



正二十面体四孔六边形格网系统编码运算

王蕊¹ 贾进¹ 杜灵瑀¹ 周建彬¹ 李祝鑫²

1 信息工程大学地理空间信息学院,河南 郑州,450001

2 61287部队,四川 成都,610000

摘要:全球离散格网系统是数字化的多分辨率地球参考模型,在结构上支持多源位置相关信息的融合处理。基于正二十面体剖分的六边形全球离散格网系统具有较好的几何属性,相关研究已引起学术界的广泛关注,如何建立封闭球面上的六边形格网系统编码运算方案是当前的研究难点。研究表明,基于正多面体剖分的全球离散格网系统与正多面体格网系统拓扑等价,两者的编码运算结果也完全相同。根据这一原理,结合四孔六边形格网系统在正二十面体表面的分布特点,基于六边形格点四叉树定义顶点瓦片与面瓦片结构,提出了正二十面体四孔六边形格网系统编码运算方案。该方案通过高效编码运算实现了格网单元跨面操作,克服了现有成果需借助低效浮点数运算实现相同操作的缺陷。对比实验表明,该方案的跨面邻近单元搜索效率约是六边形四元平衡结构方案的19.6倍。

关键词:全球离散格网系统;正二十面体;六边形;编码运算

中图分类号:P208

文献标志码:A

全球离散格网系统(discrete global grid system, DGGS)是一类以地球作为组织框架的空间参考系统。相较于传统空间数据模型,其更适合解决大尺度的问题^[1],且单元运算可完全借助编码实现,有助于提高数据处理效率。六边形具有拓扑关系一致、采样效率高等理想几何特性,有助于实现空间分析算法^[2-3]及数据的可视化表达^[4],但其不具备叠合一致性,无法直接建立多分辨率格网层次关系。Gibson等^[5]提出一种多维数据结构,其二维特例可通过广义平衡三进制(generalized balanced ternary, GBT)运算实现不同层次六边形单元索引。然而,GBT采用高分辨率单元聚合生成低分辨率单元,在球面上会导致格网破碎或重叠。

基于多面体剖分建立六边形全球离散格网系统仍被认为是目前最有效的方法。正二十面体面数最多,投影到球面后变形最小^[6],因而被广泛使用。Sahr^[7]认为,正二十面体等积三孔六边形格网系统(icosahedral snyder equal area aperture 3 hexagonal grid, ISEA3H)具有较好的几何属性,他借助与GBT类似的码元分布,建立改进

广义平衡三进制(modified generalized balanced ternary, MGBT)编码方案,但其编码不唯一,仅适用于表达矢量数据。之后,Sahr^[8]又将MGBT分解为两种子单元模块的组合,建立三孔六边形树(aperture 3 hexagon tree, A3HT)唯一编码方案,并扩展至正二十面体。Vince^[9]建立了正二十面体三孔六边形格网的进制索引,并实现了傅里叶变换。童晓冲等^[10]建立了四孔六边形格网系统的四元平衡结构(hexagonal quad balanced structure, HQBS),并实现正二十面体格网编码运算。

数学家已从理论上证明,封闭球面拓扑等价于封闭正二十面体表面,全球离散格网系统编码运算拓扑等价于正二十面体格网系统编码运算。然而无限平面格网的编码运算规则不适用于正二十面体封闭表面,如何实现正二十面体上的高效率编码运算是目前的研究难点。A3HT通过指定正二十面体上首层单元的递归剖分规则建立正二十面体格网系统,该方案不是直接利用编码运算实现单元操作,而是先借助二维整数坐标系计算结果,再将之转换为编码^[8]。HQBS采用

收稿日期:2018-12-16

项目资助:国家自然科学基金(41671410);国家重点研发计划(2018YFB0505301)。

第一作者:王蕊,博士生,主要从事空间数据模型方面的研究。wr_paper@126.com

通讯作者:贾进,博士,教授。benj@reis.ac.cn

分面编码到全局编码的映射解决上述问题,但只能保证编码运算在单个三角面上连续,跨面操作仍需借助笛卡尔坐标运算实现。上述两种方案的计算效率均不高,理论上也不严密。Vince^[9]通过对三孔六边形格网层次结构的裁剪拼接建立正二十面体格网模型,由于区分奇(偶)层编码,运算规律较复杂,且相邻层次单元方向会交替变化,不利于格网层次分析。

本文结合四孔六边形格网系统在正二十面体表面的分布特点,基于六边形格点四叉树定义顶点瓦片与面瓦片结构,提出正二十面体四孔六边形格网系统编码方案,归纳编码运算规律,通过对比实验验证方案的有效性与优越性。

1 平面四孔六边形格网编码运算

四孔六边形格网单元剖分过程如图1所示,为保证编码的唯一性,本文的格网剖分选择图1阴影部分所示的子单元模块,它们关于父单元对称,几何属性更优。

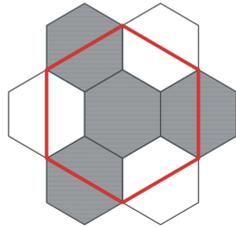


图1 四孔六边形格网子单元模块
Fig.1 Child Block of Aperture 4 Hexagonal Grids

1.1 编码方案

为便于讨论,定义平面格网系统单元中心为格点(lattice point)等效替代单元。首层格网单元可自定义。令 L_n 表示第 n 层格点集合($n \geq 1$),根据编码运算需要,本文定义的 L_1 在复数平面上的位置如图2所示,图中 R 轴和 I 轴分别表示复数坐标系的实轴和虚轴。根据图1生成的 L_2 与 L_1 层次关系见图3,为方便表示,图3只选取 L_1 中心7个格点为例,每个格点在 L_2 各生成4个子格点,中心子格点与其父单元格点重合。按照上述思路,可依次生成 $L_n(n > 2)$ 。

