

## Research Article

# Research and Construction of a Global Hexagonal Marine Gravity Gradient Reference Map for Navigation

Huang Yan <sup>1</sup>, Li Shanshan,<sup>1</sup> Zhou Chuyuan,<sup>2</sup> Li Xinxing,<sup>1</sup> Lv Minghao <sup>3</sup>, Fan Diao,<sup>1</sup> Tan Xuli,<sup>1</sup> and Wang Aoming<sup>1</sup>

<sup>1</sup>Institute of Surveying and Mapping, PLA Strategic Support Force Information Engineering University, China

<sup>2</sup>Shenzhen University, China

<sup>3</sup>96941 Troops, China

Correspondence should be addressed to Lv Minghao; [eg669fj@163.com](mailto:eg669fj@163.com)

Received 7 May 2022; Revised 27 June 2022; Accepted 4 July 2022; Published 23 March 2023

Academic Editor: Hao Wu

Copyright © 2023 Huang Yan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A high-precision marine gravity gradient reference map is key to enabling underwater gravity gradient matching navigation. At present, the construction of the reference maps is based on quadrilateral geographic grids. However, quadrilateral grids lead to detriangulation at high latitudes, which limits the global applicability of such maps to underwater gravity navigation. To circumvent the limitations of quadrilateral grids, a hexagonal grid is introduced for constructing the reference map. This paper analyzes the characteristics of the icosahedral Snyder equal area aperture 4 hexagon (ISEA4H) and H3 grid systems and selects an appropriate grid system. In addition, we calculate and analyze the grid model errors and matching positioning errors of hexagonal and quadrilateral grids at the same resolution. The experimental results show that the grid model and matching positioning errors of a hexagonal grid system are more than 14% and 15% less than those of a quadrilateral grid system, respectively, indicating the feasibility and effectiveness of applying hexagonal grids to gravity gradient matching navigation. Given the low construction efficiency of a marine hexagonal grid gravity gradient reference map, we propose an efficient CPU+GPU hybrid parallel scheme. A global total tensor marine hexagonal grid gravity gradient reference map model is then constructed.

## 1. Introduction

Given China's strategy to become a maritime power and the proposed 14th five-year plan for future ocean and deep-sea tasks, underwater vehicles have become a development focus of the national marine strategy, which plays an indispensable role in both military and civil fields in China. How to realize the long-term, autonomous, and high-precision positioning and navigation of submarines has become the research focus of multiple scholars. Inertial navigation systems (INSs) are often used in the passive positioning and navigation of underwater vehicles; however, the navigation error of INS systems cannot be independently eliminated and gradually accumulates with time. Therefore, the navigation accuracy of passive positioning and navigation using only an INS system is greatly reduced over long periods of time. This has spurred the development of underwater gravity matching

navigation technology. The goal of such technology is to use the gravity sensor carried by underwater vehicles to perform real-time gravity measurements and then to match these measurements with a high-precision gravity map using a relevant algorithm, so as to correct the positioning information and component errors of the INS system. As the second-order differential of the gravity potential, measurements of the gravity gradient are not affected by the acceleration of the underwater vehicle carrier. The gravity gradient has five independent observational components. Therefore, compared with gravity anomaly matching navigation, gravity gradient matching navigation can obtain higher precision positioning and navigation information [1, 2].

The construction of a marine gravity gradient reference map is a prerequisite for the realization of gravity gradient-assisted INS navigation; such a map would be stored in a geographic grid composed of longitude and latitude in the



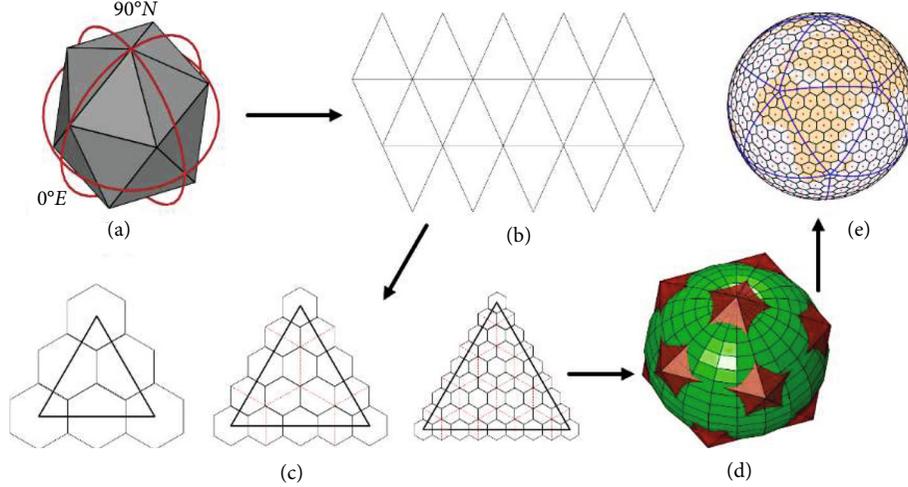


FIGURE 2: Hexagonal grid generation process.

TABLE 1: Parameter statistics of the icosahedral Snyder equal area aperture 4 hexagon (ISEA4H) grid (with a spherical radius of  $R = 6378.136$  km).

Level ( $L$ )	Number of grids	Grid area ( $\text{km}^2$ )	Grid spacing (km)
0	12	$4.260 \times 10^7$	$7.530 \times 10^3$
1	42	$1.217 \times 10^7$	$3.682 \times 10^3$
2	162	$3.156 \times 10^6$	$1.968 \times 10^3$
3	642	$7.963 \times 10^5$	$1.002 \times 10^3$
4	2,562	$1.995 \times 10^5$	$5.035 \times 10^2$
5	10,242	$4.991 \times 10^4$	$2.520 \times 10^2$
6	40,962	$1.248 \times 10^4$	$1.260 \times 10^2$
7	163,842	$3.120 \times 10^3$	$6.303 \times 10$
8	655,362	$7.800 \times 10^2$	$3.152 \times 10$
9	2,621,442	$1.950 \times 10^2$	$1.576 \times 10$
10	10,485,762	$4.875 \times 10$	7.879
11	41,943,042	12.161	3.767
12	167,772,162	3.040	1.884
13	671,088,642	0.760	0.942
14	2,684,354,562	0.190	0.471
$n$	$10 \times 4^n + 2$	$\frac{\sqrt{3}}{2^{2n+1}} L^2$	$4R \cdot \arcsin \left( \frac{1}{2^n} \sqrt{\frac{\sqrt{3}}{2\pi}} \cdot \frac{L}{2R} \right)$

Note:  $L$  represents the hexagonal grid level, and  $R$  represents the average radius of the Earth.

planes expanded in land regions are broken as little as possible. Therefore, a vertex is not located at the pole but at (5.2454°E, 2.3009°N) and the azimuth of the adjacent vertices is fixed at 7.46658°. The H3 system is composed of 16 levels of hexagonal grids. The number of grid cells in each level of the grid system and the grid information are given in Table 2.

2.3. *Hexagonal Grid System Selection.* Even though the ISEA4H and H3 grid systems both use hexagonal grids as their units, there are some differences between them.

- (1) The ISEA4H grid system adopts the four-aperture division method; that is, the number of grids between adjacent levels increases by four times (as shown in Figure 3(a)). The H3 system adopts the seven-aperture division method; that is, the number of grids between adjacent levels increases by seven times (as shown in Figure 3(b))
- (2) The ISEA4H grid system adopts the Schneider equal area projection, and the area of the hexagonal elements in the grid system is basically the same after splitting. To ensure rapid indexing between levels, the H3 system sacrifices the equal area, and its hexagonal grid unit area is not consistent
- (3) The ISEA4H grid system adopts a regular icosahedral projection. The two vertices of the icosahedron are located at the North and South Poles, and the edge passing through the North Pole is coplanar with the central meridian. Therefore, the ISEA4H grid system is relatively regular and has a stronger correlation with the South and North Poles of the Earth and longitude and latitude. The H3 system adopts a positioning similar to a map. To minimize the splitting of land regions when expanded into a plane, the positioning point deviates from the pole
- (4) In terms of the coding index and management at all levels, the H3 system is more flexible than the ISEA4H system and can quickly index grids at different levels

Based on the above comparison of the advantages and disadvantages of the two grid systems and the actual requirements of marine gravity gradient matching navigation, we selected the ISEA4H grid system as the basis to construct

the marine gravity gradient reference map. The main reasons are as follows. (1) The ISEA4H grid system has a good characteristic of equal area, and the grid size and shape are relatively consistent; this is conducive to the solution process of the gravity matching navigation algorithm. Conversely, the H3 system grid has a poor characteristic of equal area; therefore, the error caused by the inconsistent grid size needs to be compensated for in the solution process of the gravity matching algorithm. (2) Even though the H3 system has less segmentation in the land regions, this does not have a substantive effect on the construction of a marine gravity gradient reference map and projection positioning of the H3 grid has little correlation with the geographical poles and the central meridian, which is not conducive to solving the gravity matching navigation algorithm. (3) The ISEA4H grid system is a four-aperture grid system; that is, the number of grids at each level increases by four times and the average grid distance is reduced to 1/2. Meanwhile, the H3 system adopts a seven-aperture grid system; that is, the number of grids at each level is increased by seven times and the average grid distance is reduced by 0.378 times. The resolutions of the grids at all levels of the H3 system therefore vary greatly. For example, on the sixth level of the H3 system, the average side length of the grid is 3.23 km and the grid area is 36.13 km<sup>2</sup>. The number of grids per unit area is equivalent to that of a 3' 20" geographic grid, but the number of grids per unit area for the next level, that is, the seventh level, is equivalent to that of a 75" geographic grid. Because of the large variations between the levels, the corresponding resolution does not actually match the quadrilateral grid resolution used in navigation. The 12th level of the ISEA4H grid system has a grid spacing of 1.884 km and a grid area of 3.040 km<sup>2</sup>. The number of grids per unit area is therefore equivalent to the 1'-resolution geographic grid currently used in gravity matching navigation, making the ISEA4H grid system a more suitable grid division method for the construction of a gravity gradient reference map.

*2.4. Error Analysis of the Grid Geometry for Matching Navigation and Positioning.* Here, we analyze the effectiveness of a reference map based on the hexagonal grid unit for gravity matching navigation. Then, without considering the measurement error of the gravity/gravity gradiometer, the measurement error of the inertial components, or errors in the gravity/gravity gradient reference map, we examine the error caused by the geometric structure of the reference map grid in the process of discretizing the track. Using the inertial navigation solution program and the gravity gradient matching navigation program, we calculate and analyze the hexagonal grid model error and the grid matching positioning error, respectively. The experimental area is a marine area with range of 10° × 10° (from 19°N to 29°N and from 41°W to 51°W). The inertial navigation track simulation program is used to simulate the track and measurement data of an underwater vehicle. The inertial navigation sampling rate is 100 Hz. The simulation data include the output of the inertial components (the angular and velocity increments),

TABLE 2: Parameter statistics of the H3 grid (with a spherical radius of  $R = 6378.136$  km).

Level ( $L$ )	Number of grid	Area of grid (km <sup>2</sup> )	Average side length (km)
0	122	4,250,546.8477000	1,107.712591000
1	842	607,220.9782429	418.676005500
2	5,882	86,745.8540347	158.244655800
3	41,162	12,392.2648621	59.810857940
4	288,122	1,770.3235517	22.606379400
5	2,016,842	252.9033645	8.544408276
6	14,117,882	36.1290521	3.229482772
7	98,825,162	5.1612932	1.220629759
8	691,776,122	0.7373276	0.461354684
9	4,842,432,842	0.1053325	0.174375668
10	33,897,029,882	0.0150475	0.065907807
11	237,279,209,162	0.0021496	0.024910561
12	1,660,954,464,122	0.0003071	0.009415526
13	11,626,681,248,842	0.0000439	0.003559893
14	81,386,768,741,882	0.0000063	0.001348575
15	569,707,381,193,162	0.0000009	0.000509713

Note: the information in this table is taken from the Internet (<https://h3geo.org/>).

the gravity gradient observation data, and the real track of the underwater vehicle. The error settings of the inertial components are listed in Table 3.

We used the simulation program to generate two simulated tracks as experimental objects; the navigation speed was set to 10 nautical miles per hour, the navigation time of track 1 was 93,453 s (approximately 25.96 h), and the navigation time of track 2 was 90,984 s (approximately 25.27 h).

