Cartography and Geographic Information Science

ISSN: 1523-0406 (Print) 1545-0465 (Online) Journal homepage: https://www.tandfonline.com/loi/tcag20

A novel identifier scheme for the ISEA Aperture 3 Hexagon Discrete Global Grid System

Franz-Benjamin Mocnik

CARTOGRAPHY AND GEOGRAPHIC INFORMATION SCIENCE
2019, VOL. 46, NO. 3, 277–291
https://doi.org/10.1080/15230406.2018.1455157

A novel identifier scheme for the ISEA Aperture 3 Hexagon Discrete Global Grid System

Franz-Benjamin Mocnik

Institute of Geography, Heidelberg University, Germany

ABSTRACT
Geospatial data is often spatially aggregated by the use of Discrete Global Grid Systems. References to grid cells are needed for the communication of such data, and different identifier schemes have accordingly been introduced in literature. These schemes suffer, however, from being hard to understand for non-experts, and the geometry of a cell cannot be inferred from its identifier without complex computations. In this article, a novel identifier scheme that encodes the geographic coordinates of the centroid of a cell is proposed, which comes at the cost of potentially being ambiguous in case of a very fine-grained grid. We reason and computationally demonstrate that ambiguity does though not occur for real-world applications. The novel identifier scheme minimizes the amount of data to be communicated, for example, between a server and a client application, and it allows to infer approximate geometries of the cells only by their identifiers.

ARTICLE HISTORY
Received 23 December 2017
Accepted 18 March 2018

KEYWORDS
Discrete Global Grid System (DGGS); Inverse Snyder Equal-Area Projection (ISEA); ISEA Aperture 3 Hexagon Discrete Global Grid System (ISEA3H); grid; indexing; identifier; performance

1. Introduction

Geographical information is often aggregated by space and time when being analyzed. Such an aggregation is easy to achieve for one-dimensional time. For two-dimensional space, such an aggregation is though harder to achieve because most geometric shapes do not tessellate the plane or the sphere. Among the commonly used tessellations are quadratic, triangular, and hexagonal grids, but sometimes additional cells of different shape are needed (Kidd, 2005). Information is often available on a global scale. Accordingly, not only local but also global grids – *Discrete Global Grid Systems (DGGSs)* – are used. Such DGGSs are obviously advantageous for global information, but even in case of information about restricted areas, DGGSs can be of benefit. Data can, for example, be cached in a simple way if grid cells can be "reused" for different areas.

Discrete Global Grid Systems have been widely used across different domains. Among these domains are climate research (Brettschneider, 2008; Elsner & Jagger, 2010), oceanographic research (Lin, Zhou, Xu, Zhu, & Lu, 2017), ecological research (Dulvy et al., 2014; Schipper et al., 2008; Strassburg et al., 2010), and remote sensing (Nguyen, Cressie, & Bravermann, 2012; Nguyen, Katzfuss, Cressie, & Bravermann, 2014). In these fields, also methodological research about the use of DGGSs has been conducted, for example, related to climate research (Elsner, Hodges, & Jagger, 2012; Hodges, & Jagger, 2012), and to ecology (Birch, Oom, & Beecham, 2007). The project *OSMatrix* (Roick, Hagenauer, & Zipf, 2011; Roick, Loos, & Zipf, 2012) visualizes different aspects of information from OpenStreetMap (OSM), a major source of Volunteered Geographic Information (VGI). In addition, self-organizing maps have been implemented on top of DGGSs (Gevrey, Worner, Kasabov, Pitt, & Giraudel, 2006; Park, Lek, Scardi, Verdonschot, & Jørgensen, 2006), and a global gazetteer *Wāhi* uses DGGSs (Adams, 2017). Further examples have been listed by Birch et al. (2007). A widely used implementation of algorithms to compute such grids has been provided by Sahr (2017), and a specification of DGGSs has been provided by Purss et al. (2017).

Grid systems are not easy to achieve, in particular on a global scale. A DGGS would ideally tessellate the surface of the Earth with disks, because disks represent the idea of an ideal neighborhood. The grid cells should be of equal size, and they should all be of the same shape, for example, regular polygons without any distortion. Furthermore, the grid should ideally allow for an aggregation that does not obfuscate the geographical processes that render the data, among them diffusion processes and movement through space and time. Such an ideal DGGS does, however, not

CONTACT Franz-Benjamin Mocnik mocnik@uni-heidelberg.de
The underlying research materials for this article can be accessed at http://www.mocnik-science.net/#mocnik2018b.

exist because different properties contradict. The surface of the Earth can, for example, not be tessellated with disks, and a tessellation *only* by hexagons – these are good approximations to a disk – is not possible either.[1] Such ideal DGGSs have been approached in different ways (Kidd, 2005). Tessellations with hexagons, accompanied with some pentagonal cells, have been adopted (Carr, Kahn, Sahr, & Olsen, 1997). In fact, a tessellation with hexagons seems to be advantageous in many cases (Birch et al., 2007). There is, however, no optimal choice of a tessellation pattern in general. The statistical analysis of the characteristics of the data benefits, for example, from different tessellation patterns depending on the scale of the analysis as well as on the scale of the underlying geographical processes (Yfantis, Flatman, & Behar, 1987). General issues of DGGSs have been discussed by Battersby, Strebe, and Finn (2016), including the modifiable areal unit problem (MAUP), which has been introduced by Openshaw and Taylor (1979) in general but also applies to grid systems in particular; the choice of map projections in respect to DGGSs; and other limitations of the representation of DGGSs.

One of the most used DGGSs is the *ISEA Aperture 3 Hexagon Discrete Global Grid System* (ISEA3H; Carr et al., 1997; Sahr, White, & Kimerling, 2003), which tessellates the surface of the Earth with slightly distorted hexagons and some pentagons. To foster a broader use of the ISEA3H grid, there is a need to simplify the computation, use, and communication of the grid and its cells. The global gazetteer *Wāhi* can be seen as an approach to such simplification, because it offers an intelligible way to refer to such cells by means of URIs (Adams, 2017). Further simplification is, however, desirable, for example, in case of the following typical setup: a server computes the grid and aggregates data from a database, the result is transferred to a web client, and the web client visualizes the aggregated data. A scheme to identify and communicate grid cells efficiently is needed in such and similar cases. Existing identifier schemes exist, but they suffer from several disadvantages:

Issue 1. The identifiers do not reveal much information about the location of a cell in terms of geographic coordinates, which makes the identifier scheme hard to understand for non-experts, and experts from other domains as well.

Issue 2. Many identifier schemes require complicated computations.

Issue 3. The geometry of a cell cannot easily be inferred from its identifier.

This article is guided by the research question of **how to define an identifier scheme that compensates for the aforenamed disadvantages**. In the following section, we describe the construction of the ISEA3H grid and existing identifier schemes in order to provide the necessary background information about this particular type of DGGSs (Section 2). Subsequently, we introduce a new scheme of identifiers for the ISEA3H grid, which is based on the geographic coordinates of the centroid of the cell and is thus easy to understand, even for non-experts (Section 3). The scheme compensates for Issues 1 to 3, which comes with the downside of only being valid up to resolution 22. This resolution is, however, no real obstacle because the grid cells have a diameter of less than 50 m and an area of 1625 m^2 at this resolution, much less than is practically needed for most applications. In addition, two identifiers might refer to the same cell under very rare conditions if the computations are performed with too low precision. We provide the constants used for the generation of the ISEA3H grid with a high precision, which ensures uniqueness as is demonstrated in case of a reference implementation for resolution at least up to 21 (Section 4). The scheme of identifiers can be refined by limiting the precision of the incorporated coordinate values, depending on the resolution of the grid and the purpose. Such adaptive identifiers are shorter than non-adaptive ones (Section 5). For given coordinates, the grid cell containing these coordinates can easily be determined, which is of high importance for data aggregation. This determination turns, in fact, out to be, more or less, only the rounding of a number – this makes the determination very performant (Section 6). When a bunch of identifiers from neighboring cells have been communicated to a web client, the geometries of the cells can finally be inferred from the identifiers, which reduces the amount of transferred data. There is no need to explicitly communicate the geometry or even the location of the cell, because the information is contained in the identifier of each cell and, of its adjacent cells. This process of inferring the geometry of a cell is efficient and can be executed in real time (Section 7). The identifier scheme results in longer identifiers than in the case of a simple enumeration of the cells, but comes with the merit of containing geometric information. The merits and limitations are discussed in detail (Section 8). A reference implementation is available on http://github.com/giscience/geogrid and http://github.com/giscience/geogrid.js.