根据文献[11],在复数平面上,令 $\theta = \frac{1}{2} + \frac{\sqrt{3}}{2}i$, L_1 用以下5个集合 $D_j'(1 \leq j \leq 5)$ 表示:
 $D_1' = \{0, \theta^e \mid 1 \leq e \leq 6\}$, $D_2' = \{2\theta^e \mid 1 \leq e \leq 6\}$,
 $D_3' = \{\theta^e + \theta^{(e+1)\%6} \mid 1 \leq e \leq 6\}$, $D_4' = \{2\alpha \mid \alpha \in D_3'\}$,

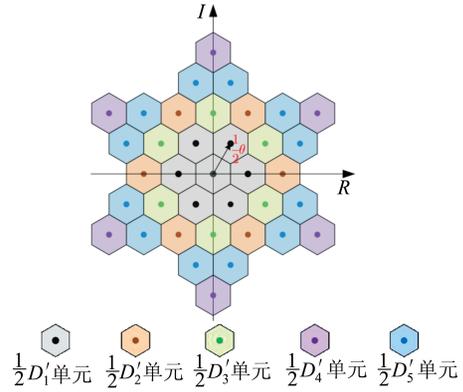


图2 L_1 格点位置示意图
Fig.2 Diagram of L_1 Location

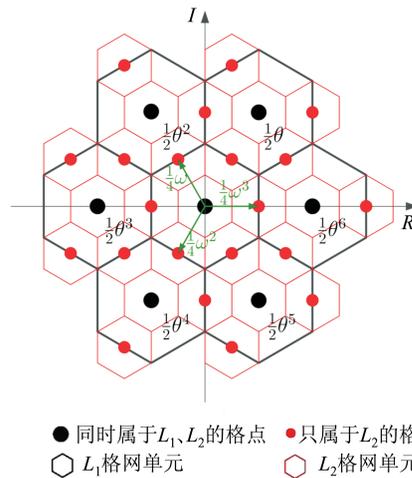


图3 L_1 与 L_2 格点层次关系示意图
Fig.3 Diagram of Relationship Between L_1 and L_2

$D_5' = \{2\theta^{e_1} + \theta^{e_2} \mid |e_1 - e_2| = 1, 1 \leq e_1, e_2 \leq 6 \text{ 或 } e_1 = 1, e_2 = 6 \text{ 或 } e_2 = 1, e_1 = 6\}$ 。其中, $\%$ 表示取余;
 $e_1, e_2 \in Z$ 。令 $\omega = -\frac{1}{2} + \frac{\sqrt{3}}{2}i$, $D = \{0, \omega, \omega^1, \omega^2, \omega^3\}$,经过严格证明,四孔六边形格网系统的格点集合可唯一表示为:

$$\begin{cases} L_n = L_1 + \sum_{j=2}^n \frac{1}{2^j} D \\ L_1 = \frac{1}{2} D' \end{cases} \quad (1)$$

式中, $D' = D_1' \cup D_2' \cup D_3' \cup D_4' \cup D_5'$ 。在 L_1 中任选一个元素,对于后 $n-1$ 层(即 $2 \leq j \leq n$),每层在集合 D 中任选一个元素参与计算, n 层相加结果为 L_n 中一个元素。按上述方法,所有元素组合的计算结果集合为 n 层格点集合 L_n 。如图3所示, L_1 中心7个格点集合可表示为 $\{0, \frac{1}{2}\theta^e \mid 1 \leq e \leq 6\}$,假设从中任意取一格点,坐标用 α 表示,则其在第二层的4个子格点集合可

表示为: $\alpha + \frac{1}{2^2}D = \alpha + \frac{1}{4} \{0, \omega, \omega^2, \omega^3\}$ 。

式(1)表明,四孔六边形格点系统本质上是一种定位计数系统,格点与实数域上的二进制、十进制等数字的本质完全相同,是复数域上一种特定形式的“数”。该定位计数系统的进制为 2, 数字位集合为 D' 和 D 。对于任意一格点复数坐标 $x \in L_n$, 可用数字位和进制的加乘表示, 即:

$$x = \frac{1}{2}d_1 + \frac{1}{4}d_2 + \dots + \frac{1}{2^n}d_n \quad (2)$$

式中, $d_1 \in D'$; $d_2, d_3 \dots d_n \in D$ 。这与十进制数类似, 因此, 省略幂指数的底, 式(2)又可记为:

$$x = d_1 d_2 \dots d_n \quad (3)$$

式中, 复数数字位 $d_i (1 \leq i \leq n)$ 可用相应的整数替换, 由此得到该格网系统的整数编码集合。对数字位 d_i 做如下替换, 当 $d_i \in D_1'$ 时, 将 d_i 替换为坐标的幂次 e , 即 $d_i \rightarrow e (1 \leq e \leq 6)$; 若 $d_i = 0$, 则 $d_i \rightarrow 0$; 当 $d_i \in D_2'$ 时, $d_i \rightarrow 100e (1 \leq e \leq 6)$, 即 100 与 e 相乘; 当 $d_i \in D_3'$ 时, $d_i \rightarrow 10(e+1) \% 6 (1 \leq e \leq 6)$; 当 $d_i \in D_4'$ 时, $d_i \rightarrow 100d_1'$, 其中 d_1' 为 α 替换后码元; 当 $d_i \in D_5'$,