*2.4.1. Grid Model Error Analysis.* The grid model error is the distance from the grid midpoint to the inertial navigation track when the discrete grid is used to approach the inertial navigation track. As shown in Figure 4, when the track is determined, this error is related solely to the grid model. It is the inherent error that occurs when using a grid to generate the track and cannot be compensated for by the algorithm.

In Figure 4, the dark blue line represents the inertial navigation track and the light blue grid represents the grid through which the track passes. These grids can be used as a discrete representation of the track. The distance from the midpoint of the grid to the track is the grid model error. Obviously, a smaller grid model error corresponds to the simulation of the discrete grid sequence being closer to the track and the track restoration degree being higher.

For the two simulated tracks, the 11th, 12th, and 13th levels of the ISEA4H hexagonal grid system were used for the calculations. According to the number of hexagonal grids per unit area and the midpoint distance of the adjacent grids at the different levels, the equivalent quadrilateral grid was selected for comparison. The error of the hexagonal grid model was then evaluated via an analysis of the calculation

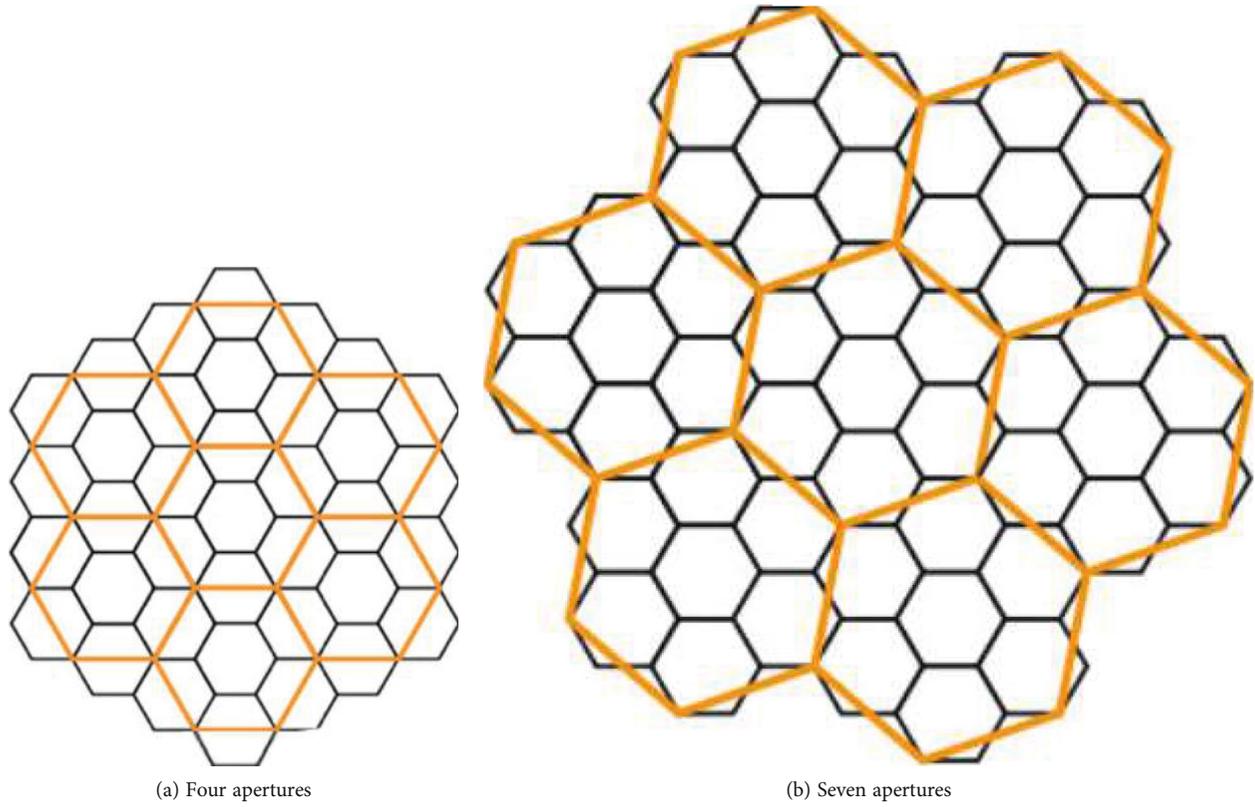


FIGURE 3: Schematic diagrams of four- and seven-aperture division.

TABLE 3: Error settings of the inertial components.

Inertial components	Error term	Error parameter
Gyro	Constant drift	$0.001^\circ/\text{h}$
	Angle random walk	$0.001^\circ/\sqrt{\text{h}}$
Accelerometer	Zero bias	$50 \mu\text{g}$
	Speed random walk	$5 \mu\text{g}/\sqrt{\text{Hz}}$

results. The correspondence between the grids is shown in Table 4.

Table 4 indicates that the 11th level of the hexagonal grid roughly corresponds to a geographical quadrilateral grid with  $2'$  resolution; the 12th level corresponds to a  $1'$ -resolution geographic quadrilateral grid; and the 13th level corresponds to a  $30''$ -resolution geographic quadrilateral grid. The inertial navigation solutions were found for the two tracks, and on this basis, the grid model errors of the hexagonal and quadrilateral grids at different levels and resolutions were calculated, compared, and analyzed. The calculation results are shown in Table 5.

Table 5 shows that, compared with those of the corresponding resolution quadrilateral grid, the grid model errors of the hexagonal grid system at different levels are reduced to a certain extent.

For track 1, compared with that of the  $2'$ -resolution quadrilateral grid, the average grid model error of the 11th

level of the hexagonal grid is reduced by 24.87% and the standard deviation is reduced by 20.12%. Compared with that of the  $1'$ -resolution quadrilateral grid, the average grid model error of the 12th level of the hexagonal grid is reduced by 20.00% and the standard deviation is reduced by 20.62%. Compared with that of the  $30''$ -resolution quadrilateral grid, the average grid model error for the 13th level of the hexagonal grid is reduced by 20.41% and the standard deviation is reduced by 20.54%.

For track 2, compared with that of the  $2'$ -resolution quadrilateral grid, the average grid model error of the 11th level of the hexagonal is reduced by 14.01% and the standard deviation is reduced by 14.25%. Compared with the  $1'$ -resolution quadrilateral grid, the average grid model error of the 23th level of the hexagonal grid is reduced by 14.54% and the standard deviation is reduced by 14.86%. Compared with the  $30''$ -resolution quadrilateral grid, the average grid model error of the 13th level of the hexagonal grid is reduced by 14.02% and the standard deviation is reduced by 14.29%.

For tracks 1 and 2, the adoption of a hexagonal grid reduces the average value and standard deviation of the grid model error by more than 14% and the grid model error accuracy of track 1 is improved by more than 20%. It can be seen therefore that a hexagonal grid has more advantages for track discretization than a quadrilateral geographic grid. In addition, for the same track, the accuracy improvement between each level of the hexagonal grid and its corresponding resolution quadrilateral grid is

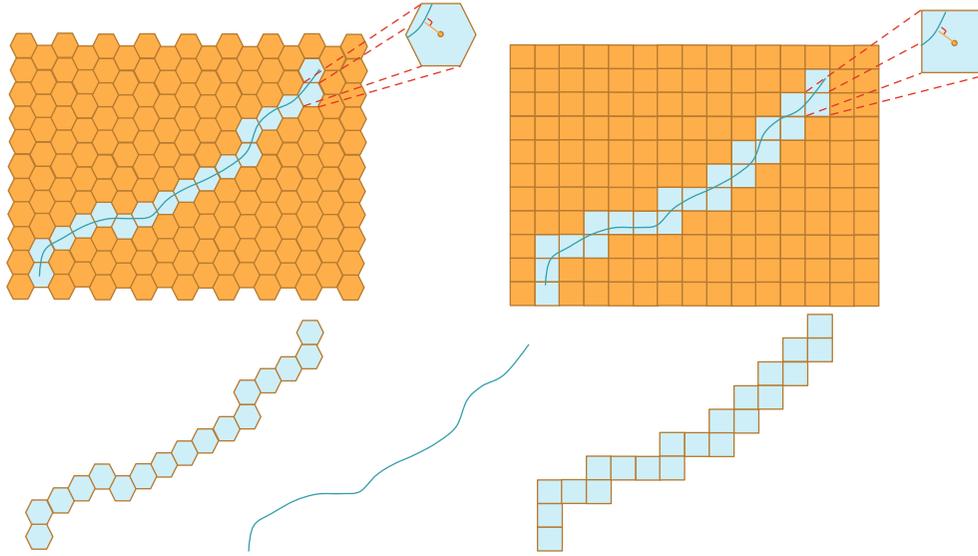


FIGURE 4: Schematic diagram of the grid model error.

TABLE 4: Correspondence between the ISEA4H hexagonal grids and the quadrilateral grids.

Grid	Resolution (level)	Distance between adjacent grid midpoint (km)	Average area of grid (km <sup>2</sup> )	Number of grids in the test area
Hexagon	11th level	3.767	12.161	92,767
	12th level	1.884	3.040	371,059
	13th level	0.942	0.760	1,484,237
Quadrilateral	2'	3.711	13.771	90,000
	1'	1.855	3.441	360,000
	30''	0.927	0.859	1,440,000

small, indicating that the grid model error for the same track is related to only the grid selection not the level or resolution.

**2.4.2. Grid Matching Positioning Error Analysis.** The grid midpoint of the matching point is usually selected as the matching track position when using the gravity gradient matching algorithm for matching navigation. The grid matching positioning error is the distance between the real track position point and the grid midpoint of the grid where the matching track located at the corresponding time (i.e., the distance between the two points  $\sqrt{\Delta x^2 + \Delta y^2}$ ), which can reflect the effect of the grid system applied to the matching navigation, as shown in Figure 5.

In Figure 5, the dark blue line represents the matching track, the pink line represents the real track, the black point represents the grid midpoint of the matching track, the red points represent the real track points, and the distance from the real track points to the grid midpoints of the matching track at the corresponding time is the grid matching positioning error. A smaller grid matching positioning error corresponds to a higher gravity gradient matching navigation accuracy.

The two simulated tracks were matched via gravity gradient filtering, and the navigation speed was set to 10 nautical miles per hour. The measurements were made every 3 min. Taking the matched tracks as the experimental objects, the matching positioning errors of the two tracks using the different grid systems were calculated. The results are shown in Table 6.

Table 6 shows that, compared with those of corresponding resolution quadrilateral grids, the grid matching positioning errors of the hexagonal grid system at different levels and are reduced to a certain extent.

For track 1, compared with that of the 2'-resolution quadrilateral grid, the average grid matching positioning error of the 11th level of the hexagonal grid is reduced by 15.26% and the standard deviation is reduced by 17.05%. Compared with that of the 1'-resolution quadrilateral grid, the average grid matching positioning error of the 12th level of the hexagonal grid is reduced by 17.25% and the standard deviation is reduced by 20.36%. Compared with that of the 30''-resolution quadrilateral grid, the average grid matching positioning error of the 13th level of the hexagonal grid is reduced by 19.73% and the standard deviation is reduced by 22.06%.

TABLE 5: Grid model error statistics, where Std indicates the standard deviation.

