ORIGINAL ARTICLE

Fast calculation of gravitational effects using tesseroids with a polynomial density of arbitrary degree in depth

Fang Ouyang[1] · Long-wei Chen[2,3] · Zhi-gang Shao[1]

Abstract

Fast and accurate calculation of gravitational effects on a regional or global scale with complex density environment is a critical issue in gravitational forward modelling. Most existing significant developments with tessroid-based modelling are limited to homogeneous density models or polynomial ones of a limited order. Moreover, the total gravitational effects of tesseroids are often calculated by pure summation in these methods, which makes the calculation extremely time-consuming. A new efficient and accurate method based on tesseroids with a polynomial density up to an arbitrary order in depth is developed for 3D large-scale gravitational forward modelling. The method divides the source region into a number of tesseroids, and the density in each tesseroid is assumed to be a polynomial function of arbitrary degree. To guarantee the computational accuracy and efficiency, two key points are involved: (1) the volume Newton's integral is decomposed into a one-dimensional integral with a polynomial density in the radial direction, for which a simple analytical recursive formula is derived for efficient calculation, and a surface integral over the horizontal directions evaluated by the Gauss–Legendre quadrature (GLQ) combined with a 2D adaptive discretization strategy; (2) a fast and flexible discrete convolution algorithm based on 1D fast Fourier transform (FFT) and a general Toepritz form of weight coefficient matrices is adopted in the longitudinal dimension to speed up the computation of the cumulative contributions from all tesseroids. Numerical examples show that the gravitational fields predicted by the new method have a good agreement with the corresponding analytical solutions for spherical shell models with both polynomial and non-polynomial density variations in depth. Compared with the 3D GLQ methods, the new algorithm is computationally more accurate and efficient. The calculation time is significantly reduced by 3 orders of magnitude as compared with the traditional 3D GLQ methods. Application of the new algorithm in the global crustal CRUST1.0 model further verifies its reliability and practicability in real cases. The proposed method will provide a powerful numerical tool for large-scale gravity modelling and also an efficient forward engine for inversion and continuation problems.

Keywords Gravitational field · Polynomial density · Gauss–Legendre quadrature · Fast Fourier transform

✉ Long-wei Chen
  longweichen_glut@glut.edu.cn

  Fang Ouyang
  FangOuyang92@163.com

  Zhi-gang Shao
  shaozg@seis.ac.cn

[1] Institute of Earthquake Forecasting, China Earthquake Administration, No. 63, Fuxing Road, Haidian district, Beijing 100036, China

[2] Guangxi Key Laboratory of Exploration For Hidden Metallic Ore Deposits, College of Earth Sciences, Guilin University of Technology, No. 319, Yanshan Street, Yanshan District, Guilin 541006, Guangxi, China

[3] School of Earth and Environment, University of Leeds, Woodhouse Lane, Leeds, West Yorkshire LS2 9JT, UK

1 Introduction

With the advent of satellite gravimetry, high-quality gravity measurements with global-scale coverage become available (Soler et al. 2019). To estimate the density distribution from observed data sets by a regularized inversion, an efficient and accurate forward engine is of great importance. The gravitational forward modelling primarily focuses on solving the volumetric integral of Newton's kernel and its derivatives. A common method for evaluating these integrals is to firstly decompose the source region into small elementary mass bodies of various geometrical shapes, such as rectangular prism and polyhedral body in Cartesian coordinates (e.g. Anderson 1976; Pohanka 1988; Holstein and Ketteridge 1996; Tsoulis 2012; D'Urso 2014a; Conway 2015; Wu and

Chen 2016; Benedek et al. 2018; Zhang and Chen 2018; Chen and Liu 2019; Wu et al. 2019), and tesseroid (spherical prism) in spherical coordinates (e.g. Heck and Seitz 2007; Grombein et al. 2013; Uieda et al. 2016; Zhao et al. 2019, 2021). Then, the overall gravitational effect is calculated by adding all contributions of each individual element together. For regional and global applications, the effect of the Earth's curvature is significant. Hence, the forward modelling are often implemented in geocentric spherical coordinates and use tesseroids, bounded by pairs of latitude, longitude, and radial surfaces, as building blocks.

Generally, the forward calculation in spherical coordinates consists of two parts: the evaluation of gravitational effects produced by a single tesseroid and the computation of cumulative contribution from all elementary cells. In contrast to the most common rectangular prism (Nagy et al. 2000; Wild-Pfeiffer 2008), the closed-form solution for a general tesseroid is not available. Due to the fact that it is an "elliptical integral", the volumetric integral for a tesseroid must be solved numerically or semi-analytically. The literature offers three main approaches: Taylor series expansion, Gauss–Legendre quadrature (GLQ), and mixed integration. The Taylor series expansion approach solves the elliptic integral using a Taylor series expansion of the triple integral and a subsequent analytical integration of the expansions (Heck and Seitz 2007; Grombein et al. 2013; Marotta et al. 2019). This approach has high accuracy at low latitudes, but exhibits decreasing accuracy towards the polar as the tesseroid gradually degenerates into an approximately triangular shape (Uieda et al. 2016). For the GLQ approach, the gravitational effects of a tesseroid are numerically calculated by a weighted sum of the contribution of Gaussian points (Li et al. 2011; Uieda et al. 2016; Lin and Denker 2019; Soler et al. 2019; Zhao et al. 2019, 2021; Qiu and Chen 2020). The accuracy of GLQ is predominantly controlled by the number of Gaussian nodes (Ku 1977). In general, the more the Gaussian nodes, the better the accuracy of GLQ, while the lower the computational efficiency. To reduce the computational cost and retain satisfactory accuracy at the same time, the idea of adaptive discretization is proposed. The adaptive discretization strategies, whether 2D (Lin and Denker 2019; Soler et al. 2019) or 3D (Li et al. 2011; Uieda et al. 2016), adopt a similar criterion that a tesseroid is divided if the distance to the computation point is smaller than a constant times the size of the tesseroid. This means that only the tesseroid close to the computation point requires a greater number of subdivisions, which as a result significantly shortens the computation time.

As an alternative, the mixed integration approach solves the volumetric integrals for a tesseroid using a 2D spherical integration in the horizontal directions and a 1D analytical integration in the radial dimension. To solve the 2D spherical integral, various numerical methods can be utilized, such as Taylor series expansion (e.g. Smith 2002), GLQ (e.g. Wild-Pfeiffer 2008; Lin and Denker 2019; Zhong et al. 2019), and double exponential quadrature rule (e.g. Fukushima 2018). Compared to the former volume integral approaches, this mixed method using a combination of the numerical calculation of a surface integral and the analytical integration over the radial coordinate can generate more accurate solutions with less computation cost. Another significant advantage lies in that the mixed method can be naturally extended to the case of a general tesseroid with a radially varying density, especially a polynomial one. In fact, most existing volume integral approaches limit their applications only to homogeneous mass bodies. However, the constant density assumption is not realistic for most geological structures. In real media, the subsurface density must increase with increasing depth, due to the compression of the material composing the crust (Kumagai 1933; Maxant 1980; Rao et al. 1993; Kennett 1998). Therefore, considering the density variation with depth is of great practical significance (Chai and Hinze 1988; Holstein 2003; Hamayun et al. 2009; D'Urso 2014b; Wu and Chen 2016; D'Urso and Trotta 2017; Ren et al. 2017, 2018). Recently, some studies based on tesseroids generalize the constant density assumption to the linear (Lin and Denker 2019) and nonlinear cases (Fukushima 2018; Soler et al. 2019; Lin et al. 2020) so as to meet the requirements of depth variable density. However, most of these methods only consider a polynomial density up to an limited order.