$d_i \rightarrow 100e_1 + e_2$ 。对于式(3)中 $d_j (2 \leq j \leq n)$, 若 $d_j = \omega^e$, 则 $d_j \rightarrow e (1 \leq e \leq 3)$; 若 $d_j = 0$, 则 $d_j \rightarrow 0$ 。

根据上述替换规则, 数字位集合 D' 被替换为整数码元集合 $E' = \{0, 1, 2, 3, 4, 5, 6, 10, 20, 30, 40, 50, 60, 100, 200, 300, 400, 500, 600, 106, 601, 201, 102, 302, 203, 403, 304, 504, 405, 604, 506, 1000, 2000, 3000, 4000, 5000, 6000\}$, 数字位集合 D 替换为码元集合 $E = \{0, 1, 2, 3\}$, 由此得到任一格点 x 的唯一编码标识:

$$x = e_1 e_2 \dots e_n \quad (4)$$

式中, 码元 $e_1 \in E', e_j \in E (2 \leq j \leq n)$ 。为方便区分, 实际表示时在 e_1 后加逗号分隔, 例如编码 20、213 为第四层编码。该编码方案是一种典型的四叉树结构, 首层每个格点都分别生长为一棵独立的四叉树, 因此将这一结构命名为六边形格点四叉树 (hexagonal lattice quad tree, HLQT)。

图 4 为首层格点 D_1' 在 L_3 中的子格点编码示意图, 由于 D_1' 对应首层码元均为 1 位数字, 图中不用逗号分隔。

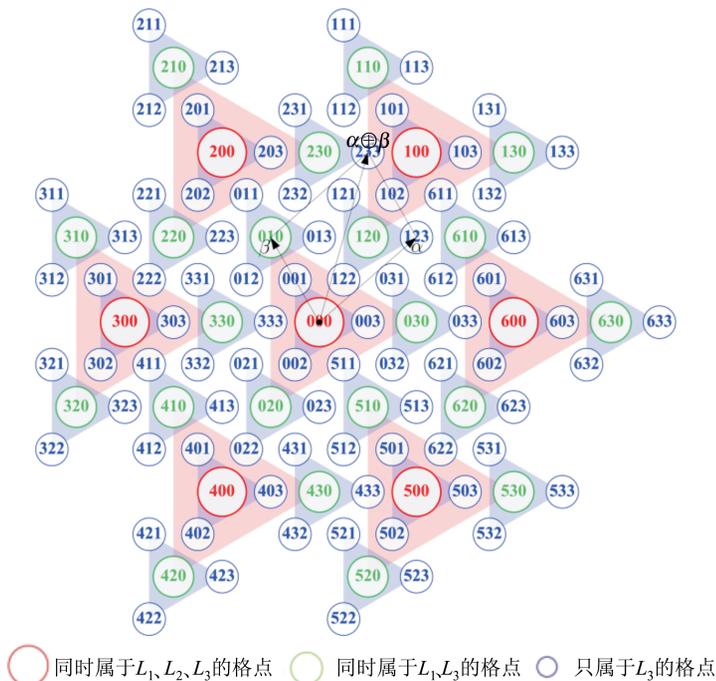


图 4 格网第三层编码及运算示意图

Fig.4 Diagram of Codes in the Third Level and Example of Code Arithmetic

1.2 编码运算

编码运算采用更适合计算机处理的一维码元序列代替传统二维实数坐标, 从而提高计算效率。

设 H_n 为 HLQT 第 n 层编码集合 ($n \geq 1$), 编码

$\alpha, \beta \in H_n$, 符号“ \oplus ”表示编码加法运算。HLQT 编码加法运算满足向量相加的平行四边形法则。如图 4 所示, $\alpha \oplus \beta = 123 \oplus 010 = 233$ 。经过归纳、验证, 无限二维平面上的 HLQT 编码运算满足交换群的性质^[11]。

设 n 层编码 $\alpha = a_1 a_2 \cdots a_n, \beta = b_1 b_2 \cdots b_n$, 编码加法计算方式与十进制加法类似, 从右侧末位码元 (即 a_n 或 b_n) 开始, 向左侧首位码元 (即 a_1 或 b_1) 逐位计算, 但进位机制略有差异: HLQT 加法运算当前位计算结果可能在其左侧多个加法位上产生进位码元。编码加法运算可分解为不同加法位对应编码相加之和:

$$\alpha \oplus \beta = a_1 a_2 \cdots a_n \oplus b_1 b_2 \cdots b_n = (\epsilon_n^{a_n} \oplus \epsilon_n^{b_n}) \oplus (\epsilon_{n-1}^{a_{n-1}} \oplus \epsilon_{n-1}^{b_{n-1}}) \oplus \cdots \oplus (\epsilon_2^{a_2} \oplus \epsilon_2^{b_2}) \oplus (\epsilon_1^{a_1} \oplus \epsilon_1^{b_1}) \quad (5)$$