Track	Grid	Resolution	Track grid number	Maximum (km)	Minimum (km)	Average (km)	Std (km)
1	Hexagon	11th level	305	2.1375	0.0046	0.9955	0.5725
	Quadrilateral	2'	361	2.5376	0.0131	1.2442	0.7167
	Hexagon	12th level	606	1.0944	0.0015	0.4964	0.2845
	Quadrilateral	1'	721	1.2668	0.0030	0.6205	0.3584
	Hexagon	13th level	1202	0.5530	0.0010	0.2456	0.1420
	Quadrilateral	30''	1435	0.6314	0.0012	0.3086	0.1787
2	Hexagon	11th level	314	2.1532	0.0174	1.0555	0.6103
	Quadrilateral	2'	348	2.5376	0.0052	1.2274	0.7117
	Hexagon	12th level	625	1.1001	0.0015	0.5233	0.3042
	Quadrilateral	1'	695	1.2668	0.0036	0.6123	0.3573
	Hexagon	13th level	1255	0.5530	0.0011	0.2638	0.1524
	Quadrilateral	30''	1390	0.6314	0.0021	0.3068	0.1778

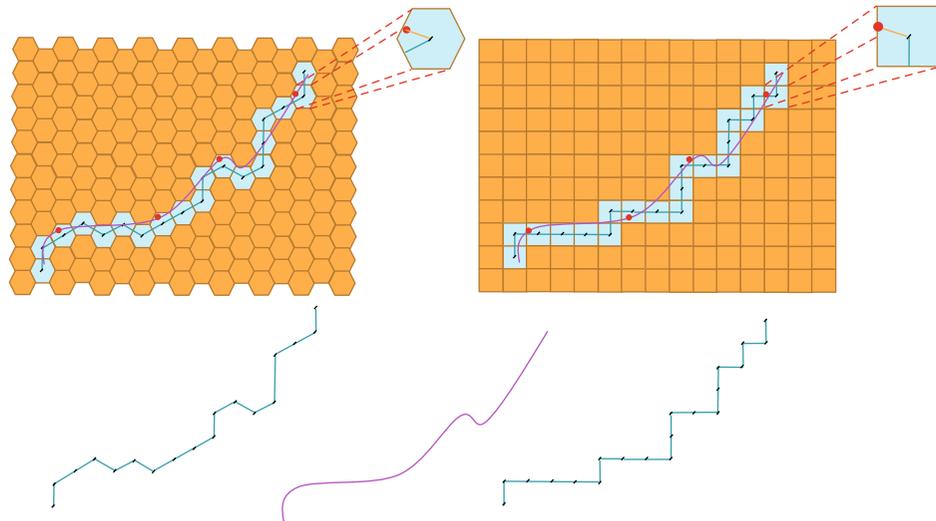


FIGURE 5: Schematic diagram of the grid matching positioning error.

TABLE 6: Statistics of the grid matching positioning error.

Track	Matching points	Grid	Resolution	Maximum (km)	Minimum (km)	Average (km)	Std (km)
1	519	Hexagon	11th level	4.5587	0.2973	2.0117	1.2539
		Quadrilateral	2'	5.4560	0.2569	2.3741	1.5116
		Hexagon	12th level	3.9888	0.1045	1.5601	0.8121
		Quadrilateral	1'	4.7149	0.1294	1.8855	1.0197
		Hexagon	13th level	2.4434	0.0345	0.9136	0.4129
		Quadrilateral	30''	3.6127	0.0311	1.1382	0.5297
2	505	Hexagon	11th level	4.7454	0.3128	1.9120	1.3632
		Quadrilateral	2'	5.6110	0.2812	2.2575	1.6521
		Hexagon	12th level	3.8101	0.1463	1.5581	0.9199
		Quadrilateral	1'	4.3525	0.2130	1.9024	1.1242
		Hexagon	13th level	2.5012	0.1038	0.9004	0.4008
		Quadrilateral	30''	3.5649	0.1249	1.1332	0.5041

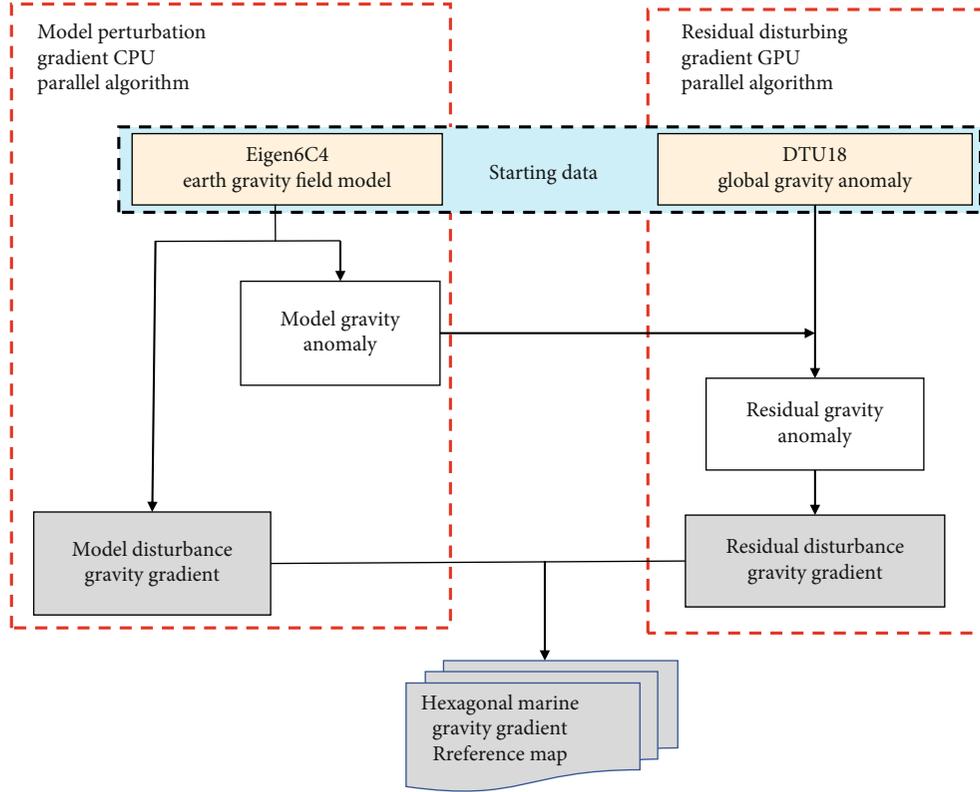


FIGURE 6: Reference map's construction flowchart.

For track 2, compared with that of the  $2'$ -resolution quadrilateral grid, the average grid matching positioning error of the 11th level of the hexagonal grid is reduced by 15.30% and the standard deviation is reduced by 17.48%. Compared with the  $1'$ -resolution quadrilateral grid, the average grid matching positioning error of the 12th level of the hexagonal grid is reduced by 18.09% and the standard deviation is reduced by 18.17%. Compared with the  $30''$ -resolution quadrilateral grid, the average grid matching positioning error of the 13th level of the hexagonal grid is reduced by 20.54% and the standard deviation is reduced by 20.50%.

For track 1, different level hexagonal grids are used for the simulation matching. The average grid matching positioning error obtained by matching with the corresponding resolution quadrilateral grid is reduced by more than 15.26% and the standard deviation is reduced by more than 17.05%. For track 2, the average grid matching positioning error is reduced by more than 15.30% and the standard deviation is reduced by more than 17.48%. It can be seen that the hexagonal grid system has more advantages than the quadrilateral geographic grid. With increasing grid resolution, the advantages of the hexagonal grid for matching and positioning gradually increase. Compared with that of quadrilateral geographic grids at the same resolution, the standard deviation of the grid matching positioning error of the hexagonal grids at all levels is reduced by more than 15%, which can effectively suppress the matching positioning error caused by the discretization of the grid model.

### 3. Fast Construction Method for a Hexagonal Marine Gravity Gradient Reference Map

Because satellite gravity gradient measurements only have access to low-frequency gradient signals, the current global marine gravity gradient grid numerical model used to assist inertial navigation is generally based on satellite altimetry gravity anomalies and calculated using remove-restore technology according to Stokes' theory [4, 10]. The gravity gradient data obtained via this method can be regarded as new observational information to some extent. However, the calculation process is complex and time-consuming and it is difficult to efficiently complete the construction requirements for a large-scale, or even global, hexagonal marine gravity gradient grid model. Therefore, it is also necessary to design an effective scheme to parallelize the construction process and improve its calculation efficiency. Figure 6 shows the calculation process of reference map's construction.

*3.1. Calculation Process for a Disturbing Gravity Gradient.* According to the Stokes remove-restore method, the disturbing gravity gradient can be expressed as [26]

$$T_{\alpha\beta} = T_{\alpha\beta}^M + \delta T_{\alpha\beta}, \alpha, \beta = x, y, z. \quad (1)$$

Here,  $T_{\alpha\beta}$  represents the disturbing gravity gradient,  $T_{\alpha\beta}^M$  represents the model disturbing gravity gradient calculated using the gravitational field model of the Earth, and  $\delta T_{\alpha\beta}$

represents the residual disturbing gravity gradient. Because of the symmetry of the gravity gradient, the sum of the main diagonal elements is 0 and six components (five independent components and one as the check condition) are usually selected for calculation when constructing the marine gravity gradient reference map. The calculation process and the method used for each component are similar. As an example, we introduce the calculation process for the  $T_{xx}$  disturbing gravity gradient.

*3.1.1. Calculation of the Model Disturbing Gravity Gradient.* The spherical harmonic series calculation of the ocean model disturbing gravity gradient can be expressed as [11]

where  $T_{xx}^M$  is the horizontal east-west component of disturbed gravity gradient,  $R$  is the average radius of the Earth,  $(\varphi, \lambda)$  represents the geocentric latitude and longitude of the ocean calculation point,  $f$  is the geocentric gravitational constant,  $M$  is the mass of the earth,  $n$  and  $m$  represent the truncation degrees and orders,  $\bar{C}_{n,m}^*$  and  $\bar{S}_{n,m}$  are the coefficients of the gravitational field model of the Earth, and  $\bar{P}_{nm}(\sin \varphi)$  represents the associated Legendre function. Because the latitude of the hexagonal grid in a single latitude circle is not completely consistent, the exchange integration order method cannot be used to improve the calculation efficiency for the model disturbing gradient calculation based on the hexagonal grid.

Because the ocean gravity gradient needs to be constructed in any region of the world, we need to consider that the recurrence of the quadratic differential of the Legendre function in equation (2) will be singular near the pole.

$$P_{nm}^{xx}(\sin \varphi) = \begin{cases} \frac{1}{2} \sqrt{\frac{n(n-1)(n+1)(n+2)}{2}} \bar{P}_{n,2}(\sin \varphi) - \frac{1}{2} (n+1)(n+2) \bar{P}_{n,0}(\sin \varphi) & m=0, \\ \frac{1}{4} \sqrt{(n-2)(n-1)(n+2)(n+3)} \bar{P}_{n,3}(\sin \varphi) - \frac{1}{4} (3n+1)(n+2) \bar{P}_{n,1}(\sin \varphi) & m=1, \\ \frac{1}{4} \sqrt{(n-3)(n-2)(n+3)(n+4)} \bar{P}_{n,4}(\sin \varphi) - \frac{1}{2} (n^2+3n-2) \bar{P}_{n,2}(\sin \varphi) + \\ \frac{1}{4} \sqrt{2n(n-1)(n+1)(n+2)} \bar{P}_{n,0}(\sin \varphi) & m=2, \\ \frac{1}{4} \sqrt{(n-m-1)(n-m)(n+m+1)(n+m+2)} \bar{P}_{n,m+2}(\sin \varphi) + \\ \frac{1}{4} \sqrt{(n+m-1)(n+m)(n-m+1)(n-m+2)} \bar{P}_{n,m-2}(\sin \varphi) - \\ \frac{1}{2} (n^2+3n-m^2+2) \bar{P}_{n,m}(\sin \varphi) & m>2. \end{cases} \quad (3)$$

$$\begin{aligned} T_{xx}^M(R, \varphi, \lambda) \\ = \frac{fM}{R^3} \sum_{n=2}^N \sum_{m=0}^n (\bar{C}_{nm}^* \cos m\lambda + \bar{S}_{nm} \sin m\lambda) P_{nm}^{xx}(\sin \varphi), \end{aligned} \quad (2)$$

Using equations (2) and (3), the model disturbing gravity gradient component of an arbitrary point on the globe can be calculated.

*3.1.2. Calculation of the Residual Disturbing Gravity Gradient.* Because the gravitational field model of the Earth is a spherical harmonic series truncated to a finite order, it can only represent the gravitational field information of

the Earth to a finite order, which will inevitably produce a truncation error. The purpose of the remove-restore method is to remove the long-wave component in the disturbing gravity gradient and calculate the remaining part using the residual gravity anomaly  $\delta\Delta g$  of the surrounding points.  $\delta\Delta g = \Delta g - \Delta g_M$ , where  $\Delta g$  represents the actual gravity anomaly and  $\Delta g_M$  represents the model gravity anomaly calculated using the gravitational field model of the Earth.

In this paper, the DTU18 model data retrieved from satellite altimetry data were selected as the actual gravity anomaly values. Because of the singularity in the calculation of the central area, the calculation process needs to be divided into a central area and an inner area.