The aforementioned methods provide desired solutions for the triple integral of an individual tesseroid. Actually, these solutions are weight coefficients, because the density term is usually pulled outside of the triple integral. Hence, the gravitational potential and its derivatives for a tesseroid can always be expressed as the products of the densities and the corresponding weight coefficients. Once the gravitational effects for a tesseroid are obtained, the overall effects of the whole source region can then be computed as a cumulative contribution of all individual tesseroids in terms of the principle of superposition. Although this process looks very simple, the computational cost substantially increases with the complexity of the model. Therefore, a precise evaluation is extremely time-consuming in practical applications, especially for the regional and global large-scale forward modelling.

To speed up the forward calculation, some attempts to apply the well-known fast Fourier transform (FFT) to gravity problems have been made. Smith (2002) adopted series expansions and 1D FFT to compute the gravity fields induced by distant global masses. He manipulated the integrals for the gravitational potential and acceleration into convolution integrals in longitude, and used 1D FFT techniques to efficiently compute the convolutions. Recently, Zhao et al. (2021) combined the 1D FFT into a 3D GLQ-based forward modelling method. Similar to Smith (2002), they accelerated the modelling by using the FFT techniques in discrete convolution
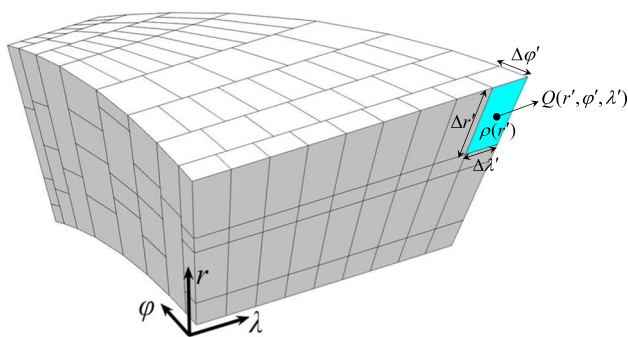
Fig. 1 Source region with irregular tesseroids in spherical coordinates (r, φ, λ). $\Delta r'$, $\Delta \varphi'$, and $\Delta \lambda'$ denote the mesh interval of a tesseroid with a polynomial density $\rho(r')$. $Q(r', \varphi', \lambda')$ is the geometric centre of the tesseroid. In the proposed method, $\Delta \lambda'$ is required to be constant along the longitudinal direction

along longitude. In the method of Zhao et al. (2021), however, the FFT-based fast algorithm is achieved using a square and symmetric Toeplitz form of the weight coefficient matrices. This requires that the number and location of the observation points must be consistent with those of the tesseroids in the longitudinal direction, which as a result may lead to undesirable computation cost and makes the method not that flexible in real cases. Moreover, these methods are established under the assumption of constant density in each tesseroid and hence face difficulties in practical applications with complex density environment.

In this work, we propose a new efficient and accurate method for 3D large-scale forward modelling that can be applied to the case of a general tesseroid with a polynomial density up to an arbitrary degree and with variable intervals in latitude and depth (see Fig. 1). Firstly, a combination of 1D analytical integration in the radial direction and 2D GLQ with an adaptive discretization in horizontal directions is introduced to calculate the gravitational effects of an individual tesseroid. The radial analytical integration here can be applicable to a polynomial density up to an arbitrary order by using a simple recursive relation. Next, a flexible and efficient discrete convolution algorithm based on 1D FFT and a general Toeplitz form of weight coefficient matrices is presented to compute the overall effects from all combined tesseroids. Then, the influence of some key parameters are discussed, and several shell models are established to verify the computational efficiency and accuracy of the proposed method. Finally, we apply the method to the global crustal model CRUST1.0 to further verify its flexibility and practicability.

2 Gravitational effects of a tesseroid with a polynomial density

In this section, a mixed method is presented to evaluate the gravitational potential, the gravitational vector and the
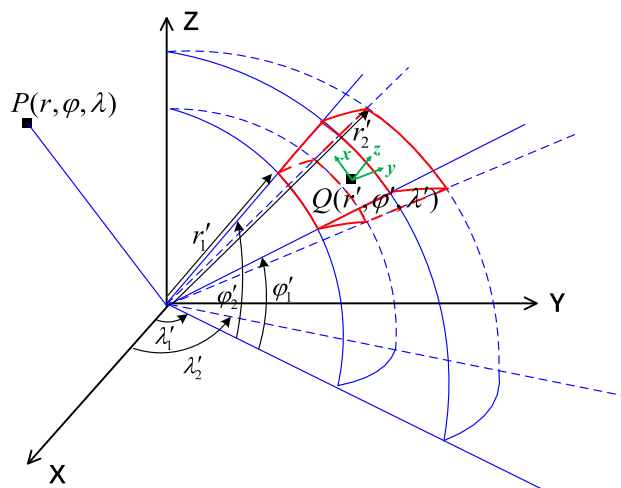
Fig. 2 A tesseroid in spherical coordinates (r, φ, λ). The spherical coordinates are referred to the geocentric Earth-fixed equatorial reference system defined by (X, Y, Z). $P(r, \varphi, \lambda)$ and $Q(r', \varphi', \lambda')$ denote the observation point and the centre of a tesseroid in the source region, respectively. The topocentric local Cartesian coordinate system (north/east/up) is represented by (x, y, z)

Marussi tensor for a tesseroid with a polynomial density function up to an arbitrary degree.

2.1 Gravitational potential

A tesseroid in geocentric spherical coordinates is defined by three pairs of boundaries: a pair of concentric spheres ($r'_1 =$ constant, $r'_2 =$ constant), a pair of parallels ($\varphi'_1 =$ constant, $\varphi'_2 =$ constant), and a pair of meridians ($\lambda'_1 =$ constant, $\lambda'_2 =$ constant (see Fig. 2). According to the Newton's integral (Heiskanen and Moritz 1967; Grombein et al. 2013), the gravitational potential of a tesseroid with a radially varying density $\rho(r')$ can be expressed as

$$V(r, \varphi, \lambda) = G \int_{\lambda'_1}^{\lambda'_2} \int_{\varphi'_1}^{\varphi'_2} \int_{r'_1}^{r'_2} \frac{r'^2}{\ell} \rho(r') \cos\varphi' dr' d\varphi' d\lambda' \quad (1)$$

where G is the gravitational constant, ρ the depth-dependent density, and ℓ the distance between the computation point $P(r, \varphi, \lambda)$ and the mass point $Q(r', \varphi', \lambda')$ with the following form

$$\ell = \sqrt{r^2 + r'^2 - 2rr' \cos\psi} \quad (2)$$

$$\cos\psi = \sin\varphi \sin\varphi' + \cos\varphi \cos\varphi' \cos(\lambda' - \lambda) \quad (3)$$

We assume that the density variation in depth for such a tesseroid follows an Nth-order polynomial function,

$$\rho(r') = \sum_{k=0}^{N} l_k(r') \rho_k \quad (4)$$

with

$$l_k(r') = \sum_{p=0}^{N} L_p^k r'^p$$

$$= \frac{(r' - r_0')(r' - r_1') \cdots (r' - r_{k-1}')(r' - r_{k+1}') \cdots (r' - r_N')}{(r_k' - r_0')(r_k' - r_1') \cdots (r_k' - r_{k-1}')(r_k' - r_{k+1}') \cdots (r_k' - r_N')}$$

$$(5)$$

where L_p^k is the polynomial coefficient, and r_k' and ρ_k are the kth node and nodal density along the radial direction in the tesseroid, respectively.