式中, $\epsilon_j^e = \underbrace{0 \cdots 0}_{j-1 \uparrow} e \underbrace{0 \cdots 0}_{n-j \uparrow}$, 显然, $\epsilon_j^e \in H_n (1 \leq j \leq n)$, $j-1$ 与 $n-j$ 表示码元 0 的个数。

对于非首位加法位上的码元相加, 即 $\epsilon_j^{e_1} \oplus \epsilon_j^{e_2} (1 < j \leq n)$, 根据上文编码方案可得:

$e_1, e_2 \in E$, 具体计算结果以及进位规则如下:

- 1) 若 $e_1 = e_2, \epsilon_j^{e_1} \oplus \epsilon_j^{e_2} = \begin{cases} \underbrace{0 \cdots 0}_{j-2 \uparrow} e_1 0, j > 2 \\ (2e_1) 0, j = 2 \end{cases};$
- 2) 若 $e_1 \neq e_2$, 且 e_1, e_2 均不为 0, $\epsilon_j^{e_1} \oplus \epsilon_j^{e_2} = (\frac{e_1 + e_2}{4^{\lfloor \frac{e_1 - e_2}{2} \rfloor}}) \underbrace{e \cdots e}_{j-1 \uparrow}$, 其中 e 为 $\{0, e_1, e_2\}$ 对 E 的补集中的元素;
- 3) 若 $e_1 \neq e_2$, 且 e_1, e_2 中有一个为 0, 设 e_1 为码元 0, 则 $\epsilon_j^{e_1} \oplus \epsilon_j^{e_2} = \underbrace{0 \cdots 0}_{j-1 \uparrow} e_2$ 。

计算首位码元, 包括 a_1, b_1 及非首位码元相加对首位产生的进位码元。具体运算规律以表 1 加法查找表的形式给出, 表 1 中的首行及左列分别为相加码元。

表 1 加法运算查找表
Tab.1 Addition Table of Seven Code Cells

相加码元	相加码元																		
	0	1	2	3	4	5	6	10	20	30	40	50	60	100	200	300	400	500	600
0	0	1	2	3	4	5	6	10	20	30	40	50	60	100	200	300	400	500	600
1	1	100	20	2	0	6	10	106	102	200	3	5	600		201	30	4	60	601
2	2	20	200	30	3	0	1	100	201	203	300	4	6	102		302	40	5	10
3	3	2	30	300	40	4	0	1	200	302	304	400	5	20	203		403	50	6
4	4	0	3	40	400	50	5	6	2	300	403	405	500	1	30	304		504	60
5	5	6	0	4	50	500	60	600	1	3	400	504	506	10	2	40	405		605
6	6	10	1	0	5	60	600	601	100	2	4	500	605	106	20	3	50	506	

2 编码运算在正二十面体上的扩展

HLQT 是平面上的编码结构, 正二十面体表面为封闭三角面集合, HLQT 编码运算规则不能完全适用, 为了避免复杂、低效的跨三角面运算, 需要将 HLQT 编码扩展到正二十面体上, 并设计简洁、高效的编码运算方案。

2.1 正二十面体顶点结构

正二十面体共有 20 个三角面及 12 个顶点, 每个顶点周围都连续排列有 5 个共顶点的三角面 (图 5(a)), 将该顶点结构作为正二十面体表面基础结构。沿任意一条边 $V_1 V_i (i = 2, 3, 4, 5, 6)$ 将其剪开, 展开至平面, 得到平面上围绕公共顶点连续排列的 5 个三角形 (图 5(b) 中 $T_1 \sim T_5$)。图 5(b) 等效于从围绕公共顶点连续排列、完整覆盖平面的 6 个三角形中去除其中 1 个三角形 (从图 5(c) 中去除三角形 T_6)。这一变化过程为将平面格网扩展至顶点结构, 建立正二十面体上的格点数学模型提供了思路。

2.2 面瓦片与顶点瓦片

以图 5(c) 作为研究对象, 作三角面内接六边形, 则还存在一个以 V_1 为中心、与 6 个内接六边形为相邻的顶点六边形。定义三角面内接六边形内格点集合为 $P_n^i (1 \leq i \leq 6)$, 其中顶点六边形内格点集合为 P_n^0 , 相邻六边形边界格点集合为 E_n , 令以 V_1 为中心的格点集合 $R_n' = P_n^0 \cup E_n$ 。当 $n=1$ 时, 以不同彩色单元区分 $P_1^i (1 \leq i \leq 6)$, 如图 6(a) 所示, P_1^0 与 E_1 均用黑色单元表示, $R_1' = P_1^0 \cup E_1$ 。以图 6(a) 中格点 e_1, e_2 示例 E_n 的分布特点, e_1 为 P_1^1 与 P_1^3 外接六边形相交边界上的格点, e_2 为 P_1^3 与 P_1^4 外接六边形相交边界上的格点。按照上述规则定义的格点集合 $P_n^i (1 \leq i \leq 6)$ 与 R_n' (黑色单元) 所覆盖的平面部分, 单元之间不重叠且不遗漏。

为建立顶点结构上的格点集合, 借助图 5 所示的正二十面体顶点结构变化, 去除一个 $P_n^i (1 \leq i \leq 6)$ 以及 R_n' 在 T_i 内格点, 拼接处保留一条边界上的格点。以图 6(a) 为例, 去除 P_1^6 以