(1) *Calculation of the Residual Disturbing Gravity Gradient in the Inner Area.* The influence of the residual gravity anomaly on the disturbing gravity gradient is [11]

$$\delta T_{xx} = \frac{R}{4\pi} \iint_{\sigma} \left[ \frac{1}{r^2 \cos^2 \varphi} \left( \frac{\partial^2 S(r, \psi)}{\partial \psi^2} \left( \frac{\partial \psi}{\partial \lambda} \right)^2 + \frac{\partial S(r, \psi)}{\partial \psi} \frac{\partial^2 \psi}{\partial \lambda^2} \right) - \frac{\tan \varphi}{r^2} \frac{\partial S(r, \psi)}{\partial \psi} \frac{\partial \psi}{\partial \varphi} + \frac{1}{r} \frac{\partial S(r, \psi)}{\partial r} \right] \delta \Delta g d\sigma, \quad (4)$$

where

$$\begin{aligned} \frac{\partial S(r, \psi)}{\partial r} &= -\frac{t^2}{R} \left[ \frac{1-t^2}{D^3} + \frac{4}{D} + 1 - 6D - t \cos \psi \right. \\ &\quad \left. \cdot \left( 13 + 6 \ln \frac{1-t \cos \psi + D}{2} \right) \right], \\ \frac{\partial S(r, \psi)}{\partial \psi} &= -t^2 \sin \psi \left[ \frac{2}{D^3} + \frac{6}{D} - 8 - 3 \frac{1-t \cos \psi - D}{D \sin \psi} \right. \\ &\quad \left. - 3 \ln \frac{1-t \cos \psi + D}{2} \right], \\ \frac{\partial^2 S(r, \psi)}{\partial r^2} &= \frac{t^2}{R^2} \left[ (1-t \cos \psi) \left( \frac{3(1-t^2)}{D^5} - \frac{4}{D^3} \right) - \frac{1+t^2}{D^3} \right. \\ &\quad \left. - \frac{10}{D} - 18D + 2 - 3t \cos \psi \right. \\ &\quad \left. \cdot \left( 15 + 6 \ln \frac{1-t \cos \psi + D}{2} \right) \right], \\ \frac{\partial^2 S(r, \psi)}{\partial \psi^2} &= -t^2 \cos \psi \left[ \frac{2}{D^3} + \frac{6}{D} - 8 - 3 \frac{1-t \cos \psi - D}{D \sin^2 \psi} \right. \\ &\quad \left. - 3 \ln \frac{1-t \cos \psi + D}{2} \right] \\ &\quad + t^3 \sin^2 \psi \left[ \frac{6}{D^5} + \frac{6}{D^3} + 3 \frac{D-1}{D^2 \sin^2 \psi} \right. \\ &\quad \left. + 3 \frac{D+1}{D(1-t \cos \psi + D)} - 3 \frac{1-t \cos \psi - D}{D \sin^2 \psi} \right. \\ &\quad \left. \cdot \left( \frac{2 \cos \psi}{t \sin^2 \psi} + \frac{1}{D^2} \right) \right], \\ \frac{\partial^2 S(r, \psi)}{\partial r \partial \psi} &= \frac{t^3 \sin \psi}{R} \left[ \frac{3(1-t^2)}{D^5} + \frac{4}{D^3} + \frac{6}{D} - 13 \right. \\ &\quad \left. - 6 \ln \frac{1-t \cos \psi + D}{2} + \frac{6 \cos \psi (D+1)}{D(1-t \cos \psi + D)} \right] \\ t &= \frac{R}{r}, D = (1 - 2t \cos \psi + t^2)^{1/2} \\ \psi &= \cos^{-1} \left( \sin \varphi \sin \varphi' + \cos \varphi \cos \varphi' \cos (\lambda' - \lambda) \right). \end{aligned} \quad (5)$$

Using equations (4) and (5), the gravity gradient component of the residual disturbing  $\delta T_{xx}$  at any point in the world can be calculated.

(2) *Singular Integral Processing in the Central Region.* It can be seen from equations (2) and (4) that the Stokes kernel function and its derivative are used in the calculation process; the Stokes kernel function is equal to the square of the distance between the flow point and the calculation point divided by the radial diameter. Therefore, when the flow point approaches the calculation point, its derivative will abnormally increase, resulting in a singularity, especially when calculating the radial disturbing gravity gradient [4, 12]. Therefore, the original formula cannot be used to calculate the disturbing gravity gradient in the central area. We use the Taylor series expansion method to solve the singularity problem in the central region.

First, the regional geoid can be approximately regarded as a plane. Then, the coordinate of the calculation point on the geoid is  $(x, y, 0)$  and the central area is regarded as a circular area with the calculation point at the center and a distance of  $1-2'$  as the radius. Then, the radius is  $s_0 = R\psi_0$ , where  $\psi_0$  is the angular distance between the center of the central region and the edge. After gridding the central area, the radius  $s_0$  can be approximated to  $\sqrt{\Delta x \times \Delta y}$ .

If the rectangular coordinate  $(x, y)$  is transformed into the polar coordinate system with the calculation point as the origin, the coordinate of the flow point in the polar coordinate system is  $(s, \alpha)$ , where  $s = \sqrt{(x' - x)^2 + (y' - y)^2}$ ,  $x$  and  $y$  are the coordinates of the calculation point, and  $x'$  and  $y'$  are the coordinates of the flow point. Then, it is easy to obtain  $x = s \cos \alpha$ ,  $y = s \sin \alpha$ , and  $dx' dy' = s ds d\alpha$ . Therefore, the residual disturbing gravity gradient in the central area can be obtained as follows:

$$\delta T_{xx} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{s_0} \frac{\delta \Delta g}{s^2} (3 \cos^2 \alpha - 1) ds d\alpha. \quad (6)$$

The Taylor series expansion  $\delta \Delta g$  at the calculation point in equation (6) can then be obtained:

$$\delta \Delta g = \delta \Delta g_p + x g_1 + y g_2 + \frac{1}{2} (x^2 g_{11} + 2xy g_{12} + y^2 g_{22}) + \dots, \quad (7)$$

where  $g_1 = \partial \Delta g / \partial x$ ,  $g_2 = \partial \Delta g / \partial y$ ,  $g_{11} = \partial^2 \Delta g / \partial x^2$ ,  $g_2 = \partial^2 \Delta g / \partial y^2$ , and  $g_{12} = \partial^2 \Delta g / \partial x \partial y$ .

Using the above conversion relationship between rectangular and polar coordinates, equation (7) can be converted into polar coordinates in the following form:

$$\begin{aligned} \delta \Delta g &= \delta \Delta g_p + s (g_1 \cos \alpha + g_2 \sin \alpha) \\ &\quad + \frac{s^2}{2} (g_{11} \cos^2 \alpha + 2g_{12} \sin \alpha \cos \alpha + g_{22} \sin^2 \alpha) + \dots \end{aligned} \quad (8)$$

In equation (8), only the linear term (assuming that the higher-order terms are small) is considered and the residual gravity anomaly grid data can be used to obtain the coefficient parameters  $g_1$ ,  $g_2$ ,  $g_{11}$ ,  $g_{12}$ , and  $g_{22}$ .

Using the above conversion relationship between rectangular and polar coordinates, equation (7) can be converted into polar coordinates in the following form.

$$\begin{aligned}
g_{1_{n,m}} &= \frac{\delta\Delta g_{n+1,m} - \delta\Delta g_{n-1,m}}{2\Delta x}, \\
g_{2_{n,m}} &= \frac{\delta\Delta g_{n,m+1} - \delta\Delta g_{n,m}}{2\Delta y}, \\
g_{11_{n,m}} &= \frac{g_{1_{n+1,m}} - g_{1_{n-1,m}}}{2\Delta x}, \\
g_{12_{n,m}} &= \frac{g_{1_{n,m+1}} - g_{1_{n,m-1}}}{2\Delta y}, \\
g_{22_{n,m}} &= \frac{g_{2_{n,m+1}} - g_{2_{n,m-1}}}{2\Delta y}.
\end{aligned} \tag{9}$$

Here,  $n$  represents the number of grid rows,  $n = 1, \dots, N$ , and  $m$  represents the number of grid columns,  $m = 1, \dots, M$ . Substituting equations (8) and (9) into equation (6) gives

$$\delta T_{xx} = \frac{5g_{11} - g_{22}}{16} s_0. \tag{10}$$

The gravity gradient component of the residual disturbing  $\delta T_{xx}$  in the central region at any point in the world can then be calculated using equation (10).

**3.2. The CPU+GPU Hybrid Parallel Scheme.** It can be seen from Section 3.1 that, because the latitude of the hexagonal grid in a single latitude circle is not completely consistent, when using the hexagon as the grid unit to construct the marine gravity gradient reference map, it is impossible to use methods such as exchanging the integral order to reduce the time-consuming calculation process of the model disturbing gradient. Moreover, the calculation of the residual disturbing gradient increases rapidly with increasing resolution, calculation range, and integration radius. Therefore, it is necessary to accelerate the construction of the hexagonal marine gravity gradient reference map using the CPU+GPU hybrid parallel algorithm to parallelize the process. According to the calculation process described in Section 3 and the calculation characteristics of the model disturbing and residual disturbing gradients, the calculation of the model disturbing gradient is allocated to the CPU and that of the residual disturbing gradient is assigned to the GPU.

**3.2.1. Model Perturbation Gradient CPU Parallel Algorithm.** CPUs have a strong overall regulation ability, and the memory in the CPU is easier to expand to create a large memory space for calculations. Currently, the number of CPU threads on a personal computer (PC) can reach 16 or more and the threads of a multi-CPU server can exceed 100. Even though the number of threads on a CPU is often less than that on a GPU, the single thread computing power of a CPU is much stronger than that of a GPU. The single-thread processing frequency of a CPU can generally reach more than 2 GHz. Therefore, CPUs are suitable for overall

parallel tasks with large memory consumption and relatively concentrated computing. OpenMP is a parallel algorithm designed and developed for multicore/shared memory processors [13, 14]. Using OpenMP does not necessitate large modifications to the original serial program and only requires making simple improvements to avoid memory read/write conflicts during multithreaded computing. Therefore, it is suitable for parallelizing the model disturbing gradient calculation, which has a large amount of recursive computations. After determining the parallel computational mode, it is necessary to create parallel units. The parallel computing unit is the smallest unit used to allocate tasks to each thread during parallel computing, and the memory space allocated to each computing unit should be independent of the other units. Because of the differing latitudes of the hexagonal grids in the same latitude circle, the method of exchanging the integral order cannot be used to reduce the calculation amount; accordingly, the calculations can only be done one by one using a separate grid as the calculation unit. Therefore, it was decided to take a grid point as the parallel computing unit.

The division of the parallel computing units in the above manner can only be implemented when calculating the small-scale and low-order model disturbing gradient. However, when calculating large-scale and high-order models, such a division will lead to excessive memory consumption, such that most PCs will have difficulty completing the calculation. To reduce the dependence of the parallel computation on the memory, this paper proposes the ‘‘array dimensionality reduction + thread grouping’’ method, which can reduce the memory occupied by the parallel computation and weaken the dependence of the parallel computation on memory.

**Array dimensionality reduction:** when calculating the Legendre function  $\bar{P}_{nm}(\cos \theta)$ , it is often necessary to create a two-dimensional array  $P[n][M]$  to store  $\bar{P}_{nm}(\cos \theta)$ . However, the Legendre function  $\bar{P}_{nm}(\cos \theta)$  is not a complete two-dimensional array but rather an array in the form of a triangular matrix. Directly using a two-dimensional array to store the function will waste half of the memory; therefore, a dimensionality reduction is performed, as shown in Figure 7. We can save half the memory by removing all the zeros in the two-dimensional array and then arranging the Legendre function stored in the lower triangle into a one-dimensional array from left to right for each row.