Using (1), (4), and (5), the gravitational potential over a tesseroid becomes

$$V(r, \varphi, \lambda) = \sum_{k=0}^{N} \sum_{n=0}^{N} L_n^k I_{n+2} \rho_k = \sum_{k=0}^{N} w_k \rho_k \tag{6}$$

where

$$w_k = \sum_{n=0}^{N} L_n^k I_{n+2} \tag{7}$$

$$I_{n+2} = G \int_{\lambda_1'}^{\lambda_2'} \int_{\varphi_1'}^{\varphi_2'} \cos\varphi' \left(\xi_{n+2} \big|_{r_1'}^{r_2'} \right) d\varphi' d\lambda' \tag{8}$$

$$\xi_{n+2} = \int \frac{r'^{n+2}}{\ell} dr' \tag{9}$$

with

$$\xi_0 = \begin{cases} \text{sign}(r' - r) \ln|r' - r|, & \cos\psi = 1 \\ \ln(r' + r), & \cos\psi = -1 \\ \ln|\ell + r' - r\cos\psi|, & \cos\psi \neq \pm 1 \end{cases}$$

$$\xi_m = \frac{1}{m} \left(r'^{m-1} \ell + (2m - 1)r\cos\psi \xi_{m-1} \right.$$

$$\left. - (m - 1)r^2 \xi_{m-2} \right)$$

$$m = 1, 2, ..., n + 2 \tag{10}$$

where $n = 0, 1, ..., N$, and w_k denotes the weight coefficient for the gravitational potential. ξ_{n+2} is the radial integration over r', which is solved analytically in a recursive way. By doing this, the original triple integral in (1) degenerates into several surface integrals given by I_{n+2}, and these double integrals can then be numerically evaluated using the 2D GLQ integration. Note that in (6), only $I_{n+2} = I_{n+2}(r, \varphi, \lambda)$ is related to the location of the computation point, while L_n^k and ρ_k are constants independent of r, φ, λ.

2.2 Gravitational vector fields

According to Wild-Pfeiffer (2008), the gravitational vector can be obtained using the first-order partial derivatives of the gravitational potential V by

$$g_x = \frac{1}{r} \partial_\varphi V, \quad g_y = \frac{1}{r\cos\varphi} \partial_\lambda V, \quad g_z = \partial_r V \tag{11}$$

where ∂_j $(j = \varphi, \lambda, r)$ denotes the first-order partial differentiation with respect to a spatial variable. Using (6), we have

$$\partial_j V = \sum_{k=0}^{N} \sum_{n=0}^{N} L_n^k I_{n+2,j} \rho_k = \sum_{k=0}^{N} w_k^j \rho_k \tag{12}$$

with

$$w_k^j = \sum_{n=0}^{N} L_n^k I_{n+2,j} \tag{13}$$

$$I_{n+2,j} = G \int_{\lambda_1'}^{\lambda_2'} \int_{\varphi_1'}^{\varphi_2'} \cos\varphi' \left. (\partial_j \xi_{n+2}) \right|_{r_1'}^{r_2'} d\varphi' d\lambda' \tag{14}$$

The recursive analytical expressions for $\partial_j \xi_{n+2}$ are given in Appendix A.

2.3 Gravitational gradient tensor

The components of the symmetric Marussi tensor can be expressed as follows (Tscherning 1976):

$$M_{xx} = \frac{1}{r} \left(\frac{\partial_{\varphi\varphi}^2 V}{r} + g_z \right)$$

$$M_{yy} = \frac{1}{r} \left(\frac{\partial_{\lambda\lambda}^2 V}{r\cos^2\varphi} - g_x \tan\varphi + g_z \right)$$

$$M_{zz} = \partial_{rr}^2 V$$

$$M_{xy} = \frac{1}{r} \left(\frac{\partial_{\varphi\lambda}^2 V}{r\cos\varphi} + g_y \tan\varphi \right)$$

$$M_{xz} = \frac{1}{r} \left(\partial_{\varphi r}^2 V - g_x \right)$$

$$M_{yz} = \frac{1}{r} \left(\frac{\partial_{\lambda r}^2 V}{\cos\varphi} - g_y \right) \tag{15}$$

where ∂_{ij}^2 $(i, j = \varphi, \lambda, r)$ denotes the second-order partial differentiation with respect to the spatial variables. According to Eq. (6), we also have

$$\partial_{ij}V = \sum_{k=0}^{N}\sum_{n=0}^{N} L_n^k I_{n+2,ij}\rho_k = \sum_{k=0}^{N} w_k^{ij}\rho_k \qquad (16)$$

with

$$w_k^{ij} = \sum_{n=0}^{N} L_n^k I_{n+2,ij} \qquad (17)$$

$$I_{n+2,ij} = G \int_{\lambda_1'}^{\lambda_2'}\int_{\varphi_1'}^{\varphi_2'} \cos\varphi' \ (\partial_{ij}^2\xi_{n+2})\Big|_{r_1'}^{r_2'}\ d\varphi'd\lambda' \qquad (18)$$

The recursive expressions for $\partial_{ij}^2\xi_{n+2}$ are provided in Appendix A. Note that in the case of the computation point being located on the polar axis ($\varphi = \pm 90°$, i.e. $\cos\varphi = 0$), Eqs (11) and (15) will lead to indeterminate values. Therefore, our method cannot calculate the gravitational effects for observation points on the polar axis.

2.4 2D GLQ integration

After the analytical integration along the radial direction, the resultant surface integrals $I_p, I_{p,j}, I_{p,ij}$ in the above equations are then evaluated using the 2D GLQ that approximates the double integral using a weighted sum of the integration kernel,

$$\int_{\lambda_1'}^{\lambda_2'}\int_{\varphi_1'}^{\varphi_2'} f(\varphi',\lambda')d\varphi'd\lambda'$$

$$\approx \frac{\Delta\varphi'\Delta\lambda'}{4}\sum_{n=1}^{\bar{N}}\sum_{m=1}^{\bar{M}} A_n A_m f(\bar{\varphi}_n',\bar{\lambda}_m') \qquad (19)$$

with

$$\bar{\varphi}_n' = \frac{\varphi_1'+\varphi_2'}{2} + \frac{\Delta\varphi'}{2}t_n, \quad \Delta\varphi' = \varphi_2' - \varphi_1'$$

$$\bar{\lambda}_m' = \frac{\lambda_1'+\lambda_2'}{2} + \frac{\Delta\lambda'}{2}t_m, \quad \Delta\lambda' = \lambda_2' - \lambda_1' \qquad (20)$$

where A_n, A_m and t_n, t_m are the Gaussian weights and nodes along the latitudinal and longitudinal directions, respectively. $\bar{N}, \bar{M}$ are the number of Gaussian points.

3 Numerical implementation

Next, we introduce the fast algorithm based on the Toeplitz structure of the weight coefficient matrix to improve the computational efficiency and the 2D adaptive discretization strategy that ensures the accuracy at observation points near the tesseroids.