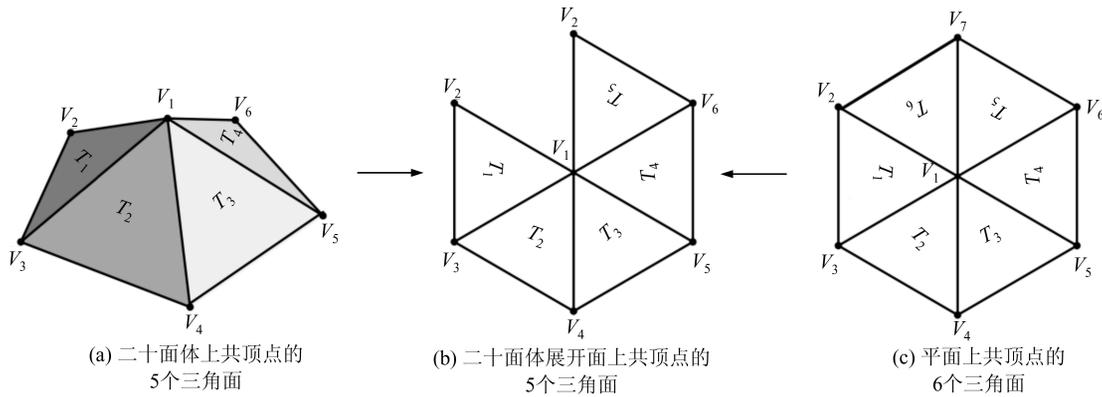


图 5 正二十面体顶点结构展开示意图

Fig.5 Expanded View of Vertex Structure of Icosahedron

及 R_1' 在 T_6 内格点并保留 V_1V_2 边界格点,结果如图 6(b)所示。将删减后的 R_n' 内格点集合定义为 R_n ,如图 6(c)所示拼合边界,得顶点结构格点集合,如图 6(d)所示。需要注意的是,图 6(d)中正二十面体顶点处六边形单元在拼接后变为五边形。将 P_n 与 R_n 分别命名为面瓦片与顶点瓦片。

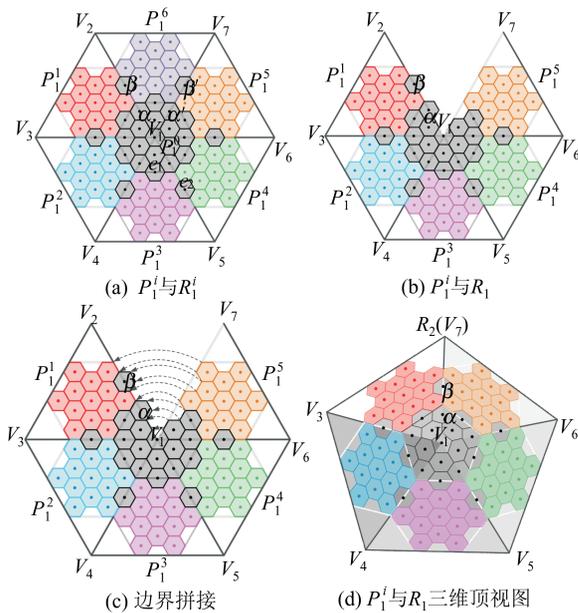


图 6 面瓦片与顶点瓦片

Fig.6 Face Tile and Vertex Tile

记正二十面体表面的四孔六边形离散格网系统第 n 层对应的格点集合为 $G_n (n \geq 1)$, G_n 可分为两个部分:第一部分为 20 个面瓦片 $P_n^j (j = 0, 1, 2 \dots 19)$, j 为面瓦片中心所属三角面编号;第二部分为 12 个顶点瓦片 $R_n^k (k = 0, 1, 2 \dots 11)$, k 为顶点瓦片中心对应顶点编号。即:

$$G_n = \bigcup_{j=0}^{19} P_n^j \cup \bigcup_{k=0}^{11} R_n^k \quad (6)$$

式中, \cup 表示各格点集合的并集。 P_n^j 之间除了中

心不同之外完全等价,只需在格点编码前添加三角面索引编号加以区别;同理,各 R_n^k 只需在格点编码前添加顶点编号加以区别。顶点瓦片与面瓦片编码方案与 HLQT 编码方案相同,顶点瓦片只是对面瓦片单元的稍加删减与扩充。

2.3 编码运算规律

面瓦片位于单个三角面内,满足 HLQT 编码运算规律。顶点瓦片位于 5 个相连三角面上,运算涉及跨三角面单元的处理。由于顶点瓦片由平面六边形裁剪拼接而得,在运算不经过拼接边界时,仍满足 HLQT 编码运算规律;当运算经过拼接边界,由于裁剪了一个三角面的编码单元,平面编码运算规律不再适用,运算需跨越一个三角面。

如图 6 所示,边界 V_1V_2 上格点在拼接后与 V_1V_7 上对应格点一一重合,例如 α 与 α' 、 β 与 β' 。当运算经过格点 α (图 6(a)) 且向 V_1V_7 方向时,可将格点 α 转换至 V_1V_7 上对应格点 α' 编码再运算。经验证,两条边界上格点编码对应转换关系可完全通过编码运算描述,设边界上对应格点编码分别为 $\alpha = b_1b_2 \dots b_n$ 和 $\alpha' = c_1c_2 \dots c_n$, 码元转换关系如下:

$$c_1 = \begin{cases} b_1, b_1 \in \{0, 2\} \\ 201, b_1 = 203 \\ 20, b_1 = 30 \\ 2000, b_1 = 3000 \end{cases} \quad (7)$$

$$c_j = \begin{cases} b_j, b_j \in \{0, 1\} \\ 2, b_j = 3 \\ 3, b_j = 2 \end{cases} \quad (8)$$