2. The ISEA Aperture 3 Hexagon Discrete Global Grid System

One of the best known and most used *Discrete Global Grid Systems* (DGGSs) is the *ISEA Aperture 3 Hexagon*

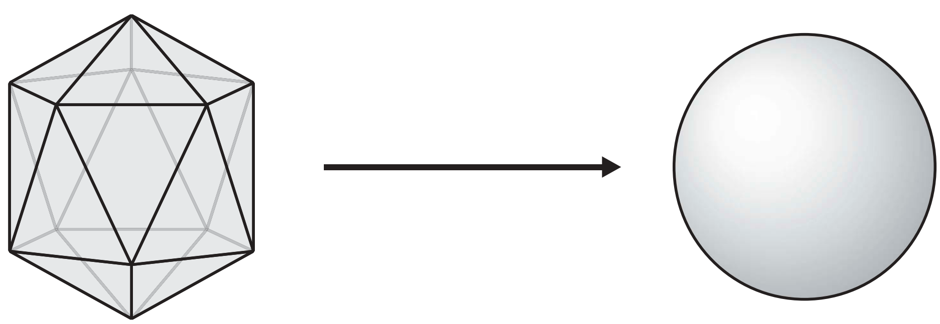
Discrete Global Grid System (ISEA3H). The ISEA3H grid aims at partitioning the surface of the globe into hexagonal cells of equal size. Geometric computations on the surface of the sphere are, however, more complicated than in flat space. The globe, represented as a sphere, is thus mapped to an icosahedron, which consists of twenty flat triangles (Figure 1a). On each of these flat triangles, a grid can easily be introduced. This grid is then mapped back from the icosahedron to the sphere, which results in the desired grid on the surface of the globe. To ensure that all hexagonal cells are of equal size, the *Inverse Snyder Equal-Area Projection* (ISEA) is used. This approach of constructing the ISEA3H grid is discussed in the remainder of the section in more detail.

The ISEA projection has been first published by Snyder (1992). The projection maps the icosahedron, or another Platonic solid, to the sphere by dividing each of the faces into smaller triangles. Each of these triangles consists of the center of the face and two adjacent vertices of the face (Figure 1b). A point in such a triangle is mapped to the sphere by adapting the angle and the distance to the center such that the resulting projection becomes equal-area. As a result, angles and distances are slightly distorted. The angular distortion is less than 17.27°, and the scale variation is less than 16.3%. The computational mapping from the icosahedron to the sphere by the ISEA projection is slow due to iterative root finding but optimizations have been discussed (Harrison, Mahdavi-Amiri, & Samavati, 2012).

The ISEA3H grid consists of a rule to generate a grid on the icosahedron, and an inverse projection from the icosahedron to the sphere, including a relative orientation. As the inverse usually creates "ruptures" at the vertices of the icosahedron, the icosahedron is often rotated relative to the sphere (by choosing an orientation) such that it is symmetric along the equator and such that as many as possible vertices of the icosahedron are mapped to points in the ocean. In case of an optimal choice, one face of the icosahedron is mapped to the sphere such that one of its vertices is mapped to latitude 58.28252559° N, longitude 11.25° E; and the other two vertices of the face to the equator (Sahr et al., 2003). Only one vertex of such an rotated icosahedron is not mapped to a point in the ocean; it is located in Sichuan Province, southwest China.

A regular grid of hexagonal cells can be introduced on each of the faces of the icosahedron. The construction of the grid starts with one hexagonal cell only and can iteratively be refined (Figure 2). The number of refinement steps is called the *resolution* of the resulting grid. It

(a)



(b)

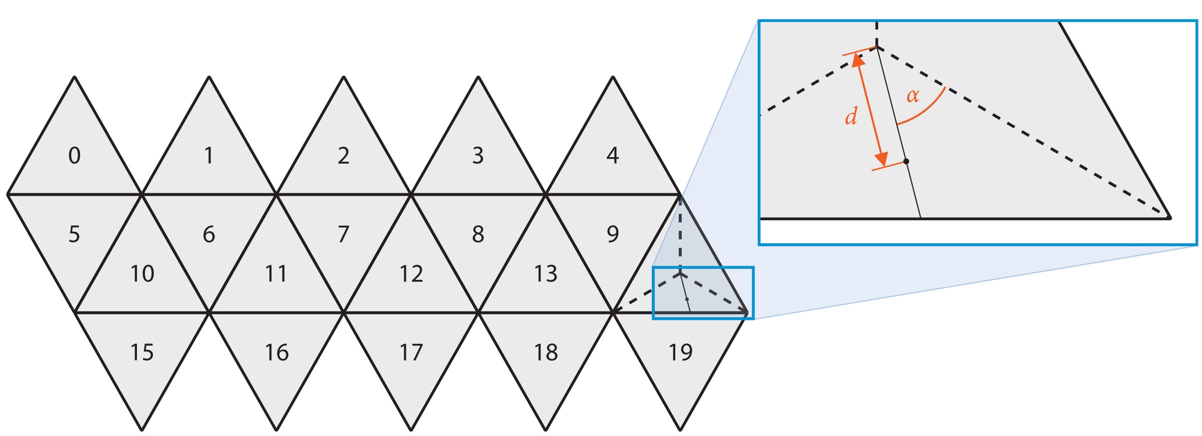


Figure 1. Inverse Snyder Equal-Area Projection (ISEA). (a) The Inverse Snyder Equal-Area Projection maps the icosahedron to the sphere. (b) Unfolded icosahedron with indexed faces. A point on the icosahedron can be mapped to the sphere by subdividing the corresponding face into three triangles (as is exemplarily shown here for face 14), and by then adjusting the angle α and distance d such that the resulting inverse projection is equal-area.

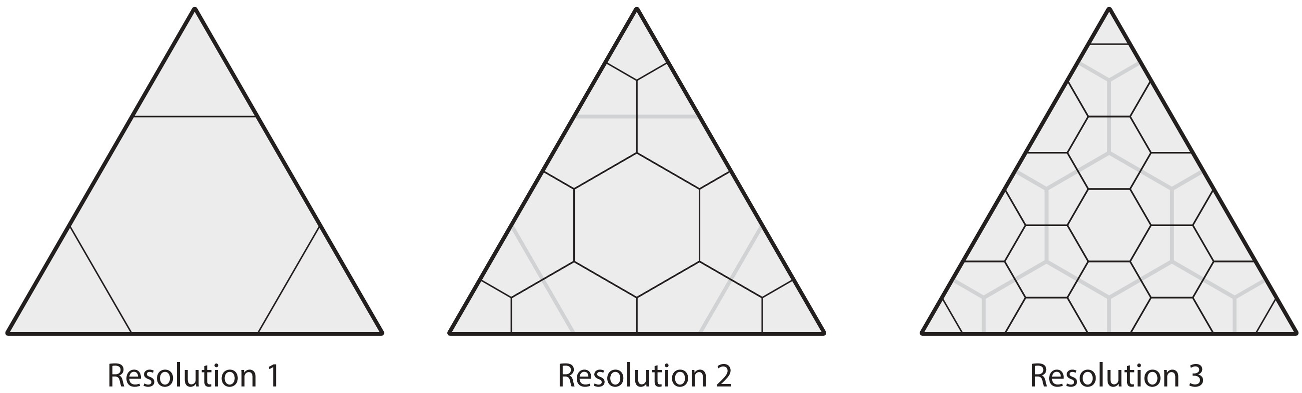


Figure 2. Grid at different resolutions. In case of resolution $r > 1$, the grid of resolution $r - 1$ is depicted in light gray.

should be noted that a refinement step does not subdivide single hexagonal cells – a hexagon cannot be partitioned into smaller hexagons – but rather subdivides the entire triangle face. The ratio of the number of hexagonal cells in refinement step $r + 1$ to the one in step r is called the *aperture* of the grid. For hexagonal cells, the smallest possible aligned aperture is aperture 3. In the case of aperture 3, the grid introduced on single faces of the icosahedron fit well together: the cells at the edge of a face match the cells of neighboring faces, and some hexagonal cells thus stretch across two faces. At the vertices of the icosahedron, five faces meet, which results in a pentagonal cell instead of a hexagonal one. Such a pentagonal cell has a vertex of the icosahedron as a center. The area of a pentagonal cell is only 5/6 of a hexagonal cell. While the number of hexagonal cells depends on the resolution of the grid (Table 1), there are always twelve pentagonal cells – one for every vertex of the icosahedron. The pentagonal cells are all of the same size, which necessarily incorporates some angular distortion.

Several methods to index the cells of the ISEA3H grid as well as of other DGGSs have been discussed and used, each of them having its own advantages and disadvantages (Mahdavi-Amiri, Samavati, & Peterson, 2015). An intuitive indexing method assigns to each cell coordinates, usually consisting of two (Mahdavi-Amiri, Bhojani, & Samavati, 2013; Mahdavi-Amiri, Harrison, & Samavati, 2015; Sahr, 2008) or three integers (Ben, Ton, & Chen, 2010; Vince, 2006b). These integers refer to the position of a cell's centroid in a coordinate system that is defined by axes (Figure 3a). Such indexing is either performed on each face separately (Mahdavi-Amiri et al., 2015), or on the unfolded icosahedron (Mahdavi-Amiri et al., 2013; Sahr, 2008), but these methods differ both

Table 1. The ISEA3H grid at different resolutions.