3.1 Fast algorithm based on a general Toeplitz

We divide the source region into $N_{r'} \times N_{\varphi'} \times N_{\lambda'}$ tesseroids in the radial, latitudinal and longitudinal directions, respectively. For each tesseroid, its geometric centre is $(r_E', \varphi_I', \lambda_J')$ with a mesh interval $\Delta r'$, $\Delta\varphi'$, $\Delta\lambda'$, respectively. According to (6), (12), and (16), the gravitational potential V and its derivatives $\partial_j V$, $\partial_{ij}V$ produced by all mass tesseroids can be expressed as

$$V(r,\varphi,\lambda)$$
$$= \sum_{E=1}^{N_{r'}}\sum_{I=1}^{N_{\varphi'}}\sum_{J=1}^{N_{\lambda'}}\left(\sum_{k=0}^{N} w_k^E(r,\varphi,\varphi_I',\lambda_J'-\lambda)\rho_k^E(\varphi_I',\lambda_J')\right)$$

$$\partial_j V(r,\varphi,\lambda)$$
$$= \sum_{E=1}^{N_{r'}}\sum_{I=1}^{N_{\varphi'}}\sum_{J=1}^{N_{\lambda'}}\left(\sum_{k=0}^{N} w_k^{E,j}(r,\varphi,\varphi_I',\lambda_J'-\lambda)\rho_k^E(\varphi_I',\lambda_J')\right)$$

$$\partial_{ij} V(r,\varphi,\lambda)$$
$$= \sum_{E=1}^{N_{r'}}\sum_{I=1}^{N_{\varphi'}}\sum_{J=1}^{N_{\lambda'}}\left(\sum_{k=0}^{N} w_k^{E,ij}(r,\varphi,\varphi_I',\lambda_J'-\lambda)\rho_k^E(\varphi_I',\lambda_J')\right)$$
$$(21)$$

where ρ_k^E denotes the kth nodal density in the Eth element along the radial direction. w_k^E, $w_k^{E,j}$, and $w_k^{E,ij}$ are the weight coefficients given by (7), (13), and (17), respectively. As can be seen, in the longitudinal direction, the gravitational potential and its derivatives are discrete convolutions of the weight coefficient and the density. Since the expressions for $V, \partial_j V, \partial_{ij}V$ have a similar form, we will focus only on V henceforth for simplicity, and $\partial_j V, \partial_{ij}V$ can finally be obtained by replacing w_k^E with $w_k^{E,j}, w_k^{E,ij}$.

Assume that there are N_λ observation points distributed along the longitudinal direction at $(r,\varphi) = (r_O,\varphi_L)$ with an equal mesh interval $\Delta\lambda = \Delta\lambda'$. Then, we have $\lambda_J' = \lambda_0' + J\Delta\lambda$, $\lambda_P = \lambda_0 + P\Delta\lambda$, and $\lambda_J' - \lambda_P = \hat{\lambda}_0 + (J-P)\Delta\lambda$ where $\hat{\lambda}_0 = \lambda'_0 - \lambda_0, \lambda_0', \lambda_0$ are the starting points of λ' and λ, and J, P are integers. Thus, the gravitational potential in Eq. (21) can be rewritten as

$$\mathbf{V}(r_O,\varphi_L) = \sum_{E=1}^{N_{r'}}\sum_{I=1}^{N_{\varphi'}}\sum_{k=0}^{N} \mathbf{W}_k^E(r_O,\varphi_L,\varphi_I')\boldsymbol{\rho}_k^E(\varphi_I') \qquad (22)$$

where $\mathbf{V} = [V(\lambda_1),\dots,V(\lambda_{N_\lambda})]^{\mathrm{T}}$, $\boldsymbol{\rho}_k^E = [\rho_k^E(\lambda_1'),\dots, \rho_k^E(\lambda'_{N_{\lambda'}})]^{\mathrm{T}}$, and

$$\mathbf{W}_k^E = \begin{bmatrix} w_k^E(\hat{\lambda}_0) & w_k^E(\hat{\lambda}_0 + \Delta\lambda) & \cdots & w_k^E(\hat{\lambda}_0 + (N_{\lambda'} - 1)\Delta\lambda) \\ w_k^E(\hat{\lambda}_0 - \Delta\lambda) & w_k^E(\hat{\lambda}_0) & \cdots & w_k^E(\hat{\lambda}_0 + (N_{\lambda'} - 2)\Delta\lambda) \\ \vdots & \vdots & \vdots & \vdots \\ w_k^E(\hat{\lambda}_0 + (1 - N_\lambda)\Delta\lambda) & w_k^E(\hat{\lambda}_0 + (2 - N_\lambda)\Delta\lambda) & \cdots & w_k^E(\hat{\lambda}_0 + (N_{\lambda'} - N_\lambda)\Delta\lambda) \end{bmatrix} \tag{23}$$

where $\mathbf{W_k^E}$ is the weight coefficient matrix. According to Vogel (2002), a matrix in a form of $\mathbf{W}_k^E$, which is constant along diagonals, is called Toeplitz. Here, $\mathbf{W}_k^E$ has a general Toeplitz form that can be either symmetrical or non-symmetrical and square or non-square. For such a Toeplitz matrix, a product like $\mathbf{W}_k^E \boldsymbol{\rho}_k^E$ can be efficiently computed by FFTs (Sansó and Sampietro 2022):

$$\mathbf{v} = \text{IFFT}\{\text{FFT}[\mathbf{c}] \ast \text{FFT}[\mathbf{s}]\} \tag{24}$$

where $\mathbf{v}$, $\mathbf{c}$, and $\mathbf{s}$ are matrices with a length of $N_\lambda + N_\lambda'$, and

$$\mathbf{c} = \left[w_k^E(\hat{\lambda}_0), \ldots, w_k^E(\hat{\lambda}_0 - (N_\lambda - 1)\Delta\lambda), 0, \right.$$
$$\left. w_k^E(\hat{\lambda}_0 + (N_{\lambda'} - 1)\Delta\lambda), \ldots, w_k^E(\hat{\lambda}_0 + \Delta\lambda) \right]^T$$
$$\mathbf{s} = \left[\rho_k^E(\lambda'_1), \ldots, \rho_k^E(\lambda'_{N_{\lambda'}}), 0, \ldots, 0 \right]^T \tag{25}$$

By extracting the first N_λ components of $\mathbf{v}$, the results of $\mathbf{W}_k^E \boldsymbol{\rho}_k^E$ are obtained. It is noteworthy that only the elements in the first row and column of $\mathbf{W}_k^E$ are required in implementing the algorithm, and thus, the computational cost and memory occupation are significantly reduced as compared to conventional purely summation methods (e.g. Uieda et al. 2016). In fact, the above fast FFT-based calculation is a general version of the method of Zhao et al. (2021). More exactly, the weight coefficient matrix they used is a special case of Eq. (23). In Zhao et al. (2021)'s algorithm, $\mathbf{W_k^E}$ is required to be symmetrical and square. It means that the number of the observation points must be equal to that of the tesseroids and their longitudinal coordinates must coincide with the centres of the tesseroids, i.e. $N_\lambda = N_{\lambda'}$ and $\lambda = \lambda'$ (as shown in Fig. 3a). In practical applications, this may result in undesirable computation and inflexibility. Here, we extend $\mathbf{W_k^E}$ to a general form (see Eq. (23)), which makes the placement of the observation points independent of the discretization of the source region. Our fast algorithm only requires $\Delta\lambda = \Delta\lambda'$ and puts no other constraints on N_λ, $N_{\lambda'}$ and λ, λ' (see Fig. 3b). Hence, it enjoys higher flexibility than the method of Zhao et al. (2021).
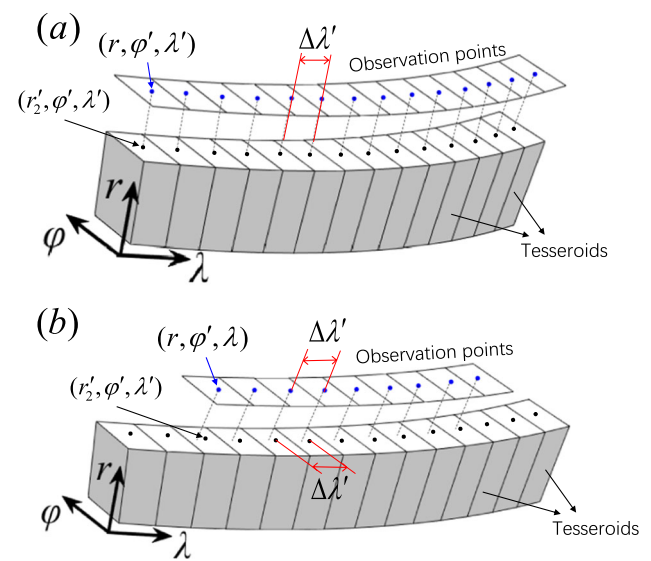
Also we notice from Eq. (1) that, if the observation points align with the source points in the longitudinal direction ($\hat{\lambda}_0 = 0$ and $N_\lambda = N_{\lambda'}$, as shown in Fig. 3a), $\mathbf{W}_k^E$ degenerates to those used by Zhao et al. (2021). In this case, the forward modelling can be further speeded up by utilizing the parity of the gravitational components, that is,