式中, $2 \leq j \leq n$ 。由此可实现顶点瓦片上纯编码跨面单元运算。

以接边处跨面邻近单元搜索为例,图 7 为 R_2 编码分布,由于首位码元可能不止一位数字,单

元内编码用逗号分隔首位与第2位码元,为了便于示意接边编码转换,灰色填充单元表示被剪掉的边界编码。邻近单元跨面的单元有两类:(1)格点位于边界上的单元。例如 $\alpha=2,2$,则 $\alpha'=2,3$, α 的邻近单元中有3个(蓝色单元)在拼接后实际为 α' 的邻近单元,根据编码加法运算和式(7)、式(8)求顶点瓦片上 α 的6个邻近单元为: $\alpha\oplus 5,1=3,3$; $\alpha\oplus 0,2=3,0$; $\alpha\oplus 3,3=3,1$; $\alpha'\oplus 5,1=1,2$; $\alpha'\oplus 0,3=1,0$; $\alpha'\oplus 1,2=1,1$ 。(2)一条边与边界重合的单元,例如 $\beta=3,1$,只有一个邻近单元1,1跨面,该邻近单元计算过程如下: $\beta\oplus 0,3=2,2$; $2,2 \xrightarrow{\alpha \rightarrow \alpha'} 2,3$; $2,3\oplus 1,2=1,1$ 。其余5个邻近单元可通过正常编码加法运算得到。

此外,还需注意顶点单元为五边形,只有5个邻近单元,加上5个方向单位向量的对应编码即可。

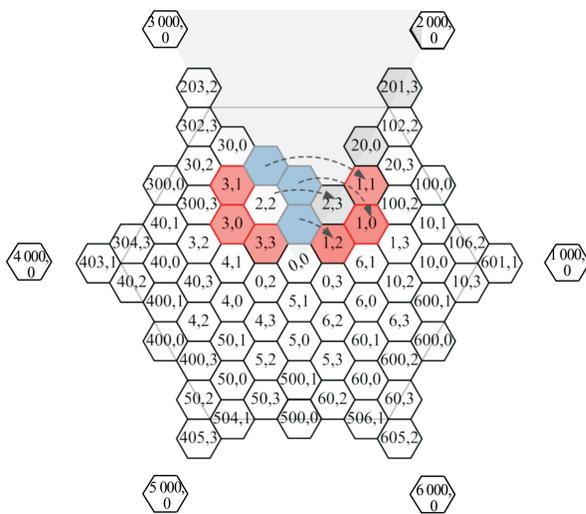


图7 R_2 编码分布
Fig.7 Codes for R_2

3 格网生成与跨面编码运算效率对比实验

本文在 Visual C++ 2012 平台上实现了基于正二十面体的四孔六边形格网构造、显示算法。根据用户输入的格网层次 $n(n \geq 1)$, 首先生成编码集合 P_n, R_n' , 然后根据 §2.2 的规则去除 R_n' 中部分编码得到 R_n , 接着分别复制 P_n 及 R_n 到正二十面体 20 个三角面及 12 个顶点上得到 $P_n^j (j=0, 1, 2 \dots 19)$ 和 $R_n^k (k=0, 1, 2 \dots 11)$, 通过施奈德等积多面体投影生成球面格网, 最后显示正二十面体格网与球面格网的三维图形。整个算法流程如图 8 所示, 运行结果如图 9、图 10 所示, P_n^j 及