Resolution	Number of pentagonal cells	Number of hexagonal cells	Distance[a] between centroids of hexagonal cells	Area of hexagonal cell
1	12	20	3717.40 km	17,002,187.390803 km^2
2	12	80	2146.24 km	5,667,395.796934 km^2
3	12	260	1239.13 km	1,889,131.932311 km^2
4	12	800	715.41 km	629,710.644104 km^2
5	12	2420	413.04 km	209,903.548035 km^2
6	12	7280	238.47 km	69,967.849345 km^2
7	12	21,860	137.68 km	23,322.616448 km^2
8	12	65,600	79.49 km	7774.205483 km^2
9	12	196,820	45.89 km	2591.401828 km^2
10	12	590,480	26.50 km	863.800609 km^2
11	12	1,771,460	15.30 km	287.933536 km^2
12	12	5,314,400	8.83 km	95.977845 km^2
13	12	15,943,220	5.10 km	31.992615 km^2
14	12	47,829,680	2.94 km	10.664205 km^2
15	12	143,489,060	1.70 km	3.554735 km^2
16	12	430,467,200	0.98 km	1.184912 km^2
17	12	1,291,401,620	0.57 km	0.394971 km^2
18	12	3,874,204,880	0.33 km	0.131657 km^2
19	12	11,622,614,660	0.19 km	0.043886 km^2
20	12	34,867,844,000	0.11 km	0.014629 km^2
21	12	104,603,532,020	0.06 km	0.004876 km^2
22	12	313,810,596,080	0.04 km	0.001625 km^2

[a]The distance between two hexagonal cells varies due to the distortion of angles and distances. The table provides the minimum distance between the centroids of two regular hexagonal cells on the sphere.

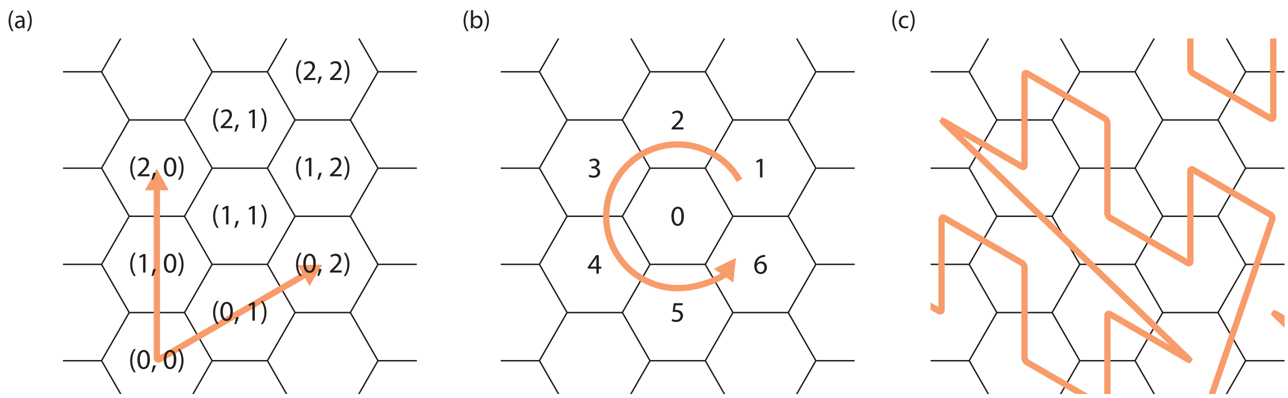


Figure 3. Examples for indexing methods for Discrete Global Grid Systems. (a) Indexing by coordinates along axes (example with two axes); (b) Indexing by neighborhoods (simplified example); (c) Indexing along space-filling curves (example: Morton order).

from geographic coordinate systems. Such coordinate-based indexing methods allow for an efficient computation of all cells. Hierarchy-based indexing methods derive the indices at a given resolution r by the indices at resolution $r-1$ (Dutton, 1996; Lucas & Gibson, 1991; Tong, Ben, Wang, Zhang, & Pei, 2013; Vince, 2006a; Vince & Zheng, 2009). This allows to efficiently derive the indices of cells in the neighborhood of a given cell (Figure 3b). Even identifier schemes using space-filling curves have been discussed (Bai & Zhao, 2004; Bartholdi & Goldsman, 2001; White, 2000), which are advantageous in terms of the length of indices (Figure 3c).

3. A new scheme of identifiers

A fast and efficient way of determining the identifier (ID) of a cell is important for applications that collect and aggregate data by cells. Algorithms that only need local information instead of global knowledge about the entire grid seem to be advantageous, in particular when performing the collection and aggregation in real time or when a low latency is required. As has been discussed in the last section, different identifier schemes have been proposed and are in use. Many of these schemes share a global view: the ID of a given cell can only be computed when the position of the cell inside the grid is known and after the position of further cells has been computed. In this section, we propose a straightforward scheme for IDs, which assigns to each cell a number that encodes the geographic coordinates of the centroid of the cell (latitude and longitude), as well as the resolution. While this scheme seems to be trivial, it is not clear how this information can be encoded in an efficient way and whether the resulting IDs are unique.

The proposed IDs are composed of four parts (Figure 4). Part A encodes whether the cell is a hexagon or a pentagon, while part B encodes the resolution of the grid, as well as the signs of the latitude and longitude values of the centroid of the cell. Part C encodes the absolute value of the latitude of the centroid of the cell, and part D, of its longitude. Part A consists of a sign; part B, of two digits; part C, of eight digits; and part D, of nine digits. As part B is always less than 88 and as an ID always consists of nineteen digits or less, it can be represented as a signed 64-bit integer, a native word length for many modern computers. More detailed, the parts adhere to the following specification:

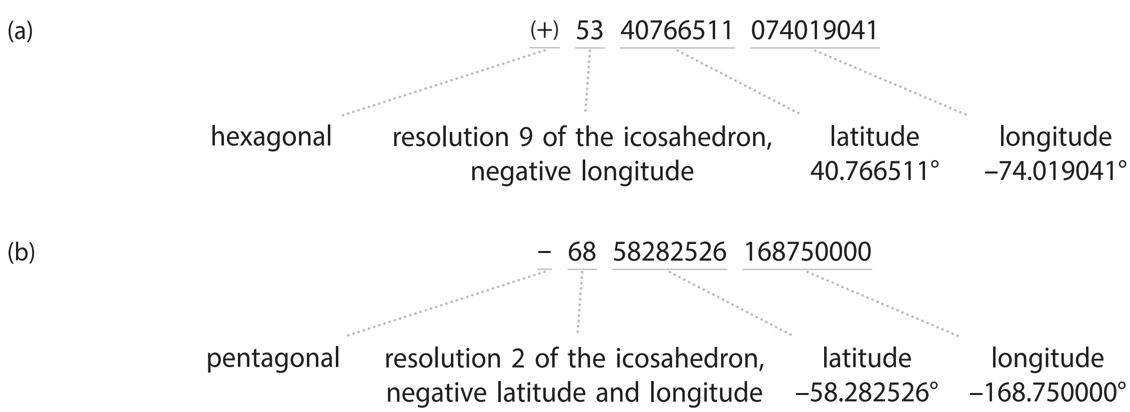


Figure 4. Examples of grid cell identifiers. (a) Grid cell containing the Statue of Liberty in New York City, USA; (b) Pentagonal grid cell in the Southern Ocean.

Part A (Hexagon/pentagon). This part consists of the sign of the resulting ID. The sign is positive in case of a hexagonal cell, and negative in case of a pentagonal cell.

Part B (Resolution of the grid; signs of latitude and longitude). This part consists of two digits, representing the resolution of the grid incremented by 22 in case of negative latitude, by 44 in case of negative longitude, and by 66 in case of negative latitude and longitude. In case that the latitude/longitude is strictly less than $0.5{\cdot}10^{-6\circ}$ or that the difference of longitude to 180° or −180° is strictly less than $0.5{\cdot}10^{-6\circ}$ the respective sign is always regarded as being positive.

Part C (Latitude). This part represents the latitude with two pre-decimal and six decimal places. The sign of the latitude is ignored in this part.

Part D (Longitude). This part represents the longitude with three pre-decimal and six decimal places. The sign of the longitude is ignored in this part, and the longitude is regarded to be greater than −180° and strictly less than 180°. The longitude is by definition regarded as being 0° if the latitude differs from −90° or 90° by strictly less than $0.5{\cdot}10^{-6\circ}$.

This identifier scheme only works up to resolution 22 because the resolution could otherwise not be reconstructed from part B. In this case, a value of 25 could, for example, be interpreted as resolution 3 and negative latitude, or resolution 25 and positive latitude. The reason behind not choosing a larger value than 22 in part B is to ensure that it can be represented as a signed 64-bit integer. We, however, discuss in the following section that the identifier would otherwise be valid up to a resolution of about 32 because ambiguous IDs cannot occur for these resolutions.

4. Uniqueness of the identifiers

Identifiers are ideally unique, which allows to distinguish different cells. While the location of a grid cell is unique, IDs according to the scheme introduced in the previous section could be ambiguous due to the limited precision of the representation of the values of latitude and longitude, and due to rounding errors during the computation. Both restrictions are different in nature: the former is an artifact of the identifier scheme and cannot be avoided without extending the range of digits to be used. The latter error is, however, an artifact of the computation and can thus be avoided. We discuss both aspects in this section.

The identifier scheme requires latitude and longitude to be represented with only six decimal places. On a great circle, this corresponds to about 0.11 m, which restricts the resolution to around 32 or less – grid cells of resolution 32 have an approximate distance of 0.15 m. In fact, the same difference in longitude can represent very differing distances on the sphere, depending on the latitude. The poles represent singularities: all locations with −90° or 90° latitude are mapped to the poles, independent of their longitude. To ensure uniqueness of the ID, the longitude is considered 0 near the poles, as described above.

Rounding errors inevitably occur during the computation. As long as these errors are small enough compared to the resolution, ambiguous cell IDs due to rounding errors are very unlikely but cannot be excluded in general. Values with a low number of decimal places occur more often in IDs, and these values are more prone to such ambiguity. To avoid such effects, the precision at which the computation is performed is added to the computed values of latitude and longitude. The resulting rounded values do, as a result, much less likely depend on rounding errors, and ambiguity can practically be avoided.