V, g_x, g_z, M_{xx}, M_{yy}, M_{zz} are even functions of $(\lambda' - \lambda)$, while g_y, M_{xy}, M_{yz} are odd functions. For even functions, $\mathbf{W}_k^E$ in Eq. (23) can be reduced to

$$\mathbf{W}_k^E = \begin{bmatrix} w_k^E(0) & w_k^E(\Delta\lambda) & \cdots & w_k^E((N_\lambda - 1)\Delta\lambda) \\ w_k^E(\Delta\lambda) & w_k^E(0) & \cdots & \vdots \\ \vdots & \vdots & \ddots & w_k^E(\Delta\lambda) \\ w_k^E((N_\lambda - 1)\Delta\lambda) & \cdots & w_k^E(\Delta\lambda) & w_k^E(0) \end{bmatrix} \tag{26}$$

Similarly, for the case of odd functions, we have

$$\mathbf{W}_k^E = \begin{bmatrix} 0 & w_k^E(\Delta\lambda) & \cdots & w_k^E((N_\lambda - 1)\Delta\lambda) \\ -w_k^E(\Delta\lambda) & 0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & w_k^E(\Delta\lambda) \\ -w_k^E((N_\lambda - 1)\Delta\lambda) & \cdots & -w_k^E(\Delta\lambda) & 0 \end{bmatrix} \tag{27}$$

Fig. 3 Comparison of the observation points and mesh grids in the longitudinal dimension used by Zhao et al. (2021)' method (**a**) and our method (**b**). In the method of Zhao et al. (2021), the number of the observation points is required to be equal to that of the tesseroids along the longitudinal direction ($N_\lambda = N_{\lambda'}$) and the observation points should align with the centres of the tesseroids in the horizontal directions ($\lambda = \lambda'$). In our fast algorithm, only $\Delta\lambda = \Delta\lambda'$ is required, and one can take $N_\lambda \neq N_{\lambda'}$ and $\lambda \neq \lambda'$
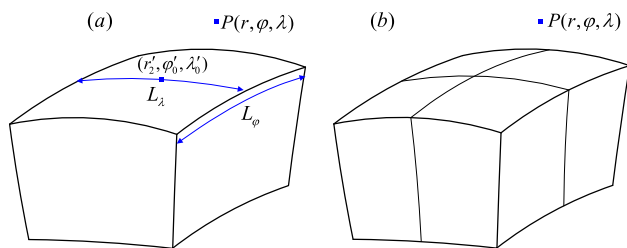
Fig. 4 Illustration of the 2D adaptive discretization. $(r_2', \varphi_0', \lambda_0')$ is the geometric centre of the top surface of the tesseroid, and L_φ, L_λ are the dimensions of the tesseroid in the latitude and longitude. P is the observation point

As can be seen, only the elements in the first row of $\mathbf{W}_k^E$ are required in this case, which can further reduce the computational cost.

3.2 2D adaptive discretization

A 2D adaptive discretization strategy based on Uieda et al. (2016) and Lin and Denker (2019) is also combined in our algorithm to improve the accuracy at points close to the mass sources. The 2D adaptive discretization follows a criterion that a tesseroid is divided along the latitudinal or longitudinal direction when the distance to the computation point is smaller than a constant (D) times the maximum size of the tesseroid (see Fig. 4), that is,

$$\ell_0/\max(L_\varphi, L_\lambda) \geq D \tag{28}$$

where D is called the distance–size ratio. A larger D means a finer division of the tesseroid close to the computation point. ℓ_0 is the distance between the computation point $P(r, \varphi, \lambda)$ and the geometric centre of the upper horizontal surface of the tesseroid $(r_2', \varphi_0', \lambda_0')$:

$$\ell_0 = \sqrt{r^2 + r_2'^2 - 2rr_2' \cos\psi_0}$$
$$\cos\psi_0 = \sin\varphi \sin\varphi_0' + \cos\varphi \cos\varphi_0' \cos(\lambda_0' - \lambda)$$
$$\varphi_0' = (\varphi_1' + \varphi_2')/2, \quad \lambda_0' = (\lambda_1' + \lambda_2')/2 \tag{29}$$

Here the division criterion is slightly different from those used in Uieda et al. (2016) and Lin and Denker (2019), but all these adaptive discretizaion strategies work in a similar way. L_φ and L_λ are the dimensions of the tesseroid in the latitude and longitude:

$$L_\varphi = r_2'(\varphi_2' - \varphi_1'), \quad L_\lambda = r_2' \cos\varphi_1(\lambda_2' - \lambda_1') \tag{30}$$

By checking the inequality given by Eq. (28) for each tesseroid, the algorithm judges whether the tesseroid needs to be discreticized. Following the criterion, the tesseroid is divided in half along the dimension that failed the relation

in a recursive way until Eq. (28) holds for all sub-elements. However, it should be noted that if the observation point coincides with the centre of the top surface of the tesseroid, i.e. $\ell_0 = 0$, the criterion given by Eq. (28) will lead to a "dead circulation". To avoid this problem, we use the following inequality as the stop condition of the discretization when the observation point is located on the surface of the tesseroid:

$$\max(L_\varphi, L_\lambda) < \max\left(L_\varphi^0, L_\lambda^0\right)/\alpha \tag{31}$$

where L_φ^0 and L_λ^0 are the dimensions of the original tesseroid without subdivision, and α is a parameter empirically determined based on numerical tests. The inequality indicates that the adaptive discretization will stop once the maximum horizontal dimension of the current tesseroid is less than that of the original one.

Compared with the 3D adaptive scheme that is implemented along all three dimensions, this 2D adaptive discretization requires a smaller number of sub-elements due to the removal of a dimension from the discretizaion (see Fig. 4). Consequently, it is more computationally efficient.

4 Numerical verification

In this section, we test the performance of the proposed method and determine the optimal values of some key parameters, including the number of GLQ nodes, the distance–size ratio D, and the parameter α. Several spherical shell models with different density variations in depth are used to verify the correctness and accuracy of the proposed method. These shells have a density model $\rho(r')$ and are located on a spherical Earth with inner radius R_1 and outer radius R_2. The closed-form expressions of the gravitational effects for shell models with a polynomial density of arbitrary degree and a exponential density are provided in Appendix B. The codes for the new algorithm are freely available at https://github.com/Yonfou/gravity-modelling-with-polynomial-density.