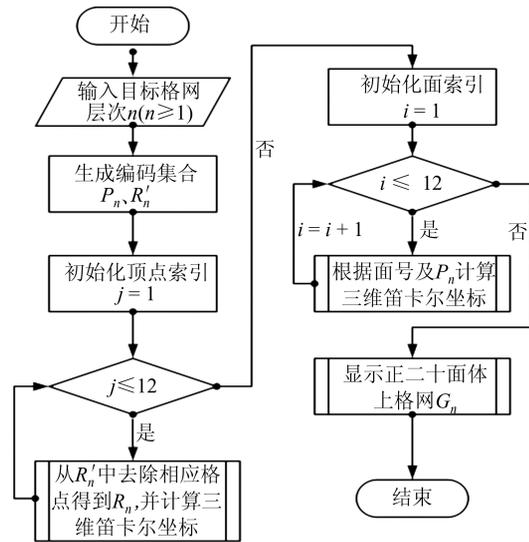


图8 格网系统构造、显示流程图
Fig.8 Flowchart of Construction and Display for the Grid System

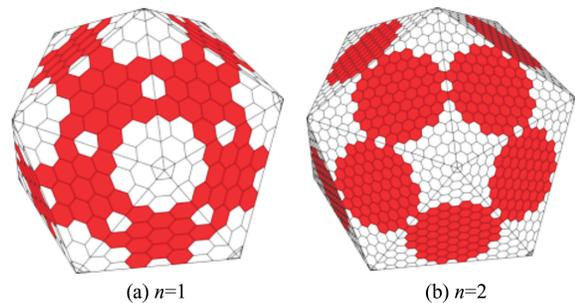


图9 正二十面体上四孔六边形格网
Fig.9 Icosahedral Aperture 4 Hexagonal Grids

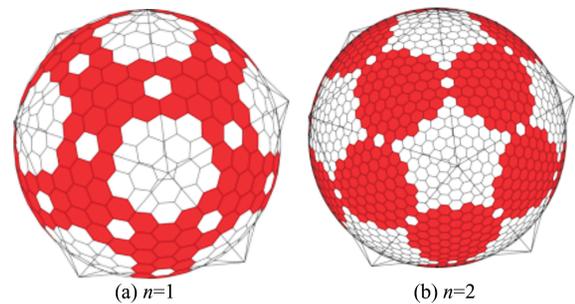


图10 球面四孔六边形格网
Fig.10 Spherical Aperture 4 Hexagonal Grids

R_n^k 分别用红色及白色填充。

为验证本文提出的顶点结构在编码跨面运算中的优势, 将本文方案与 HQBS 对比, 设计了跨面邻近单元搜索实验, 即 §2.3 所述两类单元邻近搜索方法, 并统计各层符合条件的编码个数。全部程序编译为 Release 版本, 并在一台兼容机 (软件配置: Windows 7 x64 旗舰版 + SP1; 硬件配置: Intel Core i5-6500 CPU@3.20 GHz, 8 GB RAM, KingSton 120 GB SSD) 上进行测试。统计两种方案在 6~19 层的运算效率, 即在单位时间

内查找邻近单元的平均个数。单位时间设置为 1 ms,效率比= $\frac{\text{本文方案效率}}{\text{HQBS效率}}$,实验结果见表 2,效率对比曲线、HQBS效率曲线分别如图 11、图 12 所示。

表 2 邻近单元搜索对比实验结果

Tab.2 Comparative Result of Search for Adjacent Cell

层次	单元个数	HQBS效率	本文方案效率	效率比
6	84	168.2	8 886.6	52.8
7	170	318.1	7 746.4	24.4
8	340	329.3	6 297.5	19.7
9	682	324.6	6 027.7	18.6
10	1 364	292.3	5 437.3	18.6
11	2 730	276.6	4 871.3	17.6
12	5 460	255.9	4 500.1	17.6
13	10 922	249.2	3 683.5	14.8
14	21 844	242.8	3 399.0	14.0
15	43 690	232.7	3 346.5	14.4
16	87 382	224.6	3 044.8	13.6
17	174 764	213.6	2 828.0	13.2
18	349 526	203.9	2 556.2	12.5
19	699 052	190.3	2 383.4	12.5

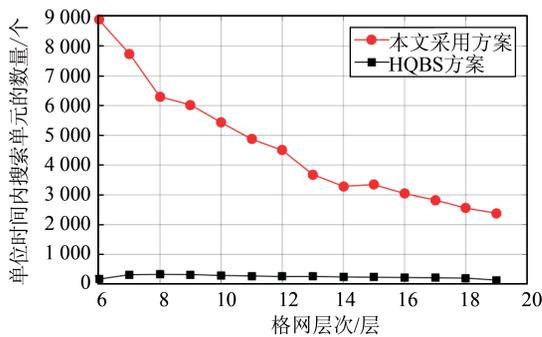


图 11 邻近搜索效率对比曲线

Fig.11 Comparison of Searching Efficiency Curves for Adjacent Cell

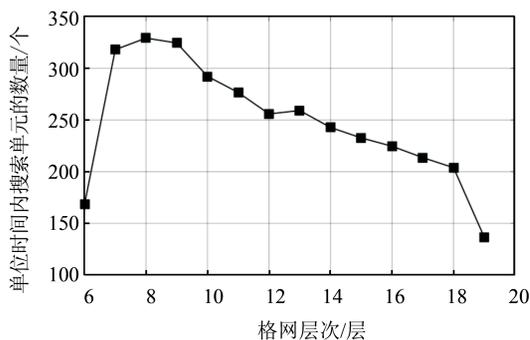


图 12 HQBS效率曲线

Fig.12 Searching Efficiency Curve for HQBS

分析上述实验结果,得出以下结论:(1)本文方案的跨面邻近单元搜索效率约为 HQBS 的

19.6 倍。这主要因为 HQBS 跨面运算需要借助效率不高的浮点数运算来实现三角面间的旋转、平移,而本文提出的纯编码跨面运算规律只涉及整数编码的直接转换,简单高效,更适合计算机处理。(2)两种方案的编码加法运算效率均随层次升高呈下降趋势。主要原因是:编码长度随层次升高线性增加,编码加法运算需逐个码元处理,编码越长,计算量越大。由于本文方案的绝对效率是 HQBS 的几十倍,HQBS 效率的数值变化在图 11 上难以明显看出,图 12 表明 HQBS 的运算效率也呈下降趋势。(3)本文方案效率足以满足绝大多数应用的需求。以对编码运算效率要求较高的实时地形三维可视化为例,对于视点较远的大区域场景,通常采用低分辨率层次格网承载数据,尽管格网单元数目较多,但是效率极高的低层次编码运算可保证处理效率。对于视点较近的局部场景,通常采用高分辨率层次格网承载数据,经过可视区域裁剪后,有效单元数目较少且基本稳定,效率较高的高层次编码运算亦可保证处理效率。