Computations are often performed with high precision. Some of the constants described by Snyder (1992) are only provided with low accuracy. To ensure higher accuracy, the golden ratio needs first to be computed:

$$\varphi = \frac{1+\sqrt{5}}{2}.$$

The latitude of the center of the faces 6–10 (Figure 1b) is given by

$$F = \arctan\left(\frac{1}{2\varphi^2}\right).$$

The spherical distance (in degrees) from the center of the polygon to any of its vertices on the sphere can be computed as:

$$g = F + 2\arctan\varphi - 90.$$

The length of a face of the icosahedron is, in turn, given as $2G$ where

$$G = R' \cdot \tan g \cdot \sin 60,$$

and R' denotes the midradius of the icosahedron. Finally, the ratio of the midradius of the icosahedron R' to the radius of the sphere R is given by

$$\frac{R'}{R} = \frac{1}{\tan g} \cdot \sqrt{\frac{\pi(\tilde{G}-\theta)}{45 \cdot \sin(2\theta)}},$$

where $\tilde{G} = 36$ is the spherical angle (in degrees) between the radius vector to the center and the radius vector to an adjacent edge of a spherical triangle on the sphere that corresponds to a face of the icosahedron, and $\theta = 30$, the corresponding plane angle (in degrees) for a plane triangular face of the icosahedron. Instead of computing this ratio by the division of the areas of corresponding triangles (resulting in a lower accuracy), as is done by Snyder (1992), the ratio can geometrically be computed as

$$\frac{R'}{R} = \left(\frac{1}{2\sqrt{5}} + \frac{1}{6}\right)\sqrt{\pi\sqrt{3}}.$$

Rounding errors can, even despite the use of the above constants, have twofold consequences: First, the centroids of some cells belong to the boundary of two or more faces of the icosahedron. These centroids can be computed by mapping the corresponding point of the respective face to the sphere, and both computations may result in differing pairs of coordinates due to rounding errors. This is despite that both pairs of coordinates should coincide. This issue applies, in particular, to pentagonal cells, located at the vertices of the icosahedron – the points where five faces meet. Secondly, a cell which is completely contained in the inner of a face may gain different IDs on different machines due to rounding errors. If the computations are though performed several times by one implementation on the same machine, the results can be expected to coincide. This second consequence thus only applies if IDs are computed on several machines, which hinders the use of an ID for the communication of grid cells across different systems. An ID does always provide a good approximation of the location of its centroid on Earth, and two IDs can be checked for equality by computing the distance between the centroids of these cells. Independent of these two consequences, the IDs can always be used to efficiently cache information on one machine – in the worst case, the same information is cached in respect to two or more different IDs. In addition, the IDs can be used to efficiently communicate the information to a client application, as is described in Section 6.

An exemplary implementation[2] of the ISEA projection and the ISEA3H grid demonstrates that the IDs are, despite of these theoretical issues, unique for most real-world applications (Table 2). The table illustrates two different aspects of this implementation. First, ambiguous IDs are counted. An ID of a cell is considered as being ambiguous if there exists another cell on the same face with the same ID. We have argued that the IDs for a given resolution remain unique at least up to resolution 32 – different cells have different IDs. The exemplary implementation corroborates this fact up to resolution 21. Secondly, non-unique IDs were counted. Theoretically, more than one ID could be assigned to a cell at the boundary of a face, when assuming the centroid of such a cell to be part of different faces. For the examination of non-unique IDs, the overall number of IDs is counted. This number should, ideally, coincide with the number of cells. In case of non-unique IDs, the number of IDs is lower than the number of cells. If the number of IDs would though be greater, more than one ID would be assigned to some cell located at the boundary of a face. As can be seen by the table, none of these issues arise for grids up to resolution 21. Despite the fact that some IDs may theoretically stay ambiguous even for lower resolutions in case of other implementations, practical issues will rarely arise. To ensure uniqueness of IDs even for different resolutions of the grid, we restrict the resolution to a maximum of 22.

Table 2. Uniqueness of identifiers (IDs) for the ISEA3H grid at different resolutions.

Resolution	Number of unique IDs	Number of cells	Number of ambiguous[a] IDs	Difference between number of unique IDs and the number of cells
1	32	32	0	0
2	92	92	0	0
3	272	272	0	0
4	812	812	0	0
5	2432	2432	0	0
6	7292	7292	0	0
7	21,872	21,872	0	0
8	65,612	65,612	0	0
9	196,832	196,832	0	0
10	590,492	590,492	0	0
11	1,771,472	1,771,472	0	0
12	5,314,412	5,314,412	0	0
13	15,943,232	15,943,232	0	0
14	47,829,692	47,829,692	0	0
15	143,489,072	143,489,072	0	0
16	430,467,212	430,467,212	0	0
17	1,291,401,632	1,291,401,632	0	0
18	3,874,204,892	3,874,204,892	0	0
19	11,622,614,672	11,622,614,672	0	0
20	34,867,844,012	34,867,844,012	0	0
21	104,603,532,032	104,603,532,032	0	0

[a]An ID of a cell is considered as being ambiguous if there exists another cell on the same face with the same ID. The ambiguous IDs are counted per face and then added.

5. Adaptive identifiers

Identifiers can serve several purposes, among them the unique identification of a cell and the communication of its centroid. In case of the identification, the introduced IDs can potentially be shortened without becoming ambiguous. Even in case that the coordinates of the centroid are to be inferred from the IDs, the coordinates of the centroid may only be needed with limited precision. This is very similar to the *Military Grid Reference System* (MGRS), in which coordinates can be encoded at different precision levels (Defense Mapping Agency, 1989; National Geospatial-

Intelligence Agency, 2014). The MGRS can be seen as a variant of the *Universal Transverse Mercator* (UTM) coordinate system, which makes use of UTM zones and latitude bands, 100,000 km^2 grid cells, and UTM coordinates for easting and northing inside these grid cells to encode a location on Earth. Depending on the required precision, the values for easting and northing are provided with a precision ranging from 1 m to 100 km. In this section, we propose an adaptive identifier scheme for ISEA3H grid cells that allows for shorter modifiers of varying length, depending on the purpose and the resolution of the grid.

The length of an ID can easily be adapted by using coordinate values with a differing number of decimal places in part C and part D. There are two major factors that guide how many decimal places are needed. First, the resolution guides the required precision. In case of unique IDs, the precision should be larger than the distance between two cells Δ_r for a given resolution r. As a uniform precision shall be used for all IDs of the grid, it is sufficient to determine the required number of decimal places p at the equator as

$$p_r = \left\lceil -\log_{10} \frac{\Delta_r}{2\pi r_{\text{authalic}}/360} \right\rceil$$

where r_{authalic} is the authalic radius of the Earth. Secondly, the purpose guides whether additional precision is required. Only $q = p_r$ decimal places need to be considered in case that no additional precision is required (*adaptive unique IDs*). If the coordinates of the centroid of a cell are, however, to be inferred from the ID with a precision of 1% of the diameter of a cell, that is, with a precision of $\Delta_r/100$, two additional decimal places need to be considered, which results in $q = p_r + 2$ decimal places (*adaptive 1% IDs*). We allow zero decimal places as a minimum, and six decimal places as a maximum, the latter also being the case for non-adaptive IDs (Table 3).

Table 3. Adaptive identifiers (IDs) for the ISEA3H grid at different resolutions.

		Adaptive unique IDs		Adaptive 1% IDs	
Resolution	Distance	Decimal places	Saving[a]	Decimal places	Saving[a]
1	3717.40 km	0	63%	1	53%
2	2146.24 km	0	63%	1	53%
3	1239.13 km	0	63%	1	53%
4	715.41 km	0	63%	2	42%
5	413.04 km	0	63%	2	42%
6	238.47 km	0	63%	2	42%
7	137.68 km	0	63%	2	42%
8	79.49 km	1	53%	3	32%
9	45.89 km	1	53%	3	32%
10	26.50 km	1	53%	3	32%
11	15.30 km	1	53%	3	32%
12	8.83 km	2	42%	4	21%
13	5.10 km	2	42%	4	21%
14	2.94 km	2	42%	4	21%
15	1.70 km	2	42%	4	21%
16	0.98 km	3	32%	5	11%
17	0.57 km	3	32%	5	11%
18	0.33 km	3	32%	5	11%
19	0.19 km	3	32%	5	11%
20	0.11 km	4	21%	6	0%
21	0.06 km	4	21%	6	0%
22	0.04 km	4	21%	6	0%

[a]"Saving" refers to how many digits are omitted, compared to non-adaptive IDs.

The use of adaptive IDs saves space, in particular at lower resolutions. The grid cell containing the Statue of Liberty in New York City, USA, at resolution 9 is, for example, referred to as

5340766511074019041 (non-adaptive ID)

5340767074019 (adaptive 1% ID)

534080740 (adaptive unique ID)

The savings are larger at lower resolutions and become less for higher resolutions. Adaptive 1% IDs are, for example, identical to non-adaptive IDs at a resolution of 20 or higher (Table 3).

