**, *32*, 174–187. [\[CrossRef\]](#)
- Zhao, X.; Cui, M.; Li, A.; Zhang, M. An Adjacent Searching Algorithm of Degenerate Quadtree Grid on Spheriactal Facet. *Geomat. Inf. Sci. Wuhan Univ.* **2009**, *34*, 479–482.
- Vince, A.; Zheng, X. Arithmetic and Fourier Transform for the PYXIS Multi-Resolution Digital Earth Model. *Int. J. Digit. Earth* **2009**, *2*, 59–79. [\[CrossRef\]](#)
- Amiri, A.M.; Bhojani, F.; Samavati, F. One-to-Two Digital Earth. In *Advances in Visual Computing*; Bebis, G., Boyle, R., Parvin, B., Koracin, D., Li, B., Porikli, F., Zordan, V., Klosowski, J., Coquillart, S., Luo, X., et al., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2013; Volume 8034, pp. 681–692. ISBN 978-3-642-41938-6.
- Gibb, R.G. The RHEALPix Discrete Global Grid System. *IOP Conf. Ser. Earth Environ. Sci.* **2016**, *34*, 012012. [\[CrossRef\]](#)
- Veach, E.; Rosenstock, J.; Engle, E.; Snedegar, R.; Basch, J.; Manshreck, T. S2 Geometry Library: Computational Geometry and Spatial Indexing on the Sphere. 2017. Available online: <https://s2geometry.io> (accessed on 6 May 2022).
- Uber Technologies Inc. H3: Hexagonal Hierarchical Geospatial Indexing System. 2018. Available online: <https://github.com/uber/h3/releases/tag/v4.0.0> (accessed on 6 May 2022).
- Lin, B.; Zhou, L.; Xu, D.; Zhu, A.-X.; Lu, G. A Discrete Global Grid System for Earth System Modeling. *Int. J. Geogr. Inf. Sci.* **2018**, *32*, 711–737. [\[CrossRef\]](#)

31. Ben, J.; Li, Y.; Zhou, C.; Wang, R.; Du, L. Algebraic Encoding Scheme for Aperture 3 Hexagonal Discrete Global Grid System. *Sci. China Earth Sci.* **2018**, *61*, 215–227. [[CrossRef](#)]
32. Sahr, K. Central Place Indexing: Hierarchical Linear Indexing Systems for Mixed-Aperture Hexagonal Discrete Global Grid Systems. *Cartogr. Int. J. Geogr. Inf. Geovis.* **2019**, *54*, 16–29. [[CrossRef](#)]
33. Hall, J.; Wecker, L.; Ulmer, B.; Samavati, F. Disdyakis Triacotahedron DGGS. *IJGI* **2020**, *9*, 315. [[CrossRef](#)]
34. Wang, R.; Ben, J.; Zhou, J.B.; Zheng, M.Y. A Generic Encoding and Operation Scheme for Mixed Aperture Three and Four Hexagonal Discrete Global Grid Systems. *Int. J. Geogr. Inf. Sci.* **2021**, *35*, 513–555. [[CrossRef](#)]
35. Open Geospatial Consortium. Discrete Global Grid Systems—Part 1 Core Reference System and Operations and Equal Area Earth Reference System. 2021. Available online: <https://docs.ogc.org/as/20-040r3/20-040r3.html#toc11> (accessed on 26 June 2022).
36. Open Geospatial Consortium. OGC Testbed-16: DGGS and DGGS API Engineering Report. 2021. Available online: https://docs.ogc.org/per/20-039r2.html#_development_tasks_relevant_to_other_DGGS_libraries (accessed on 1 July 2022).
37. Amiri, A.M.; Samavati, F.; Peterson, P. Categorization and Conversions for Indexing Methods of Discrete Global Grid Systems. *IJGI* **2015**, *4*, 320–336. [[CrossRef](#)]
38. Du, L.; Ben, J.; Ma, Q.; Wang, R.; Li, Z. An Algorithm for Generating Discrete Line Transformation of Planar Triangular Grid Based on Weak Duality. *Geomat. Inf. Sci. Wuhan Univ.* **2020**, *45*, 105–110. [[CrossRef](#)]
39. Chen, Y.; Cao, Z.; Wang, J.; Shi, Y.; Qin, Z. Encoding Conversion Algorithm of Quaternary Triangular Mesh. *IJGI* **2021**, *11*, 33. [[CrossRef](#)]
40. Zhao, Z. Research on Model-oriented Spherical Rhombus Discrete Grids. Master's Thesis, Nanjing Normal University, Nanjing, China, 2015.
41. Liu, K. A Universal Data Model of Discrete Global Grid System Based on Fiber Bundle Theory. Master's Thesis, Nanjing Normal University, Nanjing, China, 2019.
42. Xu, H. Research of High-Dimensional Space Query Algorithm Based on Space-Filling Curves. Ph.D. Dissertation, Harbin University of Science and Technology, Harbin, China, 2011.
43. Sahr, K. DGGRID, Version 7.5. 2022. Available online: <https://discreteglobal.wpengine.com/> (accessed on 6 May 2022).
44. Amiri, A.M.; Harrison, E.; Samavati, F. Hierarchical Grid Conversion. *Comput. Aided Des.* **2016**, *79*, 12–26. [[CrossRef](#)]