4.1 Optimal selection of key parameters

The number of GLQ nodes $\bar{M}/\bar{N}$, the distance–size ratio D, and the parameter α are three important parameters that have a significant influence on the computational accuracy and efficiency. The number of GLQ nodes controls the accuracy of the surface integration over a single tessseroid, while D determines how many times each tesseroid will be divided. Besides, α works when the observation point is located on the surface of the tesseroid. It determines the minimum horizontal size of the mesh and hence also indirectly influences the performance of the method. In order to ensure both acceptable numerical accuracy and computation efficiency for the
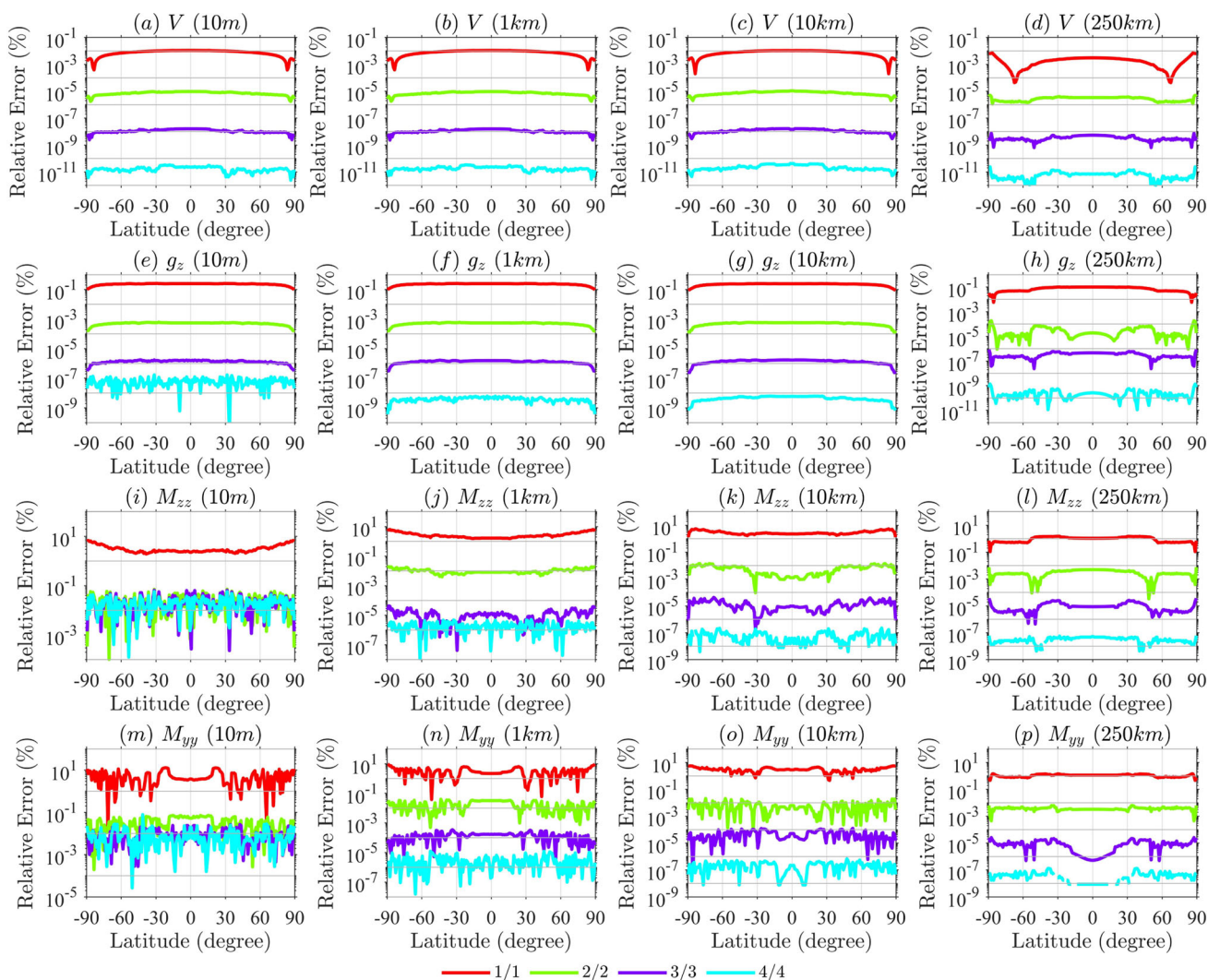
Fig. 5 Relative errors of V (**a–d**), g_z (**e–h**), M_{zz} (**i–l**), and M_{yy} (**m–p**) as a function of the latitude at different observation heights. Different combinations of the Gauss nodes ($\bar{M}/\bar{N} = 1/1, 2/2, 3/3, 4/4$) are tested

proposed algorithm, it is necessary to find the optimal values for these parameters.

We start with a homogeneous shell model with $R_1 = 6271$ km, $R_2 = 6371$ km, and $\rho = 1000$ kg/m^3. In the numerical experiments, four nonzero independent gravitational components V, g_z, M_{zz}, and M_{yy} are considered to investigate the influence of the three parameters. The whole shell is discreticized into $180 \times 360 = 64,800$ tesseroids of size $1° \times 1°$ along the latitudinal and longitudinal directions with one layer in depth. The observation points are located on a spherical surface at a constant height above the top of the shell, having consistent coordinates (φ, λ) with the centres of the tesseroids. In addition, because the magnitudes of the gravitational effects of the spherical shells vary largely, we use the relative errors to measure the computational accuracy instead of the absolute errors here. Moreover, a maximum tolerated relative error of 0.1% is established as

a error threshold in the optimal selection of $\bar{M}/\bar{N}$ and D considering the trade-off between accuracy and computation time.

Figure 5 shows the maximum relative errors between the numerical and analytical solutions calculated using different combinations of the GLQ nodes (i.e., $\bar{M}/\bar{N} = 1/1, 2/2, 3/3, 4/4$) at different observation heights ranging from 10 m to 250 km. As can be seen, the relative errors of the gravitational effects generally decrease as the number of GLQ nodes increases. To achieve an error level of 0.1% for all gravitational components, $\bar{M}/\bar{N} = 2/2$ is the best choice in terms of accuracy and efficiency. Figure 5 clearly shows that the latitude of the observation point has an impact on the numerical results. We can notice that oscillations occur in the relative errors, especially for the gradient components. This phenomenon is caused by the inherent singularity of the integral kernels for the tesseroids near the observation