The algorithm for determining the ID of a cell is described in Algorithm 1. An exemplary implementation of the algorithm[3] confirms that, in fact, adaptive unique IDs and adaptive 1% IDs stay unique up to resolution 20, and ambiguous IDs do not occur. Higher resolutions have not been tested but can be expected to behave similarly. In case of $q = 6$ decimal places, the algorithm resembles non-adaptive IDs.

6. Determining the grid cell for a given location

The ISEA3H grid, or even other DGGSs, are often used for data aggregation. Instead of relating data to coordinates, as is very common for geographical data, the data are related to grid cells. Such a relation is often determined by the location of an object or a process, as in the case of aggregating the number of houses or trees, but the relation may also be determined by more generic facts. For example data quality may be evaluated for the entirety of the grid cell, without referring to single locations or entities inside the grid cell. In both examples, the data aggregation is of spatial nature, and the spatial extent of objects or processes is aggregated by grid cells.

Whenever a number of locations shall be aggregated by grid cells, it needs to be determined to which grid cell a given location belongs. Grid cells themselves are often represented by their IDs, and it is thus a common task to determine the centroid of the grid cell to which a given location belongs to – the computation of the ID is a rather simple task if the centroid is known. Similar to a regular square grid, the centroid of the ISEA3H grid cell to which a

Algorithm 1. Determine the adaptive ID of a grid cell; in case of $q = 6$ decimal places, the non-adaptive ID is determined

Data: latitude φ and longitude λ of the centroid of the cell; ρ indicates whether the cell is pentagonal; resolution r of the grid; number of decimal places q; precision of the computation ξ

Result: adaptive ID

$\chi \leftarrow \frac{1}{2} \cdot 10^{-q}$
if ρ **then** $A \leftarrow -1$ **else** $A \leftarrow 1$
$B \leftarrow r$
if $\varphi \leq -\chi$ **then** $B \leftarrow B + 22$
if $\lambda \leq -\chi$ **and** $180 - |\lambda| \geq \chi$ **then** $B \leftarrow B + 44$
$C \leftarrow \lfloor (\varphi + \xi) \cdot 10^q \rceil$
$D \leftarrow \lfloor (\lambda + \xi) \cdot 10^q \rceil$
return $A \cdot (B \cdot 10^{2q+5} + C \cdot 10^{q+3} + D)$

given location belongs to can, by and large, be determined by rounding. First, the given coordinates need to be mapped from the sphere to the icosahedron. Secondly, the coordinates of the centroid of the grid cell on the corresponding face of the icosahedron are computed (Algorithm 2). This centroid can be mapped back to the sphere, which is the desired result.

Algorithm 2 computes in a first step the length l of a side of the minimum triangle that completely contains a hexagonal cell at the given resolution of the grid (line 1). In the following, the components of the coordinates on the face of the icosahedron are interchanged in case of even resolution (line 2), because the grid cells expose an alternating orientation for increasing resolution of the grid (Figure 2). After these preparatory steps, potential centroids are computed by rounding the components of the coordinates (lines 3–8). Two cases are distinguished: the case that the coordinates are located in the white area, and the case that they are located in the gray area in Figure 5. If the x coordinate of the location is sufficiently near to the x coordinate of a potential centroid, that is, the coordinates are located in the white area, the coordinates of the corresponding centroid can easily be determined (lines 9–12). Otherwise, there are two potential grid cells to which the coordinates could belong to and hence two potential centroids (lines 14–31). As the grid cells are Voronoi cells on the face of the icosahedron, the original coordinates belong to the grid cell whose centroid is nearer to the coordinates of the location (lines 26–31).

Algorithm 2. Determine the centroid of the grid cell that contains given coordinates

Data: coordinates $c = (x, y)$ on a face of the icosahedron; length l_0 of the sides of the face; resolution r of the grid

Result: centroid of the corresponding grid cell of the icosahedron

1 $l \leftarrow \left(\frac{1}{\sqrt{3}}\right)^{r-1} \cdot l_0$
2 **if** r **is** even **then** $(x, y) \leftarrow (y, x)$
3 $n_{x,\text{center}} \leftarrow \left\lfloor \frac{x}{l/2} \right\rceil$
4 $n_{y,\text{center}} \leftarrow \left\lfloor y \cdot \frac{\sqrt{3}}{l} \right\rceil$
5 $n'_{y,\text{center}} \leftarrow \left\lfloor y \cdot \frac{\sqrt{3}}{l} - \frac{1}{2} \right\rceil$
6 $x_{\text{center}} \leftarrow n_{x,\text{center}} \cdot \frac{l}{2}$
7 $y_{\text{center}} \leftarrow n_{y,\text{center}} \cdot \frac{\sqrt{3}}{l}$
8 $y'_{\text{center}} \leftarrow \left(n'_{y,\text{center}} + \frac{1}{2}\right) \cdot \frac{l}{\sqrt{3}}$
9 **if** $|x - x_{\text{center}}| \leq \frac{l}{6}$ **then**
10   **if** $n_{x,\text{center}}$ **is** odd **then** $y_{\text{center}} \leftarrow y'_{\text{center}}$
11   **if** r **is** odd **then** $(x_{\text{center}}, y_{\text{center}}) \leftarrow (y_{\text{center}}, x_{\text{center}})$
12   **return** $(x_{\text{center}}, y_{\text{center}})$
13 **else**
14   **if** $n_{x,\text{center}}$ **is** even **then**
15         $(y_{\text{center}}, y'_{\text{center}}) \leftarrow (y'_{\text{center}}, y_{\text{center}})$
16   **end**
17   **if** $x > x_{\text{center}}$ **then**
18         $x'_{\text{center}} \leftarrow x_{\text{center}} + \frac{l}{2}$
19   **else**
20         $x'_{\text{center}} \leftarrow x_{\text{center}} - \frac{l}{2}$
21   **end**
22   **if** r **is** odd **then**
23         $(x_{\text{center}}, y_{\text{center}}) \leftarrow (y_{\text{center}}, x_{\text{center}})$
24         $(x'_{\text{center}}, y'_{\text{center}}) \leftarrow (y'_{\text{center}}, x'_{\text{center}})$
25   **end**
26   **if** distance of c to $(x_{\text{center}}, y_{\text{center}}) <$
27   distance of c to $(x'_{\text{center}}, y'_{\text{center}})$ **then**
28   **return** $(x_{\text{center}}, y_{\text{center}})$
29   **else**
30   **return** $(x'_{\text{center}}, y'_{\text{center}})$
31   **end**
32 **end**

7. Inferring cell geometries from identifiers

A typical scenario for the use of a DGGS, including data aggregated by DGGSs, is the following: Data or measurement values are collected and aggregated by cells. The resulting data is encoded in a suitable way, possibly cached, and then delivered to a client application, for example, a web client. The web client visualizes the cells including the aggregated data, often enriched by a map background. For such a scenario, it is important to encode the information as short as possibly and such that it can easily be decoded.

Only the centroid needs to be communicated to the client, because the boundary of a cell can approximately be reconstructed by knowledge of the centroid of the cell and the ones of adjacent cells. The grid is uniform on each face of the icosahedron. The hexagonal cells are regular, their edges are straight line segments of equal length, and the angles between two adjacent edges are all equal. The boundary of the cells are, accordingly, Voronoi cells on a face of the icosahedron. The ISEA projection, however, does not map straight line segments to geodesic line segments, and it changes their length dependent on their location. Accordingly, the boundary of a grid cell on the sphere does, in general, not consist of geodesic line segments, and it is not a Voronoi cell. As a cell practically restricts to only a small part of the sphere and as the distortion is locally of uniform nature, the boundary of a grid cell can practically be seen as being identical to the corresponding Voronoi cell. The difference between the grid cell and the corresponding Voronoi cell becomes smaller for higher resolutions, because the approximation by a local coordinate system improves.

The computation of the (approximate) grid cells can be executed in two steps. Assume that a number of centroids C_i in a certain area are transferred to the client. In a first step, the client needs to check for every centroid whether all its neighbors are in the transferred data, because the Voronoi cells approximate the grid cells only in this case (Figure 6a). Tree data structures like a vantage-point tree can be used to efficiently find neighboring centroids. Vantage-point trees take particular advantage of the triangle inequality (Bozkaya & Ozsoyoglu, 1999; Uhlmann, 1991), which is possible because the distance on the sphere is a metric for sufficiently small regions. These trees can be used to find the six or five neighbors of a centroid, depending on whether the cell corresponding to the centroid is a hexagon or a pentagon. Having found these neighbors, it needs to be checked which of these pairs of centroids and neighbors correspond to grid cells, and which of the centroids are located near the border of the considered area and are thus not similar to regular hexagons respective pentagons (Figure 6c). As an example for such an approach that works well also for non-regular hexagons on the surface of the sphere, the angles formed by the rays from the centroid of the cell to the centroids of neighboring cells (Figure 6d). It can be assumed that two opposing angles α_i and $\alpha_{i+3 \bmod 6}$ are more similar than an angle α_i is to one of the two neighboring angles of the opposing angle, that is, to $\alpha_{i+2 \bmod 6}$ or $\alpha_{i+4 \bmod 6}$. Otherwise, the centroid would be non-symmetrically distorted. As the grid cells are, by and large, symmetric, such a cell with a non-symmetrically distorted geometry can be assumed to be near the border of the considered area – the centroids of the adjacent grid cells are not included in the transferred data and too little is known to infer the geometry of the cell – and may thus be filtered away.