4 结 语

本文结合四孔六边形格网系统在正二十面体上的分布特点,利用顶点瓦片与面瓦片,巧妙地将无限平面上的四孔六边形格网编码运算方案退化到封闭正二十面体表面。与现有方案相比,本文方案不仅形式简洁、理论严密,而且具有较高的跨面编码运算效率。本文研究思路可扩展到其他规则格网系统上,为编码运算方案的设计、分析和优化提供理论支撑。

参 考 文 献

[1] Zhou Chenghu, Ou Yang, Ma Ting. Progress in Geographical Model [J]. *Progress in Geography*, 2009,28(5):657-662 (周成虎,欧阳,马廷. 地理格网模型研究进展[J]. 地理科学进展,2009,28(5):657-662)

[2] Sahr K, White D, Kimerling J. Geodesic Discrete Global Grid System [J]. *Cartography and Geographic Information Science*, 2003,30(2):121-134

[3] Ben Jin, Tong Xiaochong, Zhou Chenghu, et al. Construction Algorithm of Octahedron Based on Hexagon Grid Systems[J]. *Journal of Geo-Information Science*, 2015,17(7):789-797 (贲进,童晓冲,周成虎,等. 基于正八面体的六边形离散格网系统生成算法[J]. 地球信息科学学报,2015,17(7):

- 789-797)
- [4] Sahr K. Hexagonal Discrete Global Grid Systems for Geospatial Computing [J]. *Archives of Photogrammetry, Cartography and Remote Sensing*, 2011, 6(22):363-376
- [5] Gibson L, Lucas D. Spatial Data Processing Using Generalized Balanced Ternary [C]. Processing of PRIP 82, IEEE Computer Society Conference on Pattern Recognition and Image Processing, Las Vegas, Nevada, USA, 1982
- [6] White D, Kimerling J, Sahr K, et al. Comparing Area and Shape Distortion on Polyhedral-Based Recursive Partitions of the Sphere [J]. *International Journal of Geographical Information Science*, 1998, 12(8):805-827
- [7] Sahr K. Location Coding on Icosahedral Aperture 3 Hexagon Discrete Global Grids[J]. *Computers, Environment and Urban Systems*, 2008, 32(3):174-187
- [8] Sahr K. Icosahedral Modified Generalized Balanced Ternary and Aperture 3 Hexagon Tree: US20110022296A1[P]. 2011-01-26
- [9] Vince A. Indexing the Aperture 3 Hexagonal Discrete Global Grid [J]. *Journal of Visual Communication and Image Representation*, 2006, 17(6):1 227-1 236
- [10] Tong Xiaochong, Ben Jin. The Principle and Methods of Discrete Global Grid Systems for Geospatial Information Subdivision Organization [M]. Beijing: Surveying and Mapping Press, 2016(童晓冲, 贲进. 空间信息剖分组织的全球离散格网理论与方法[M]. 北京:测绘出版社, 2016)
- [11] Ben J, Li Y L, Zhou C H, et al. Algebraic Encoding Method of the Aperture 3 Hexagonal Discrete Global Grid System[J]. *Science China Earth Science*, 2018, 61(2):97-109

Code Operation Scheme for the Icosahedral Aperture 4 Hexagonal Grid System

WANG Rui¹ BEN Jin¹ DU Lingyu¹ ZHOU Jianbin¹ LI Zhuxin²

1 Institute of Surveying and Mapping, Information Engineering University, Zhengzhou 450001, China

2 Troops 61287, Chengdu 610000, China

Abstract: Discrete global grid systems(DGGS) are digital reference models with multi-resolution for earth and its structure supports the fusion processing of multi-source geospatial information. The icosahedral DGGS have aroused wide attention for their excellent geometric properties and the research difficulty is the code operation scheme for the spherical hexagonal grid system currently. It is proved that DGGS based on the regular polyhedrons are topologically equivalent to the relative regular polyhedral grid systems and their schemes for code operation are topologically equivalent as well. According to this, combining the characteristics of the aperture 4 hexagonal grid system on the surface of the icosahedron, this paper defines vertex tile and face tile based on hexagon lattice quad tree and proposes the code operation scheme for the icosahedral aperture 4 hexagonal grid system. This scheme realizes the cross-face operation of cells using code operation with high efficiency and overcomes the drawback of existing schemes brought by the use of floating numbers for cross-face operation with low efficiency. In addition, the search efficiency for cross-face adjacent cells is about 19.6 times that of hexagonal quad balanced structure.

Key words: discrete global grid system; regular icosahedron; hexagon; code operation

First author: WANG Rui, PhD candidate, specializes in spatial data model. E-mail:wr_paper@126.com

Corresponding author: BEN Jin, PhD, professor. E-mail: benj@reis.ac.cn

Foundation support: The National Natural Science Foundation of China(41671410); the National Key Research and Development Program of China(2018YFB0505301).

引文格式: WANG Rui, BEN Jin, DU Lingyu, et al. Code Operation Scheme for the Icosahedral Aperture 4 Hexagonal Grid System[J]. *Geomatics and Information Science of Wuhan University*, 2020, 45(1): 89-96. DOI:10.13203/j.whugis20180191(王蕊, 贲进, 杜灵瑀, 等. 正二十面体四孔六边形格网系统编码运算[J]. *武汉大学学报·信息科学版*, 2020, 45(1): 89-96. DOI:10.13203/j.whugis20180191)