Fig. 6 Relative errors of V (**a**), g_z (**b**), M_{zz} (**c**), and M_{yy} (**d**) as a function of the observation height at different D varying from 0 (corresponding to the case of no adaptive division) to 5.5
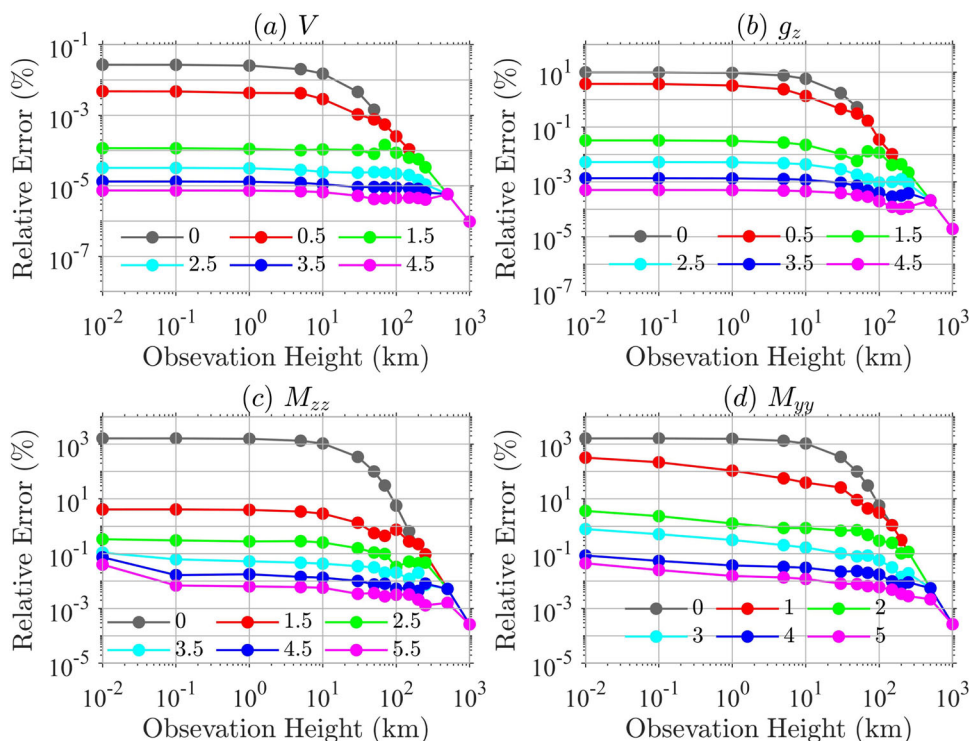
Fig. 7 Relative errors of g_z varying with mesh intervals for observation points located on the shell surface. Different values of D are used
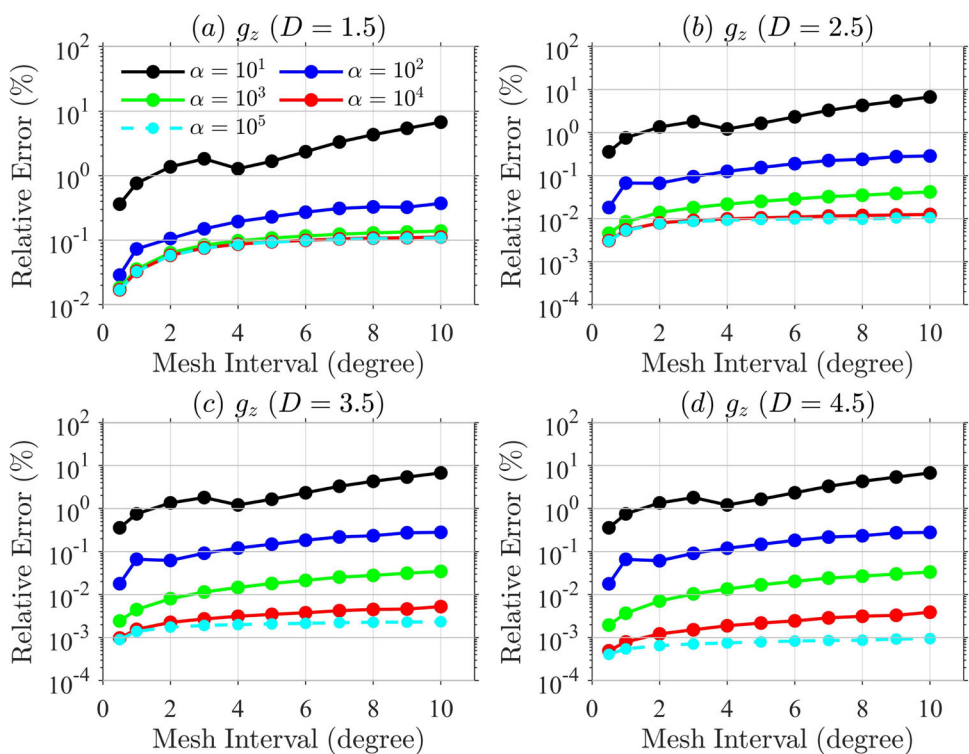

points and the rounding error for those in the far zone (Lin and Denker 2019). The gravitational effects of an observation point are usually calculated by superposing the contributions of all tesseroids over the source region. The tesseroids at different latitudes have different geometrical shapes and con-

sequently may give rise to different approximation errors, causing numerical oscillations.

Next, we select the optimal value for D by investigating the gravitational effects at different observation heights varying from 0.01 to 1000 km. Figure 6 presents the maximum

Fig. 8 Spherical shell model with radial density variation. **a** Density distribution of PREM, **b** exponential density. N marked in **a** denotes the order of the polynomial density
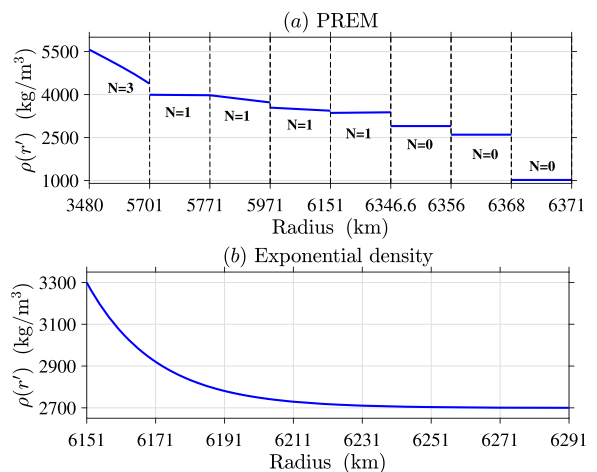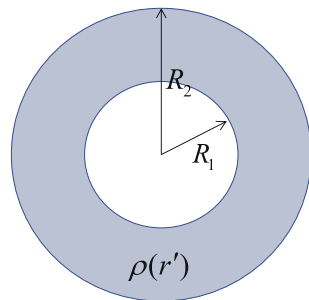
difference between the shell and tesseroid fields with different D that ranges from 0 (i.e., no adaptive division) to 5.5 at each height. We can observe that the computational accuracy improves with increasing observation heights. To be more specific, the relative errors first decrease slowly with the increase of the altitude. Then, after reaching a critical height (denoted as h_c), the error curves for $D > 0$ coincide with the one for $D = 0$ and exhibit a much more rapid decline trend. The critical height h_c for different D can be estimated using the adaptive criterion Eq. (28). It means that the adaptive discretization strategy only works at observation heights lower than h_c. At a fixed observation height, the relative errors decrease as D increases and then become stable. Figure 6 shows that the minimum value for D to achieve a relative error lower than 0.1% is 0 for the gravitational potential, 1.5 for the gravitational acceleration, and 4 for the gravitational gradient, respectively. These optimal values of D are different for different gravitational components due to the different complexity of their integral kernels.

Besides the number of GLQ nodes and the distance–size ratio D, the parameter α also plays a role in the forward modelling when the observation points are located on the surface of the shell. Here, we take g_z as an example to analyse the influence of α, because it is one of the gravitational components continuous on the shell surface. Figure 7 shows the relative errors of g_z as a function of the mesh interval at different α and D. We can see that the relative errors decrease in general as α goes up. At lower D, the errors decline first and then remain almost unchanged when α increases to a value higher than 10^3. The reason is that the minimum size of the tesseroid is larger than $1/\alpha$ times the original dimension without subdivision, and thus, the stop condition is not activated in this situation. When D becomes larger, the stop condition begins to work and the errors exhibit an obvious decrease with the increase of α. For the case of D=1.5 (the optimal value for g_z), $\alpha = 10^3$ is shown to be the best choice for all mesh intervals ranging from 0.5° to 10°. This value of

α is also the optimal one to achieve an error level of 0.1% at higher D (see Fig. 7c, d).

4.2 Validation tests

Next, we validate the correctness and capability of the proposed algorithm using the preliminary reference Earth model (PREM) with different polynomial densities in different layers and a spherical shell model with an exponential density (see Fig. 8).

4.2.1 Polynomial density model

One significant advantage of the new method is its ability to deal with density models in a form of a polynomial up to an arbitrary order along the radial direction. To demonstrate this capability of the method, we construct a shell model composed by eight layers that have different density variations and thicknesses in depth using the well-known preliminary reference Earth model (PREM) developed by Dziewonski and Anderson (1981). Figure 8a presents the radial density distribution of the model, in which the influence of the core is neglected. The total thickness of the eight-layered shell is 2221 km ranging from $R_1 = 3480$ km to $R_2 = 5701$ km (Dziewonski and Anderson 1981). In each layer, the density variation follows a polynomial function of N degree, the value of which is marked in Fig. 8a. In this PREM model, the mean radius of the Earth is set to be $R_E = 6371$ km.