In a second step, an estimation of the boundary is computed for each of the grid cells. As the projection does not map straight line segments on the icosahedron to geodesic line segments on the sphere, the centroids are theoretically insufficient for computing the location of the vertices of the cell. As a close approximation for medium and higher resolutions in case of the depiction on a map, one may though practically assume that the sides are even straight line segments on the map surface, and that the grid cells are, in fact, Voronoi cells. In this case, the vertices of the hexagonal boundary can be computed as the geometric center of three different centroids C_i, C_j, and C_k (Figure 6b).

The data to be transferred from the server to the client can even be further reduced by using adaptive IDs. In case that the resolution is smaller or equal than 19, adaptive 1% IDs can be used, yielding shorter IDs and thus reducing the amount of data to communicate. In both cases, non-adaptive and adaptive IDs, there is no need to transfer IDs and coordinate values independently, because the values of latitude and longitude can easily be inferred with a suitable precision from the ID of a grid cell.

8. Advantages and limitations of the identifier scheme

The identifier scheme that has been introduced in this article aims at being easy to understand, and it aims for being efficient when the ID and the geometry both have to be represented. Such advantages come at the cost of longer IDs (compared to other ID schemes that do not include any geometric information) and slightly inaccurate geometries. In this section, we critically discuss the merits and limitations of the introduced identifier scheme by setting the identifier scheme into context.

Identifier schemes serve for several purposes, among them the purpose of a formal representation of a cell and, of a human-readable representation. While the former purpose is beyond dispute, the latter one is of differing importance in different use cases. The importance of human-readable IDs is though increasing, which manifests, among others, in the increasing attention that is paid to human readability. For instance, Adams (2017) discusses the importance of simple IDs in the context of the semantic web with the aim to introduce a global gazetteer, called *Wāhi*. The grid system *what3words* is another example with very easy-to-understand and easy-to-remember IDs. In fact, the system assigns three different words to each cell of about 9 m^2 size. The ID "chip.twice.update", for example, refers to the Statue of Liberty in New York City, USA. In contrast to such identifier schemes, many schemes have, in case of the ISEA3H grid, been introduced that serve solely for the

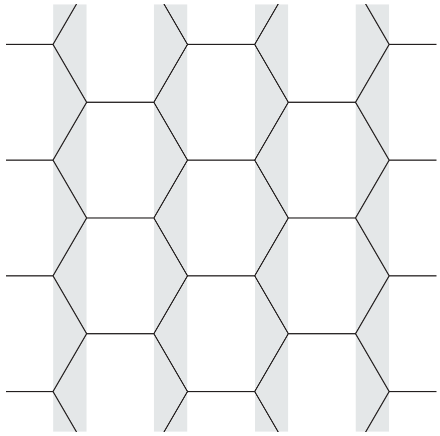


Figure 5. Determining the grid cell for given coordinates. In the white areas, determining the centroid of the corresponding grid cell is only rounding, whereas in the gray areas simple computations are necessary.

purpose of being a formal machine-readable presentation. Among these schemes are such that refer to space-filling curves (Bai & Zhao, 2004; Bartholdi & Goldsman, 2001; White, 2000), or identifier schemes that use two (Mahdavi-Amiri et al., 2013, 2015; Sahr, 2008) or three integers (Ben et al., 2010; Vince, 2006b).

The different types of identifier schemes differ in their advantages. While systems like what3words are easy to remember, they do not offer any idea of whether two cells are neighboring or even adjacent. The same applies to many identifier schemes in case of the ISEA3H grid, for example, to space-filling curves. Identifier schemes that rely on a two-dimensional index on the icosahedron allow for an intuitive understanding of whether two cells are neighboring or even adjacent. At the edges of the icosahedron, this is, however, not any longer possible because the schemes of the two adjacent faces meet. In case of the identifier scheme introduced in this article, a comparison of two identifiers can determine whether the corresponding cells are located near to each other. It is though not obvious whether these cells are

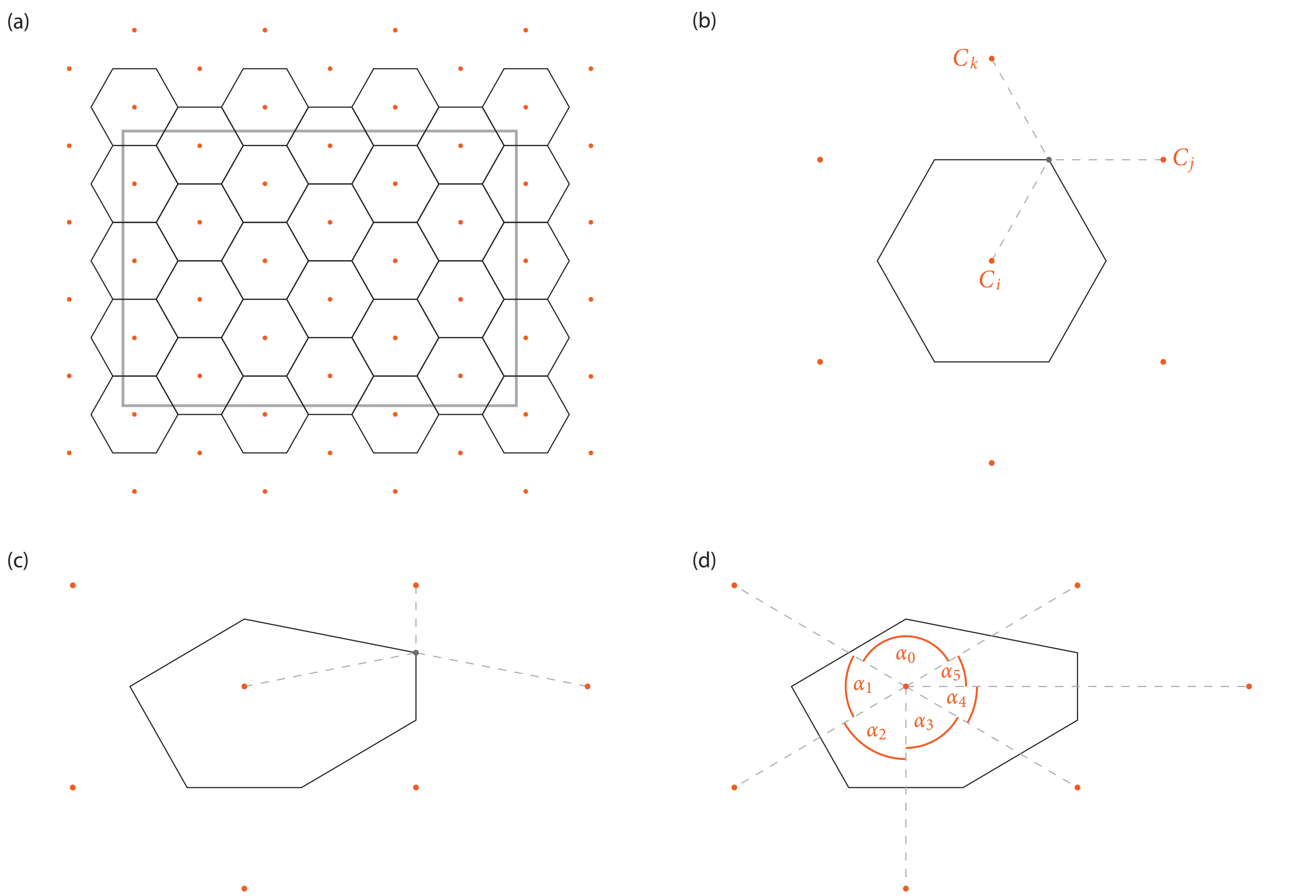


Figure 6. Construction of the boundary of hexagonal cells by their centroids. (a) The cells can be constructed by their centroids but for a given bounding box, a buffer around the bounding box needs to be considered. (b) The gray vertex of the hexagon can be computed as the geometric center of the centroids of adjacent cells. (c) If the centroid of an adjacent cell is missing (and another one is used), the resulting hexagon is not regular and does not correspond to a grid cell. (d) The angles α_i reveal whether there is a centroid of an adjacent cell missing.

even adjacent. As the angular distortion and the scale variation are low – 17.27° and 16.3%, respectively – one can compute whether two cells are adjacent by comparing the distance between their centroids to the typical distance between adjacent cells at the given resolution.

Identifiers are particularly short if they do not provide any intuitive idea of spatial proximity. An enumeration by a space-filling curve – this is more or less a simple enumeration – would allow the representation of the IDs by unsigned 39-bit integers at resolution 22. In comparison, the identifier scheme introduced in this article would presume a signed 64-bit integer, which is about 60% longer than in case of a space-filling curve but still allows for a native representation in most programming languages on most machines. At resolution 9, a simple enumeration would require an unsigned 18-bit integer, while adaptive unique IDs would require a signed 30-bit integer. The larger memory footprint of both the non-adaptive and the adaptive IDs is due to two factors: the ID includes the centroid of the cell (and the geometry if several IDs of a neighborhood are known), and the ID follows a simple scheme that is more human-readable. The latter effect, in particular, provides the possibility to easily identify whether two cells are in the same vicinity. If the cells need to be uniquely identified only without being human-readable and without their geometry being of importance, a simple enumeration of the cells can save memory. If the cells shall though be uniquely identified and stored or communicated in combination with their geometry, for example, when data is aggregated by the ISEA3H grid and then sent from a server to a client with the aim of visualizing the data, only the non-adaptive or adaptive IDs need to be communicated. In fact, the independent communication of the IDs and the geometries becomes obsolete, because the geometry can easily be reconstructed from the IDs, as has been discussed in Section 7. The IDs require only some more bits compared to the coordinates of the centroid, and the communication becomes more efficient because the IDs combining an identifier and the coordinates of the centroid result in a much shorter overall representation.