**Thread grouping:** in general, high-performance computers have more CPU threads and more memory, while ordinary PCs have fewer threads and less memory. Generally speaking, the number of CPU threads on a computer is often positively correlated with the size of its memory. Therefore, to make the proposed method applicable to as many types of computers as possible, this paper proposes grouping the parallel computing units according to the number of threads used in the parallel computation, as shown in Figure 8. The computing units in each group are serially computed and share memory. Taking a range of  $10^\circ \times 10^\circ$  and the 12th level of the hexagonal grid in Table 3 as an example, the number of parallel computing units is equal to the number of grids: 371,059. If the computing order is 100, each parallel unit needs to create at least a double

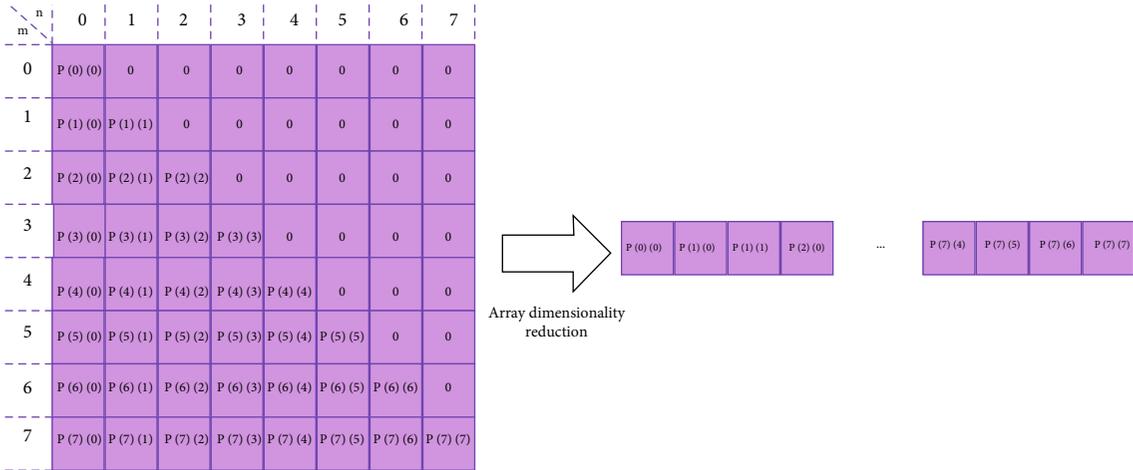


FIGURE 7: Schematic diagram of the array dimensionality reduction.

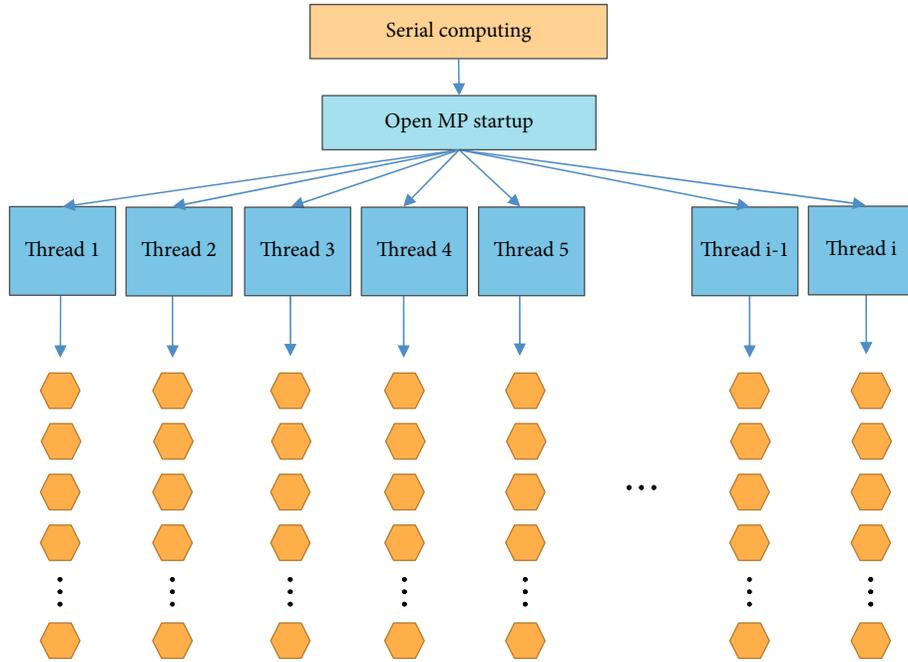


FIGURE 8: Schematic diagram of the grouping calculation.

array with a size of  $(1 + 100) \times 100/2$  to store the Legendre functions. Such an array requires 39.5 KB of memory. If it is directly parallelized, at least  $371,059 \times 39.5$  KB of memory is required, that is, 13.98 GB. With increasing order, the memory consumption will increase exponentially, which is not practical for ordinary PCs. According to the above “thread grouping” scheme, when the number of available threads  $I$  is 100, only 3.86 MB of memory is required, which is achievable for most computers.

Using the above “array dimensionality reduction + thread grouping” method, the parallel calculation of the model disturbing gradient can be completed on a multicore processor computer.

**3.2.2. Residual Disturbing Gradient GPU Parallel Algorithm.** CUDA is a heterogeneous parallel platform developed by NVIDIA for its GPU products [15]. With the further development of hardware technology, the number of GPU cores is increasing [24, 25]. Taking rtx3080 as an example, the number of CUDA cores has reached 5,120, with a price of less than 10,000 yuan, which greatly reduces the threshold for parallel computing. Ordinary computers equipped with this type of graphics card can use the CUDA platform to complete parallel computing tasks. Even though GPUs have a large number of threads, their disadvantages are more prominent than those of CPUs. First, the single-thread GPU computing ability is weak and GPUs have high requirements for the amount of computation of a

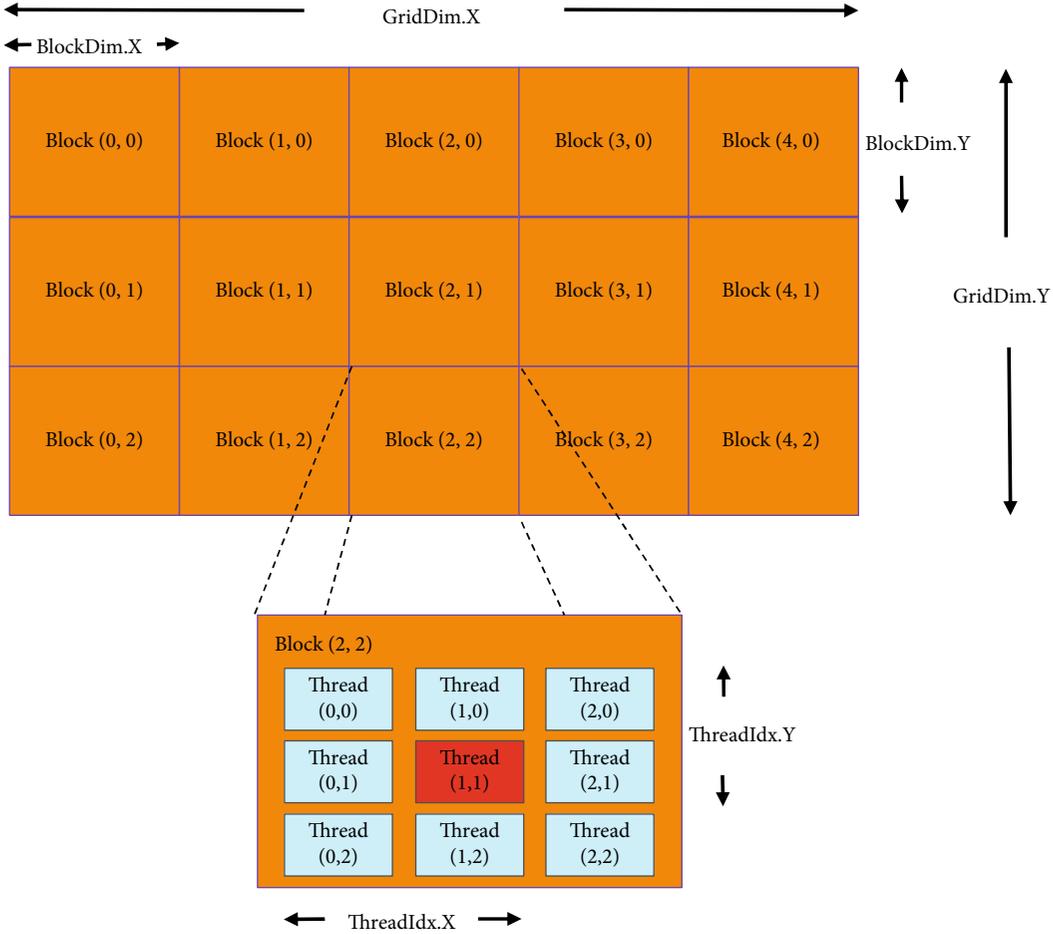


FIGURE 9: CUDA thread index diagram.

parallel computing unit. If the amount of computation of the parallel computing unit is too large, the GPU will crash. Second, the GPU video memory is smaller than the computer memory and cannot be expanded; therefore, it is suitable for operations with less initial data but more computing units. The calculation process of the residual disturbing gradient conforms to the above characteristics; therefore, a GPU was selected to accelerate its calculation process in parallel.

The CUDA programming language organizes the GPU threads in the form of a thread grid, as shown in Figure 9. There are 15 (3 × 5) thread blocks in a single thread grid, each of which contains 9 (3 × 3) threads. To accurately index these threads, four built-in variables are introduced: threadIdx, representing the index in the thread block; blockIdx, representing the index in the thread grid of the thread block; and blockDim and gridDim, representing the dimensions (i.e., the length and width) of the thread grid and the thread block, respectively [16–18]. Using these variables, the x and y indexes of a thread in the process grid can be calculated. The calculation formulas are shown in

$$\begin{aligned}
 \text{Idx} &= (\text{blockIdx.x} * \text{blockDim.x}) + \text{threadIdx.x}, \\
 \text{Idy} &= (\text{blockIdx.y} * \text{blockDim.y}) + \text{threadIdx.y}.
 \end{aligned}
 \tag{11}$$

When calculating various gravity gradient maps, the number of grid points is large. Taking a range of 10° × 10° and the 12th level of the hexagonal grid as an example, there are 371,059 grid points and the edges of the hexagonal grid are irregular. Directly using the two-dimensional grid for indexing not only increases the time consumption of the thread scheduling process [16, 19] but also is not conducive to thread indexing at the edge of the grid. Therefore, it is necessary to reduce the dimension of the thread grid; however, after dimension reduction, the original adjacent grids will be separated, resulting in a reduction of the overall spatial clustering. Therefore, spatial clustering of the one-dimensional grid needs to be ensured as much as possible during the dimension reduction process. Accordingly, this paper uses the one-dimensional thread indexing method to divide the tasks of the GPU threads, taking the calculation of the residual disturbing gravity gradient of any grid point as a calculation unit and sorting all of the calculation units. To ensure the spatial clustering of the grid data in the one-dimensional indexing process, the regional hexagonal grid is sorted accordingly, as shown in Figure 10, and the grid in the left panel is sorted into rows (or columns). Each row (or column) is connected end to end into the form shown in the right panel, the grid of which is recoded. Taking the red hexagonal grid in the 3rd row of the 15th column in

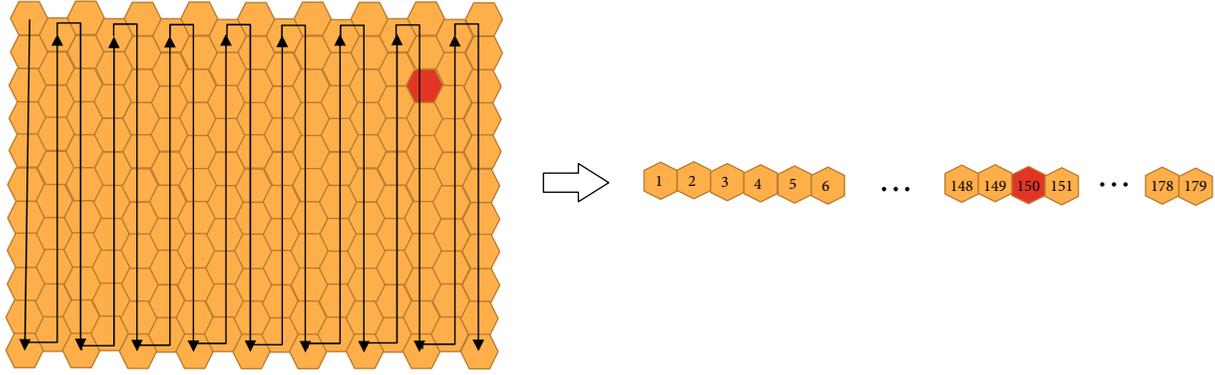


FIGURE 10: Schematic diagram of the one-dimensional coding of a hexagonal grid.

the figure as an example, its grid code after dimension reduction sorting is 150.