The presented method for inferring cell geometries from the centroids of neighboring cells is not exact but only an approximation (Section 7). For understanding the relevance of this approximative nature, the context in which the geometries are used needs to be understood. There are two important purposes for which the geometry of a cell is important: the purpose of determining the grid cell for a given location on Earth, and when displaying grid cells in a map. The former purpose has been discussed in Section 6, and an algorithm has been provided to determine the grid cell without any explicit projection of the cell geometry to a map. Instead, the cell is determined on the icosahedron, without any need for an approximation. For the latter purpose, the geometries of a number of grid cells need to be computed for displaying them on the map. In many cases, only the vertices of a grid cell are projected to a map, but the straight line segments between these vertices are not. Instead, one assumes these line segments to be approximated by straight lines in the coordinate system of the map. This assumption is actually valid for gnomonic map projections, but leads, in the example of the Mercator projection, only to an approximation. For most use cases, the approximation is good enough for higher resolutions of the grid. This approach is not new but rather widely used. We extend this approach by also determining the vertices of a grid cell as the geometric center of the centroid of the cell and of two adjacent cells. While this approach is valid on a face of the icosahedron, it is only an approximation if performed on the map surface. The difference between the correct geometry and the approximated geometry is due to the distortion of the map.

Another drawback of the introduced identifier scheme is the maximum resolution of 22. This limitation has its origin in the representation of both, the resolution of the grid and the signs of latitude and longitude in part B, that is, in the first two digits of the ID. If one would allow the identifier scheme to be used at resolution 23, part B would either become ambiguous or the ID would exceed the range of a signed 64-bit integer. This limitation is merely technical: the scheme can easily be extended to higher resolutions by allowing three digits in part B and a higher precision in parts C and D. We decided against introducing a generalized variant of the identifier scheme due to several reasons. First, 64-bit integers are available on most machines, while larger integers are not in many programming languages. Secondly, grid cells have a diameter of less than 50 m and an area of 1625 m^2 at resolution 22, which is lower than usually needed in most *global* use cases. In addition, there are more than 100 billion cells at this resolution, which will challenge many machines both in terms of computation power as well as in memory capacity if being used on a global scale. If higher resolutions are though needed, the scheme can easily be extended, as has been described above.

Two major algorithms have been provided – one for determining the grid cell for a given location (Section 6), and one for inferring cell geometries from identifiers (Section 7). When the grid cell for a given location on Earth shall be computed, there is no way to avoid either the projection of the location to the icosahedron, or the projection of the geometry of the grid cell to the sphere. The algorithm we investigated in Section 6 uses the former strategy, followed by a rounding of numbers. While the computation can be improved in detail, no considerable speed-up can be expected. Actually, the projection is by far the slowest part of the computation and cannot be avoided. If the latter strategy is used, the geometry of the grid cells need to be projected to the sphere, which requires even more projections – the geometries can though be cached in order to assign a large number of locations to the corresponding cells.

The method for inferring cell geometries for given identifiers is very efficient, because for each cell, only one location on the icosahedron needs to be projected to the sphere and subsequently to the map surface. In fact, the approximation of the geometry uses only the centroids of neighboring cells. Accordingly, only the centroids have to be projected from the icosahedron to the map surface, which is already done when computing the IDs of the cells. If the IDs of neighboring cells have already been determined, the approximation of the cell geometries does not require any additional projection from the icosahedron to the sphere but only from the sphere to the map surface. This is very different to many other algorithms that explicitly project every vertex of each cell to the map surface. As every hexagonal cell has six vertices, which are each shared by three neighboring cells, this approach would require two times as many locations on the icosahedron to be projected to the map surface compared to the method presented in Section 7. It needs to be taken into consideration that both methods, the one presented in this paper and a projection of all vertices to the map surface, require some more vertices to be projected at the border of the considered area. In summary, the algorithms provided in this article are at least as efficient as the ones that are broadly used, and performance improvements seem to be possible only in detail, for example, by improving the speed of the projection from the icosahedron to the sphere itself.

9. Summary and outlook

The ISEA3H grid is very common but existing identifier schemes suffer from a complexity that is not required for many applications. In this article, we have introduced a novel identifier scheme that is easy to understand for non-experts, because it encodes the geographic coordinates of the centroid. This comes at the cost of potentially non-unique identifiers at higher grid resolutions. We have, however, demonstrated that this issue is not decisive for real-world applications because it can be expected to only occur for grid cells with a diameter below 50 m. Furthermore, we also have introduced adaptive IDs that implicitly contain the coordinate values only with limited precision, depending on the resolution of the grid and the purpose. For most real applications, it is sufficient to use adaptive IDs instead of the non-adaptive ones. Strategies to determine the grid cell for a given pair of coordinates – this is of particular interest for the aggregation of data – as well as to infer the geometry of a cell only from its identifier and the identifiers of adjacent cells have been discussed. The corresponding algorithms are efficient, and the algorithm for the latter case can be executed in real time.

Further improvements of exact algorithms related to the ISEA projection and the ISEA3H grid seem hardly possible, because these algorithms have been discussed in detail in the existing literature. Approximate algorithms with better performance have been discussed in the literature, and further approximations might be approached in the future. A more thorough understanding of such approximations is though necessary to keep the identifiers that were introduced in this article unique, despite the errors introduced by the approximations.

The computation of all cells inside a bounding box is a very common task, which we yet did not discuss. There is a straightforward but inefficient solution: a cell inside the bounding box needs to be found, and all neighboring cells inside the bounding box are added. This strategy requires to compute the centroids of the cells on the icosahedron and their projection to the sphere. Future research might explore ways to avoid the multiplicity of computations necessary to apply the projection to different coordinates. Instead, one might compute only a certain number of centroids and interpolate all centroids in between. If this issue has successfully been resolved, fewer computations would be necessary on the server, the amount of data to communicate to a client would be further reduced, and possibly less computations would be needed on the client.

The novel identifier scheme introduced in this article compensates for some disadvantages of existing identifier schemes. Limitations and issues related to the novel identifier scheme have been discussed, among them, the issues of computational accuracy and rounding errors. Can strategies be found to compensate for these issues in respect to the novel identifier scheme? If not, which identifier scheme would compensate for these disadvantages without rendering the scheme less intuitive? It has been discussed that the novel scheme may not be suitable to communicate identifiers across different implementations or machines. Can this issue be resolved by making the novel scheme less prone to the aforenamed issues of computational accuracy and rounding errors? Or can the existing scheme even be proven to be insusceptible when statistically examined?

Notes

1. If a sphere would be tessellated by n hexagons, there would be $2n$ vertices (each hexagon has six vertices, which are shared among three hexagons), and $3n$ edges (each hexagon has six edges, which are shared among two hexagons). The Euler characteristics of the grid is thus $2n - 3n + n = 0$, which contradicts the Euler characteristics of a sphere being 2.
2. http://github.com/giscience/geogrid.

3. http://github.com/giscience/geogrid.

Acknowledgments

The author is grateful to Rafael Troilo for his technical support in running software on a high-performance server.

Disclosure statement

The author declares that he has no competing interests.

Funding

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) project "A framework for measuring the fitness for purpose of OpenStreetMap data based on intrinsic quality indicators" [Grant Number FA 1189/3-1].

ORCID

Franz-Benjamin Mocnik http://orcid.org/0000-0002-1759-6336

References

Adams, B. (2017). Wāhi, a discrete global grid gazetteer built using linked open data. *International Journal of Digital Earth*, *10*(5), 490–503. doi:10.1080/17538947.2016.1229819

Bai, J., & Zhao, X. (2004). Quadtree-based indexing of hierarchical diamond subdivisions of the global. In *IGARSS 2004. 2004 Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2878–2881. Piscataway, NJ: IEEE. doi: 10.1109/IGARSS.2004.1370294

Bartholdi, J. J., III, & Goldsman, P. (2001). Continuous indexing of hierarchical subdivisions of the globe. *International Journal of Geographical Information Science*, *15*(6), 489–522. doi:10.1080/13658810110043603

Battersby, S. E., Strebe, D., & Finn, M. P. (2016). Shapes on a plane: Evaluating the impact of projection distortion on spatial binning. *Cartography and Geographic Information Science*, *44*(5), 410–421. doi:10.1080/15230406.2016.1180263

Ben, J., Ton, X., & Chen, R. (2010). A spatial indexing method for the hexagon discrete global grid system. *Proceedings of the 18th International Conference on Geoinformatics*. doi: 10.1109/GEOINFORMATICS.2010.5567972

Birch, C. P. D., Oom, S. P., & Beecham, J. A. (2007). Rectangular and hexagonal grids used for observation, experiment and simulation in ecology. *Ecological Modelling*, *206*(3–4), 347–359. doi:10.1016/j.ecolmodel.2007.03.041

Bozkaya, T., & Ozsoyoglu, M. (1999). Indexing large metric spaces for similarity search queries. *ACM Transactions on Database Systems*, *24*(3), 361–404. doi:10.1145/328939.328959

Brettschneider, B. (2008). Climatological hurricane landfall probability for the United States. *Journal of Applied Meteorology and Climatology*, *47*, 704–716. doi:10.1175/2007JAMC1711.1

Carr, D., Kahn, R., Sahr, K., & Olsen, T. (1997). ISEA discrete global grids. *Statistical Computing and Statistical Graphics Newsletter*, *8*(2–3), 31–39.

Defense Mapping Agency. (1989). *The universal grids: Universal transverse Mercator (UTM) and universal polar stereographic (UPS)*. (DMA Technical Manual 8358.2). Retrieved from http://earth-info.nga.mil/GandG/publications/tm8358.2/TM8358_2.pdf.

Dulvy, N. K., Fowler, S. L., Musick, J. A., Cavanagh, R. D., Kyne, P. M., Harrison, L. R., … White, W. T. (2014). Extinction risk and conservation of the world's sharks and rays. *eLife*, *3*, e00590. doi: 10.7554/eLife.00590

Dutton, G. (1996). Encoding and handling geospatial data with hierarchical triangular meshes. *Proceedings of the 7th International Symposium on Spatial Data Handling (SDH)*, 15–28. https://www.oicrf.org/documents/40950/43224/Encoding+and+handling+geospatial+data+with+hierarchical+triangular+meshes.pdf/b6ce86cf-5cd1-1e37-0cef-1ec1c08afc29

Elsner, J. B., Hodges, R. E., & Jagger, T. H. (2012). Spatial grids for hurricane climate research. *Climate Dynamics*, *39*(1–2), 21–36. doi:10.1007/s00382-011-1066-5

Elsner, J. B., & Jagger, T. H. (2010). On the increasing intensity of the strongest Atlantic hurricanes. In J. B. Elsner, R. E. Hodges, J. Malmstadt, & K. Scheitlin (Eds.), *Hurricanes and climate change* (Vol. 2, pp. 175–190). Dordrecht, Netherlands: Springer. doi:10.1007/978-90-481-9510-7_10.

Gevrey, M., Worner, S., Kasabov, N., Pitt, J., & Giraudel, J.-L. (2006). Estimating risk of events using SOM models: A case study on invasive species establishment. *Ecological Modelling*, *197*(3–4), 361–372. doi:10.1016/j.ecolmodel.2006.03.032

Harrison, E., Mahdavi-Amiri, A., & Samavati, F. (2012). Analysis of inverse Snyder optimizations. In M. L. Gavrilova, & C.J.K. Tan (Eds), *Transactions on Computational Science*, *16*, (pp. 134–148). Berlin: Springer. doi: 10.1007/978-3-642-32663-9_8

Kidd, R. (2005). *Discrete global grid systems (ASCAT Soil mosture report series no. 4)*. Vienna: Institute of Photogrammetry and Remote Sensing, Vienna University of Technology.

Lin, B., Zhou, L., Xu, D., Zhu, A.-X., & Lu, G. (2017). A discrete global grid system for earth system modelling. *International Journal of Geographical Information Systems*, *32*(4), 711–737. doi:10.1080/13658816.2017.1391389

Lucas, D., & Gibson, L. (1991). Template decomposition and inversion over hexagonally sampled images. *Proceedings of the SPIE Conference on Image Algebra and Morphological Image Processing II*, 157–163. doi:10.1117/12.46112

Mahdavi-Amiri, A., Bhojani, F., & Samavati, F. (2013). One-to-two digital Earth. In G. Bebis et al. (Eds), *Advances in Visual Computing. ISVC 2013*, (pp.681–692). Berlin: Springer. doi:10.1007/978-3-642-41939-3_67

Mahdavi-Amiri, A., Harrison, E., & Samavati, F. (2015). Hexagonal connectivity maps for Digital Earth. *International Journal of Digital Earth*, *8*(9), 750–769. doi:10.1080/17538947.2014.927597

Mahdavi-Amiri, A., Samavati, F., & Peterson, P. (2015). Categorization and conversions for indexing methods of discrete global grid systems. *ISPRS International Journal of GeoInformation*, *4*(1), 320–336. doi:10.3390/ijgi4010320

National Geospatial-Intelligence Agency. (2014). *The universal grids and the transverse Mercator and polar stereographic map projections*. (Revision of DMA Technical Manual

8358.2 dated 18 September 1989). Retrieved from http://earth-info.nga.mil/GandG/publications/NGA_SIG_0012_2_0_0_UTMUPS/NGA.SIG.0012_2.0.0_UTMUPS.pdf

Nguyen, H., Cressie, N., & Bravermann, A. (2012). Spatial statistical data fusion for remote sensing applications. *Journal of the American Statistical Association*, *107*(499), 1004–1018. doi:10.1080/01621459.2012.694717

Nguyen, H., Katzfuss, M., Cressie, N., & Bravermann, A. (2014). Spatio-temporal data fusion for very large remote sensing datasets. *Technometrics*, *56*(2), 174–185. doi:10.1080/00401706.2013.831774

Openshaw, S., & Taylor, P. J. (1979). A million or so correlation coefficients: Three experiments on the modifiable areal unit problem. In N. Wrigley (Ed.), *Statistical applications in the spatial sciences* (pp. 127–144). London: Pion.

Park, Y.-S., Lek, S., Scardi, M., Verdonschot, P. F. M., & Jørgensen, S. E. (2006). Patterning exergy of benthic macro-invertebrate communities using self-organizing maps. *Ecological Modelling*, *195*(1–2), 105–113. doi:10.1016/j.ecolmodel.2005.11.027

Purss, M., Gibb, R., Samavati, F., Peterson, P., Rogers, J. A., Ben, J., & Dow, C. (2017). Discrete global grid systems abstract specification (Tech. Rep. No. 15-104r5). Open Geospatial Consortium (OGC). Retrieved from http://docs.opengeospatial.org/as/15-104r5/15-104r5.html

Roick, O., Hagenauer, J., & Zipf, A. (2011). OSMatrix – grid-based analysis and visualization of OpenStreetMap. *Proceedings of the 1st European State of the Map Conference (SOTM-EU)*, Vienna, Austria.

Roick, O., Loos, L., & Zipf, A. (2012). A technical framework for visualizing spatio-temporal quality metrics of volunteered geographic information. *Proceeding of Geoinformatik 2012*, Braunschweig. Germany. Retrieved from http://www.geog.uni-heidelberg.de/md/chemgeo/geog/gis/roick_osmatrix_geoinformatik2012.pdf

Sahr, K. (2008). Location coding on icosahedral aperture 3 hexagon discrete global grids. *Computers, Environment and Urban Systems*, *32*(3), 174–187. doi:10.1016/j.compenvurbsys.2007.11.005

Sahr, K. (2017). DGGRID software. Retrieved December 23, 2017 from http://www.discreteglobalgrids.org/software

Sahr, K., White, D., & Kimerling, A. J. (2003). Geodesic discrete global grid systems. *Cartography and Geographic Information Science*, *30*(2), 121–134. doi:10.1559/152304003100011090

Schipper, J., Chanson, J. S., Chiozza, F., Cox, N. A., Hoffmann, M., Katariya, V., … Young, B. E. (2008). The status of the world's land and marine mammals: Diversity, threat, and knowledge. *Science*, *322*(5899), 225–230. doi:10.1126/science.1165115

Snyder, J. P. (1992). An equal-area map projection for polyhedral globes. *Cartographica*, *29*(1), 10–21. doi:10.3138/27H7-8K88-4882-1752

Strassburg, B. B. N., Kelly, A., Balmford, A., Davies, R. G., Gibbs, H. K., Lovett, A., … Rodrigues, A. S. L. (2010). Global congruence of carbon storage and biodiversity in terrestrial ecosystems. *Conservation Letters*, 3(2), 98–105. doi:10.1111/j.1755-263X.2009.00092.x

Tong, X., Ben, J., Wang, Y., Zhang, Y., & Pei, T. (2013). Efficient encoding and spatial operation scheme for aperture 4 hexagonal discrete global grid system. *International Journal of Geographical Information Science*, *27*(5), 898–921. doi:10.1080/13658816.2012.725474

Uhlmann, J. K. (1991). Satisfying general proximity/similarity queries with metric trees. *Information Processing Letters*, *40*(4), 175–179. doi:10.1016/0020-0190(91)90074-R

Vince, A. (2006a). Indexing a discrete global grid. *Proceedings of the 2nd Annual International Conference on Computing and ICT Research (SREC)*, 2–27.

Vince, A. (2006b). Indexing the aperture 3 hexagonal discrete global grid. *Journal of Visual Communication and Image Representation*, *17*(6), 1227–1236. doi:10.1016/j.jvcir.2006.04.003

Vince, A., & Zheng, X. (2009). Arithmetic and Fourier transform for the PYXIS multi-resolution digital earth model. *International Journal of Digital Earth*, *2*(1), 59–79. doi:10.1080/17538940802657694

White, D. (2000). Global grids from recursive diamond subdivisions of the surface of an octahedron or icosahedron. *Environmental Monitoring and Assessment*, *64*(1), 93–103. doi:10.1023/A:1006407023786

Yfantis, E. A., Flatman, G. T., & Behar, J. V. (1987). Efficiency of kriging estimation for square, triangular, and hexagonal grids. *Mathematical Geology*, *19*(3), 183–205. doi:10.1007/BF00897746