Using the parallel computing scheme designed in Sections 4.1 and 4.2, the rapid construction of a hexagonal marine gravity gradient reference map can be completed using CPU+GPU hybrid programming.

#### 4. Experimental Verification and Analysis

To verify the effectiveness of the proposed method, the algorithm was experimentally tested on the following hardware and software experimental platforms. The hardware environment was a high-performance desktop computer with an Intel®Xeon(R) Gold 6130 @ 2.10 GHz processor having 22 cores and 44 threads. The GPU was a Tesla V100 16G with 64.00 GB of memory. The software environment was a 64-bit Windows 10 operating system, with a VS2013+ CUDA9 2 platform.

The selection range of the test area (the range of the marine grid) was from 19°N to 29°N and from 41°W to 51°W. The deepest sea depth in this area is 6,872 m, the shallowest sea depth is 0 m, and the regional average elevation is  $-3,657.22$  m. The maximum value of the gravity anomaly in this area is 89.8 mGal, the minimum value is  $-83.3$  mGal, and the average gravity anomaly is 4.4 mGal. The specific relationship between the regional gravity anomaly and the submarine topography is shown in Figure 11.

The GPU is sensitive to the selection of the data types in the calculation, and the calculation efficiency of different data types varies greatly [16, 18]. The efficiency of using single-precision floating-point data for calculations is four times that of using double-precision data. Therefore, when using GPUs for parallel calculations, single-precision data should be used as much as possible to ensure the accuracy and improve the calculation efficiency. However, in the process of calculating the residual disturbing gravity gradient tensor, it is necessary to use sine and cosine functions and the square operation. The influence of the truncation error caused by using only single-precision data on the calculation results cannot be ignored. Therefore, this section uses both single-precision floating-point data and double-precision type data and uses the GPU to calculate the total tensor of the residual disturbing gravity gradient in the experimental area. We take the serial calculation results of the CPU using

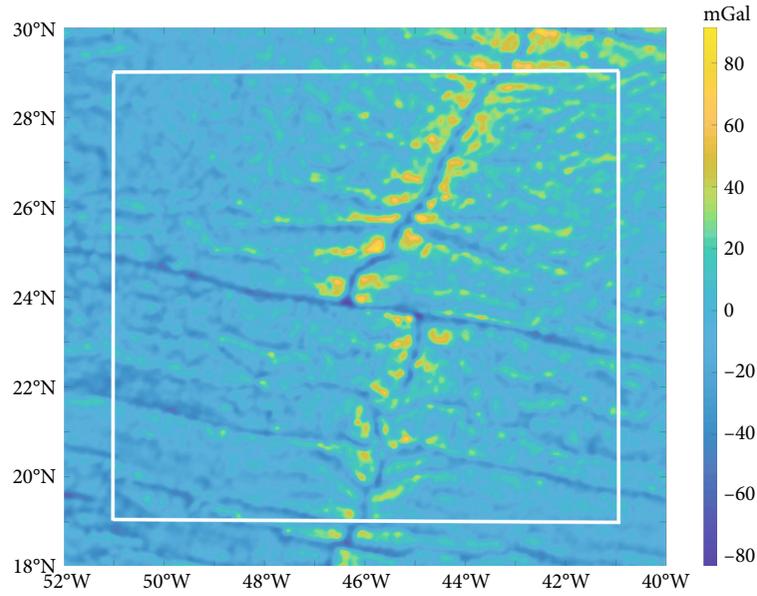
the double-precision data as the “true value” and perform a comparative analysis with the GPU calculation results, so as to select the data type of the parallel calculation and verify the reliability of the proposed parallel calculation method. By comparing the GPU calculation results with the CPU calculation results, the accuracy of the parallel calculation results of the different data types on the GPU can be obtained, as shown in Table 7.

Table 7 shows that the significant-digit limitation causes the truncation error generated by the calculation results using single-precision data to have a significant impact on the results. The maximum error is  $2.72E$ , the minimum error is  $-4.66E$ , the average error is  $0.18E$ , and the root mean square (RMS) error is  $0.37E$ . The absolute value of the error calculated with double-precision data is less than or equal to  $10^{-6}E$ , and the average difference and RMS value are less than  $10^{-12}E$ . The experimental results show that, even though the single-precision data can effectively improve the calculation efficiency, this precision has difficulty meeting the calculation requirements of the disturbing gravity gradient tensor. Therefore, we chose to use double-precision data to construct the gravity gradient reference map.

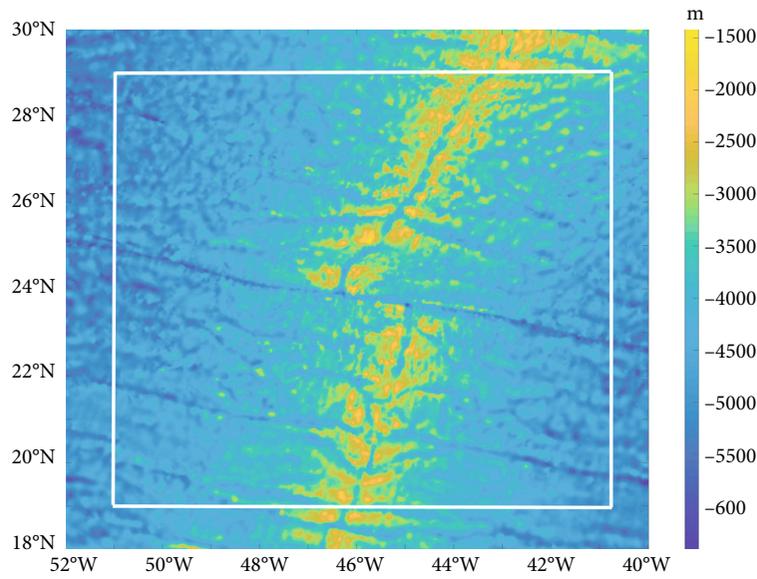
To further verify the reliability of the mathematical model of the parallel scheme and the correctness of the programming software, according to the constraint condition  $T_{xx} + T_{yy} + T_{zz} = 0$ , the above parallel calculation scheme was used to calculate the disturbing gravity gradient tensor and evaluate the internal consistency of the calculation results. The experimental results are shown in Table 8.

It can be seen from Table 8 that the maximum value of  $T_{xx} + T_{yy} + T_{zz}$  is  $0.105E$ , the minimum value is  $-0.127E$ , the standard deviation is  $0.0075E$ , and the RMS error is  $0.0091E$ . The calculation results fluctuate randomly around 0 and meet the internal consistency conditions of the disturbing gravity gradient tensor. This, to a certain extent, indicates the reliability of the mathematical model of the proposed parallel scheme and the correctness of programming software.

**4.1. Efficiency Analysis of the Parallel Algorithm.** In the previous section, the data type selection, calculation reliability, and calculation accuracy of the parallel computing were



(a) DTU18 gravity anomaly



(b) Submarine topography

FIGURE 11: Physical characteristic map of the calculation area.

TABLE 7: Accuracy analysis of performing parallel computing using different data types, where RMS indicates the root mean square.

Data type	Maximum difference ( $E$ )	Minimum difference ( $E$ )	Average difference ( $E$ )	RMS ( $E$ )
Double precision data	$10^{-6}$	$-10^{-6}$	$<10^{-12}$	$<10^{-12}$
Single precision data	2.72	-4.66	0.18	0.37

TABLE 8: Numerical characteristics of  $T_{xx} + T_{yy} + T_{zz}$ , where  $T$  is the disturbing gravity gradient tensor.

	Maximum ( $E$ )	Minimum ( $E$ )	Average ( $E$ )	Std ( $E$ )	RMS ( $E$ )
$T_{xx} + T_{yy} + T_{zz}$	0.105	-0.127	$<10^{-4}$	0.0075	0.0091

TABLE 9: Time consumption of the different parallel schemes.

Parallel scheme	Model $T^M$ calculation time	Model $T^M$ calculation acceleration ratio	Residual $\delta T$ calculation time	Residual $\delta T$ calculation acceleration ratio	Total time	Total acceleration ratio
Serial computing	120.35 h	1	7.92 h	1	128.27 h	1
Pure CPU parallelism	2.98 h	40.38	12.11 min	39.24	3.18 h	40.33
Pure GPU parallelism	6.73 h	17.86	25.32 s	1126.38	6.74 h	19.03
GPU+CPU hybrid parallel	2.98 h	40.38	25.32 s	1126.38	2.99 h	42.90

analyzed. This section analyzes the efficiency of the proposed parallel computing scheme on the basis of the results found in the previous section.

*4.1.1. Efficiency Analysis of Different Parallel Schemes.* To fully verify the effectiveness of the proposed parallel scheme, four calculation schemes are examined in this section: (1) a serial computing scheme, (2) a pure CPU parallel computing scheme, (3) a pure GPU parallel computing scheme, and (4) the proposed CPU+GPU hybrid parallel scheme. These four calculation schemes were used to calculate the disturbing gravity gradient tensor in the test area shown in Table 9. The 12th level of the ISEA4H hexagonal grid was used for the calculation, and the EIGEN-6C4 gravity field model was selected to calculate the disturbing gravity gradient tensor of the model, with the truncation order calculated to 2,160 [20–23]. The actual gravity anomaly of the DTU18 data retrieved via satellite altimetry data was selected, and the actual gravity anomaly of the hexagonal grid was interpolated. The central area incorporated the grid area  $1'$  from the midpoint of the central grid, and the integral radius was in the range of  $1^\circ$ . The calculation results are shown in Figure 12, and the calculation time statistics are shown in Table 9.

Table 9 indicates that the total tensor of the disturbing gravity gradient in the experimental area is calculated by the serial calculation in 120.35 h or more than 5 days. The pure CPU parallel computing scheme takes 3.18 h, for an acceleration ratio of 40.33 times. The pure GPU parallel computing scheme takes 6.74 h and has an acceleration ratio of 19.03 times. The CPU+GPU hybrid parallel computing scheme takes 2.99 h and has an acceleration ratio of 42.90 times. The acceleration effect of the hybrid parallel computing scheme is better than that of the CPU or GPU single system parallel computing schemes. When only the CPU is used for the parallel calculation, the acceleration ratio of calculating the disturbing gravity gradient of the model is 40.38 times and the acceleration ratio of calculating the residual disturbing gravity gradient is 39.24 times. Using only the GPU for the parallel calculation results in the acceleration ratio of calculating the disturbing gravity gradient of the model being 17.86 times and the acceleration ratio of calculating the residual disturbing gravity gradient being 1126.38 times. It can be seen that the proposed CPU+GPU hybrid parallel computing scheme can fully exploit the available computing power and effectively improve the

computing efficiency according to the CPU and GPU characteristics.

*4.1.2. Calculation Efficiency Analysis for Different Resolutions.* To further verify the universal effectiveness of the proposed CPU+GPU parallel algorithm, here we use ISEA4H hexagonal grid data at different levels to conduct numerical experiments, calculate the marine area within the  $10^\circ \times 10^\circ$  test area, and calculate the model disturbing gravity gradient to order 2,160, with an integral radius of  $1^\circ$ . The hexagonal grid levels used are the 11th, 12th, and 13th levels. The calculation time and acceleration are shown in Table 10.

It can be seen from Table 10 that, with increasing hexagonal grid level, the calculation time increases; therefore, a large-scale high-resolution hexagonal gravity gradient reference map is increasingly difficult to obtain in a short time. The proposed CPU+GPU parallel algorithm was used to calculate the marine gravity gradient reference map within the test area. The acceleration ratio of the hexagonal gravity gradient reference map at the different levels can reach more than 40 times and up to 50.92 times. It takes 0.70 h to construct the hexagonal marine gravity gradient reference map at the 11th level with a range of  $10^\circ \times 10^\circ$ , while it takes 28.45 h for a traditional serial calculation. In parallel, it takes 2.98 h to construct of the 12th-level hexagonal marine gravity gradient reference map, while the serial method takes 120.35 h or more than 5 days. In parallel, it takes 11.84 h to construct the 13th-level hexagonal marine gravity gradient reference map, while the serial method takes 602.95 h or more than 25 days. This demonstrates that the proposed parallel scheme is generally effective for the construction of hexagonal disturbing gravity gradient reference maps at different levels and can effectively improve the construction speed of gravity gradient reference maps at various resolutions.

Table 10 shows that, for increasing hexagonal grid level (resolution), the calculation ratio of the proposed hybrid parallel scheme increases. The main reason for this is that the calculation time of the disturbing gravity gradient part of the different level models is different from that of the residual disturbing gravity gradient part. Taking level 12 as an example, even though the calculation ratio using the GPU to calculate the residual disturbing gravity gradient can reach more than 1,000 times, the overall acceleration ratio is less than 43 times. The main reason is that, when using a hexagonal grid to calculate the disturbing

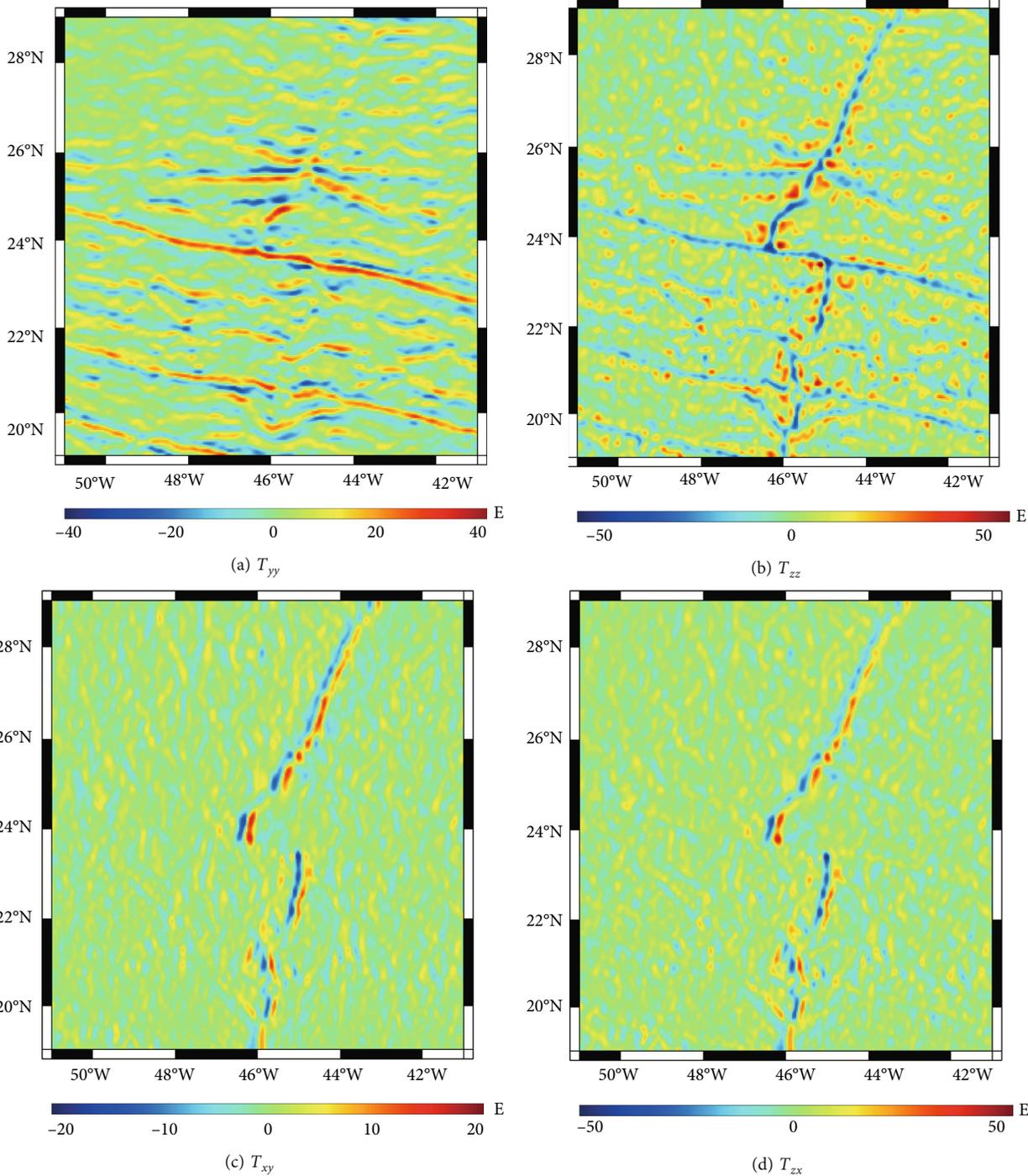


FIGURE 12: Continued.

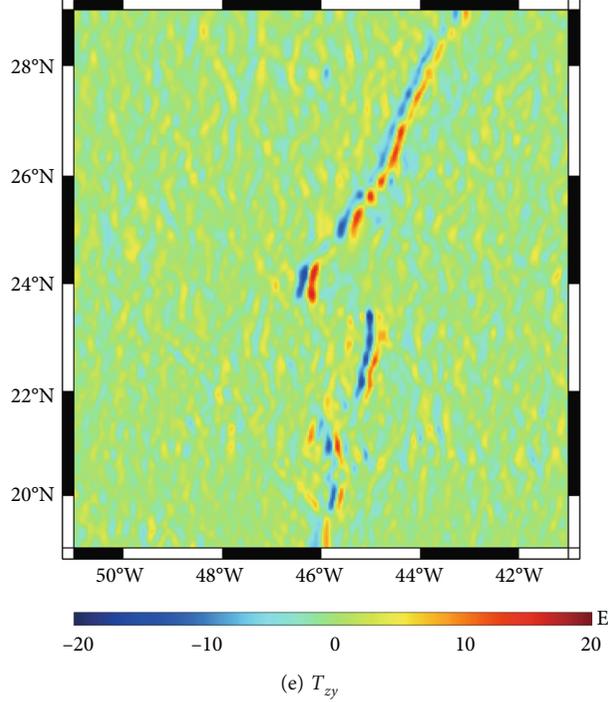


FIGURE 12: Gravity gradient reference map of the calculation area.

TABLE 10: Calculation time for different resolutions.

Level	Serial computing time (h)	Parallel computing time consuming (h)	Acceleration ratio
11th level	28.45	0.70	40.64
12th level	120.35	2.98	42.90
13th level	602.95	11.84	50.92

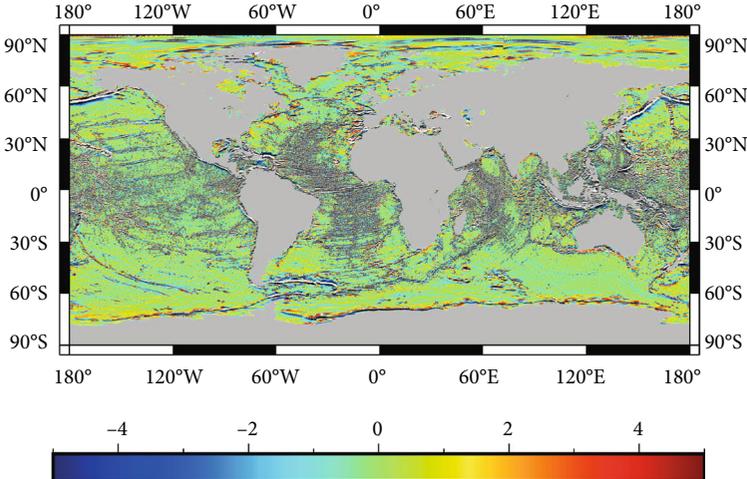
gravity gradient, because the latitude of the hexagonal grid in the same latitude circle is not completely consistent, it is impossible to use methods such as exchanging the integration order to accelerate the calculation process of the model disturbing gradient. In the process of calculating the disturbing gravity gradient using the 12th-level hexagonal grid, the calculation time of the gravity gradient part of the model accounts for 93.8% of the total calculation time, while that of the residual gravity gradient part accounts for 6.2% of the total. Therefore, the acceleration effect of the calculation of the model part is dominant. With increasing level (resolution), the calculation amount of the disturbing gradient part of the model increases linearly, while that of the residual disturbing gradient part increases exponentially; accordingly, the proportion of the calculation acceleration effect of the residual part gradually increases. Therefore, the acceleration effect of the CPU+GPU hybrid parallel scheme gradually increases with increasing level (resolution), making it more suitable than other schemes to building a high-resolution marine gravity gradient reference map.

TABLE 11: Numerical characteristics of the global marine gravity gradient.

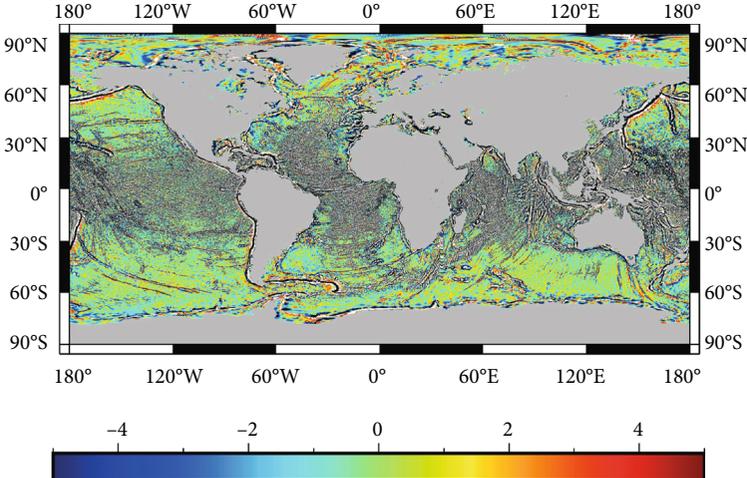
	Maximum ( $E$ )	Minimum ( $E$ )	Average ( $E$ )	Median ( $E$ )
$T_{xx}$	131.43	-118.97	$<10^{-4}$	0.0725
$T_{yy}$	102.45	-106.05	$<10^{-4}$	0.0584
$T_{zz}$	221.83	-117.77	$<10^{-4}$	0.0100
$T_{xy}$	78.66	-74.72	$<10^{-4}$	0.0000
$T_{zx}$	112.31	-111.50	$<10^{-4}$	0.0037
$T_{zy}$	103.02	-107.41	$<10^{-4}$	-0.0064

## 5. Construction of the Global Marine Hexagonal Gravity Gradient Reference Map

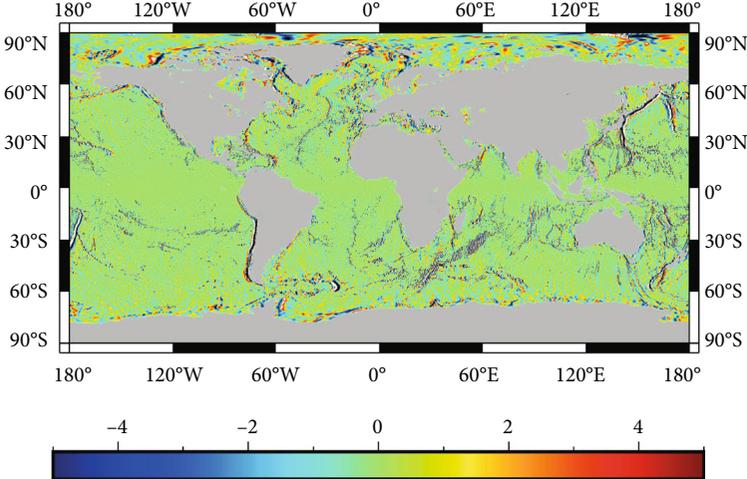
Based on the proposed CPU+GPU hybrid parallel scheme, we used the ISEA4H hexagonal grid system to rapidly construct a global, full-tensor, 12th-level hexagonal marine gravity gradient reference map. The EIGEN-6C4 gravity field model was used for the calculation, and the truncation was calculated to order 2160. Using the satellite altimetry DTU18 model gravity anomaly, the hexagonal grid data were formed first and then the residual disturbing gravity gradient was calculated. An integral radius of  $1^\circ$  was selected for the calculation in the inner area, and the central area selected the central grid point and the six surrounding grid points, totaling seven grid areas. The calculation results are shown in Table 11 and Figure 13, and the internal consistency of the gravity gradient tensor is shown in Table 12.



(a)  $T_{yy}$



(b)  $T_{zz}$



(c)  $T_{xy}$

FIGURE 13: Continued.

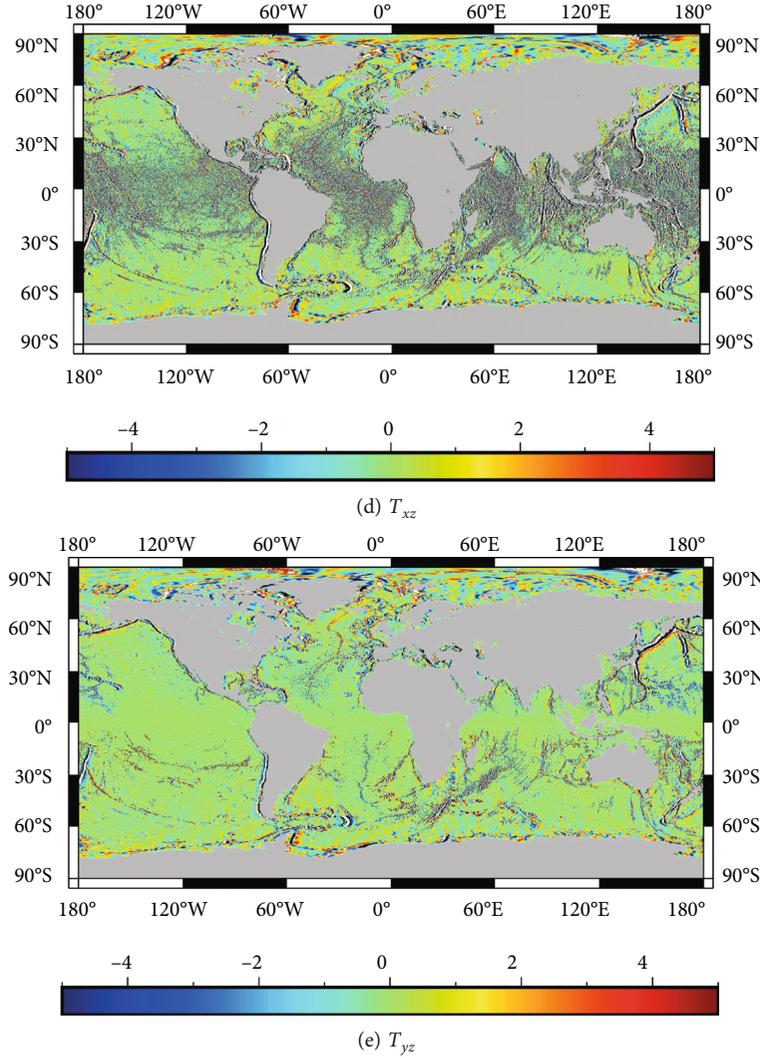


FIGURE 13: Global hexagonal ocean gravity gradient reference map model.

TABLE 12: Numerical characteristics of the global  $T_{xx} + T_{yy} + T_{zz}$ .

	Maximum ( $E$ )	Minimum ( $E$ )	Average ( $E$ )	Std ( $E$ )	RMS ( $E$ )
$T_{xx} + T_{yy} + T_{zz}$	0.175	-0.168	$<10^{-4}$	0.00012	0.00037

The results in Tables 11 and 12 show that the maximum absolute value of the global full-tensor hexagonal marine gravity gradient reference map model is less than  $0.18 E$  and that the standard deviation is less than  $0.0002 E$ , which meets the internal conditions of the disturbing gravity gradient tensor. The global marine gravity gradient value is small. The absolute value of the gravity gradient of more than 80% of the grid points is less than  $10 E$ , the average value is less than  $10^{-6} E$ , the median is near  $0 E$ , and the gravity gradient is related to the seabed topography. The variation in the gravity gradient value is obvious in areas with severe topographic fluctuations.

## 6. Discussion and Conclusions

This paper first introduced the hexagonal grid system as a grid unit for constructing a marine gravity gradient reference map, compared the advantages and disadvantages of the ISEA4H and H3 grid systems, and selected the ISEA4H hexagonal grid system according to the characteristics of gravity gradient matching navigation. Then, via calculations, statistics, and analyses of the grid model error and grid matching positioning error, the feasibility and effectiveness of the hexagonal grid for gravity gradient matching navigation were verified. Consequently, based on Stokes' remove-

restore theory and according to the characteristics of the hexagonal grid disturbing gravity gradient calculation, we proposed an efficient CPU+GPU hybrid parallel scheme, which can effectively improve the construction efficiency of a hexagonal marine gravity gradient reference map. The experimental calculations and analyses show the following.

- (1) Compared with the H3 hexagonal grid system, the ISEA4H hexagonal grid system has a better characteristic of equal area; that is, the number of grids between adjacent levels changes less. In addition, the grid levels of the ISEA4H system better correlate with quadrilateral geographic grids, which is conducive to the construction of reference maps based on existing data. Therefore, the ISEA4H grid system was selected to construct the hexagonal marine gravity gradient reference map
- (2) On the premise of only considering errors caused by the geometric structure of the reference grid in the process of discretizing a track, the grid model error and matching positioning error of a hexagonal grid and quadrilateral geographic grid with the same resolution were analyzed for the process of matching navigation. The experimental results show that the standard deviation of the grid model error of a hexagonal grid is more than 14% less than that of a quadrilateral grid with the same resolution. The standard deviation of the grid matching positioning error is reduced by more than 15% for a hexagonal grid, which effectively proves the feasibility and effectiveness of hexagonal grids in gravity gradient matching navigation
- (3) Based on the hexagonal grid, we constructed a reference map of the marine gravity gradient using the Stokes remove-restore technique. To address the low efficiency of calculating the disturbing gravity gradient on a hexagonal grid, a CPU+GPU hybrid parallel computing scheme was proposed. This scheme can make full use of the computing power of the available equipment and effectively improve the computing efficiency. Compared with a serial algorithm, the calculation acceleration ratio of the parallel computing scheme can reach more than 40 times, which is beneficial for the construction of high-resolution gravity gradient reference maps. A global full-tensor hexagonal marine gravity gradient reference map was constructed under the following accuracy conditions: an absolute error of the parallel computing of less than or equal to  $10^{-6} E$ , a maximum absolute value of less than  $0.18 E$ , and a standard deviation of less than  $0.0002 E$

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare no conflict of interest.

## Authors' Contributions

Huang Yan and Li Shanshan were responsible for conceptualization; Li Xinxing and Huang Yan were responsible for methodology and software; Fan Diao and Tan Xuli were responsible for validation; Wang Aoming was responsible for data curation; Huang Yan was responsible for writing—original draft preparation; Lv Minghao was responsible for writing—review and editing; Li Shanshan was responsible for funding acquisition. All authors have read and agreed to the published version of the manuscript.

## Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant numbers 42174007 and 42174013.

## References

- [1] W. Zhao, *Research on Inertial Navigation with Real Time Gravity Gradient Compensation*, Wuhan University, Wuhan, 2014, Doctoral thesis.
- [2] W. Zhao, Z. T. Wang, and Q. Ma, "Simulation of gravity gradient data and its applications on inertial navigation," *Journal of Geodesy and Geodynamics*, vol. 34, no. 5, p. 134, 2014.
- [3] Y. M. D. Ou, Z. M. Sun, and Z. H. Zhai, "The calculation of vertical gravity gradient anomalies and its application in bathymetry inversion," *Journal of Geodesy and Geodynamics*, vol. 36, no. 9, pp. 766–769, 2016.
- [4] B. J. Zhao, G. L. Yang, and Z. J. Li, "Research on shallow structure using the surface wave and primary wave of shallow reflection seismic data," *Journal of Geodesy and Geodynamics*, vol. 39, no. 4, pp. 361–365, 2019.
- [5] I. Brodsky, "H3: Uber's hexagonal hierarchical spatial index," 2018, <https://eng.uber.com/h3/>.
- [6] L. Middleton and J. Sivaswamy, *Hexagonal Image Processing*, Springer, London, 2005.
- [7] R. Sadourny, A. Arakawa, and Y. Mintz, "Integration of the non-divergent barotropic vorticity equation with an icosahedral-hexagonal grid for the SPHERE1," *Monthly Weather Review*, vol. 96, no. 6, pp. 351–356, 1968.
- [8] J. Thurnburn, "A PV-based shallow-water model on a hexagonal-icosahedral grid," *Monthly Weather Review*, vol. 125, no. 9, pp. 2328–2347, 1997.
- [9] X. Tong, J. Ben, Y. Wang, Y. Zhang, and T. Pei, "Efficient encoding and spatial operation scheme for aperture 4 hexagonal discrete global grid system," *International Journal of Geographical Information Science*, vol. 27, no. 5, pp. 898–921, 2013.
- [10] J. X. Huang, *Research on Fast (Parallel) Computing of Disturbing Gravity Master's Thesis*, PLA Information Engineering University, Zhengzhou, 2017.
- [11] H. Moritz, *Advanced physical geodesy*, Herbert Wichmann Verlag, 1980.
- [12] T. Chen and G. B. Zhang, "Deriving the full gravitational gradient tensor from gravity anomaly: an equivalent source

- technique,” *Progress in Geophysics*, vol. 34, no. 4, pp. 1398–1410, 2019.
- [13] Z. H. Yu, *High Performance Parallel Programming with MPI*, Tsinghua University Press, Beijing, China, Beijing, 2001.
- [14] X. Zhou, P. Yu, and A. H. Wen, “Parallel forward modelling algorithm with gravity and gravity gradient data based on MPI and OpenMP,” *Global Geology*, vol. 37, no. 3, pp. 897–904, 2018.
- [15] S. Zhang and Y. L. Zhu, *CUDA for GPU High Performance Computing*, China Water & Power Press, Beijing, China, Beijing, 2009.
- [16] Nvidia, *Nvidia Tesla P100—The Most Advanced Datacenter Accelerator Ever Built Featuring Pascal GP100, the World’s Fastest GPU*, Nvidia Whitepaper, 2016, v01.1(Nvidia WP-08019-001).
- [17] Z. X. Chen, X. H. Meng, L. H. Guo, and G. F. Liu, “Tree-dimensional fast forward modeling and the inversion strategy for large scale gravity and gravimetry data based on GPU,” *Chinese Journal of Geophysics*, vol. 55, no. 12, pp. 4069–4077, 2012.
- [18] W. Z. Liu, *Parallel Computing and Performance Optimization*, Machinery Industry Press, Beijing, China, 2015.
- [19] S. Girish, A. Abhishek, and B. Baidurya, “A fast parallel Gauss Jordan algorithm for matrix inversion using CUDA,” *Computers & Structures*, vol. 128, pp. 31–37, 2013.
- [20] H. T. Fan, C. X. Guo, X. H. Wang et al., “Adaptability analysis on ultra-high-gravity model EIGEN-6C2 in Chinese coastal zone,” *Science of Surveying and Mapping*, vol. 40, no. 9, pp. 18–22, 2015.
- [21] W. Zhong, J. Ouyang, D. Yang, X. Wang, Z. Guo, and K. Hu, “Effect of the in situ leaching solution of ion-absorbed rare earth on the mechanical behavior of basement rock,” *Journal of Rock Mechanics and Geotechnical Engineering*, vol. 14, no. 4, pp. 1210–1220, 2022.
- [22] D. Pan, K. Hong, H. Fu, J. Zhou, N. Zhang, and G. Lu, “Influence characteristics and mechanism of fragmental size of broken coal mass on the injection regularity of silica sol grouting,” *Construction and Building Materials*, vol. 269, article 121251, 2021.
- [23] D. Pan, K. Hong, H. Fu, J. Zhou, and N. Zhang, “Experimental study of the mechanism of grouting colloidal nano-silica in over-broken coal mass,” *Quarterly Journal of Engineering Geology and Hydrogeology*, vol. 54, no. 4, 2021.
- [24] H. Zhang, Y. R. Qian, and H. Y. Hou, “Parallel optimization of edge detection algorithm for geological images under CUDA,” *Computer Engineering and Design*, vol. 40, no. 3, pp. 691–698, 2019.
- [25] W. Y. Li, J. Zhang, and W. Lin, “Regularized inversion of 3D gravity data: a new GPU parallelized method based on CUDA,” *Geophysical & Geochemical Exploration*, vol. 40, no. 1, pp. 179–184, 2016.
- [26] P. A. Odera, Y. Fukuda, and Y. Kuroishi, “A high-resolution gravimetric geoid model for Japan from EGM2008 and local gravity data,” *Earth, Planets and Space*, vol. 64, no. 5, pp. 361–368, 2012.