For polynomial density variations, the proposed algorithm can achieve radial integration analytically by setting N in Eq. (4) equal to the order of the real polynomial density, without the requirement of further subdivision along the radial direction. Therefore, we divide the PREM into 180×360 regular meshes along the latitudinal and longitudinal directions with eight layers in the vertical dimension. The observation surface has the same mesh grids as the shell in horizontal directions. Its elevation ranges from 0.01 to 1000 km above

Fig. 9 Relative errors between the numerical and analytical solutions for PREM. **a** Relative errors varying with the latitude at a constant height of 1 km. **b** Maximum relative errors varying with the height of observation. The distance–size ratio D of 0 is used for V, 1.5 for g_z, and 4.0 for M_{zz}, M_{xx}, M_{yy}. The number of GLQ nodes is $\bar{M}/\bar{N} = 2/2$
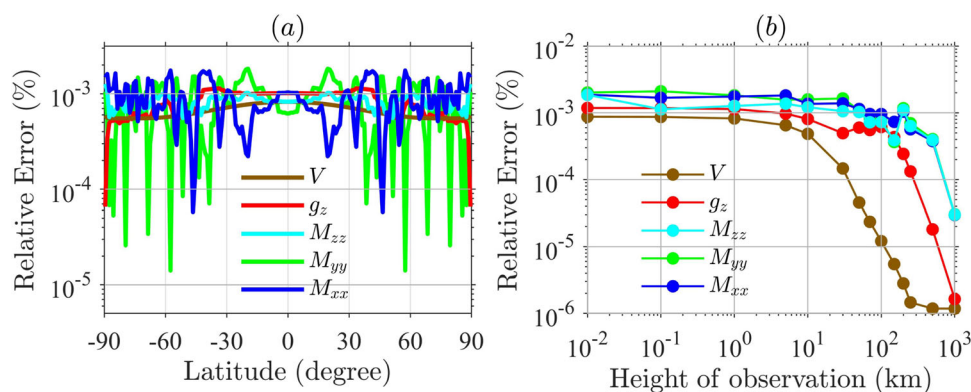
Table 1 Statistics of the errors of V (in m^2/s^2), g_z (in mGal), M_{xx}, M_{yy} and M_{zz} (in E) at a observation height of 1 km for the PREM model

Errors	V	g_z	M_{xx}	M_{yy}	M_{zz}
Analytical solution	4.22×10^7	-6.63×10^5	-1.04×10^3	-1.04×10^3	2.08×10^3
Maximum AE	347.68	7.55	0.018	0.019	0.026
Minimum AE	232.93	0.43	6.01×10^{-4}	1.47×10^{-4}	0.012
Mean AE	277.64	5.37	0.0093	0.0088	0.018
RMS	280.86	5.60	0.010	0.010	0.018
RRMS (%)[1]	6.65×10^{-4}	8.44×10^{-4}	9.83×10^{-4}	9.56×10^{-4}	8.59×10^{-4}

The RRMS error is defined as RRMS$[\phi] = $ RMS$[\phi - \hat{\phi}]/$RMS$[\hat{\phi}] \times 100\%$, where ϕ and $\hat{\phi}$ refer to the numerical and analytical solutions, respectively

the top surface of the spherical shell. The optimal values for the distance–size ratio D and the number of GLQ nodes are used in the calculation of V, g_z, M_{xx}, M_{yy} and M_{zz}.

Figure 9 shows the maximum relative errors varying with the latitude and the height of observation. It can be seen from the figure that the maximum relative errors of the nonzero gravitational components are all blow 0.01% at different latitudes and observation heights. We also provide the statistical properties of the misfits, including the maximum, minimum, and mean absolute errors (AE), the root-mean-square (RMS) error, and the relative root-mean-square (RRMS) error (see Table 1). Obviously, the proposed method works well for all gravitational components. The RMS errors are approximately 280.86 m^2/s^2 for V, 5.60 mGal for g_z, 0.01E for M_{xx} and M_{yy}, and 0.018E for M_{zz}. The RRMS errors for the gravitational components are generally less than 0.001%. These results demonstrate the good accuracy of the proposed method in the case of polynomial density variations.

4.2.2 Exponential density model

Besides the polynomial density models, the new method is also well suited to other kinds of density variations. To test the performance of the algorithm in this aspect, we consider a spherical shell with an exponential density defined as follows (Soler et al. 2019):

$$\rho(r') = Ae^{-b\frac{r'-R_1}{R_2-R_1}} + C \tag{32}$$

with

$$A = \frac{\rho(R_1) - \rho(R_2)}{1 - e^{-b}}, \quad C = \rho(R_1) - A \tag{33}$$

The thickness of the spherical shell is assumed to be 140 km ranging from $R_1 = 6151$ to $R_2 = 6291$ km. We set the values of $\rho(R_1) = 3300$ kg/m^3, $\rho(R_2) = 2700$ kg/m^3, and $b = 7$ (see Fig. 8b). The shell model is discretized into 180×360 regular meshes in the horizontal directions with $N_{r'}$ layer in the radial dimension. The observation points are on a regular grid that aligns with the spherical shell, and its elevation is 1 km.

For such a non-polynomial density model, two approaches can be utilized to achieve desirable accuracy. The first one (Method A) is to apply high-degree polynomial functions to approximate the real density variation and use only one layer in the radial dimension, i.e.,

$$\rho_A(r') = \sum_{n=0}^{N} a_n r'^n \tag{34}$$

where a_n is the polynomial coefficient and N is the order of the polynomial function. The second one (Method B) lies in the combined use of the low-degree polynomial functions and the radial subdivision. We divide the shell into $N_{r'}$ layers in the radial direction and describe the density in each layer

Fig. 10 Maximum relative errors (**a**) and computation time (**b**) varying with N obtained by Method A for the spherical shell model with an exponential density. The distance–size ratio D of 0 is used for V, 1.5 for g_z, and 3.5 for M_{zz}
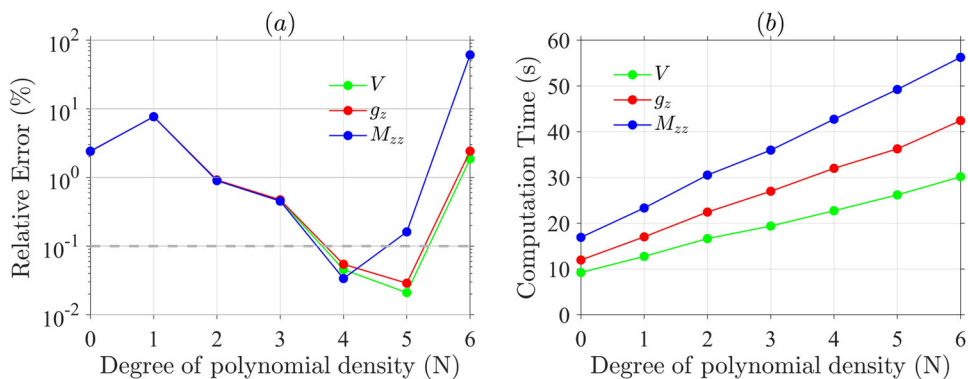
Fig. 11 Maximum relative errors and computation time varying with $N_{r'}$ obtained by Method B for the spherical shell model with an exponential density. **a** and **b** Constant density assumption in each layer. **c** and **d** Quadratic density assumption in each layer. The distance–size ratio D of 0 is used for V, 1.5 for g_z, and 3.5 for M_{zz}
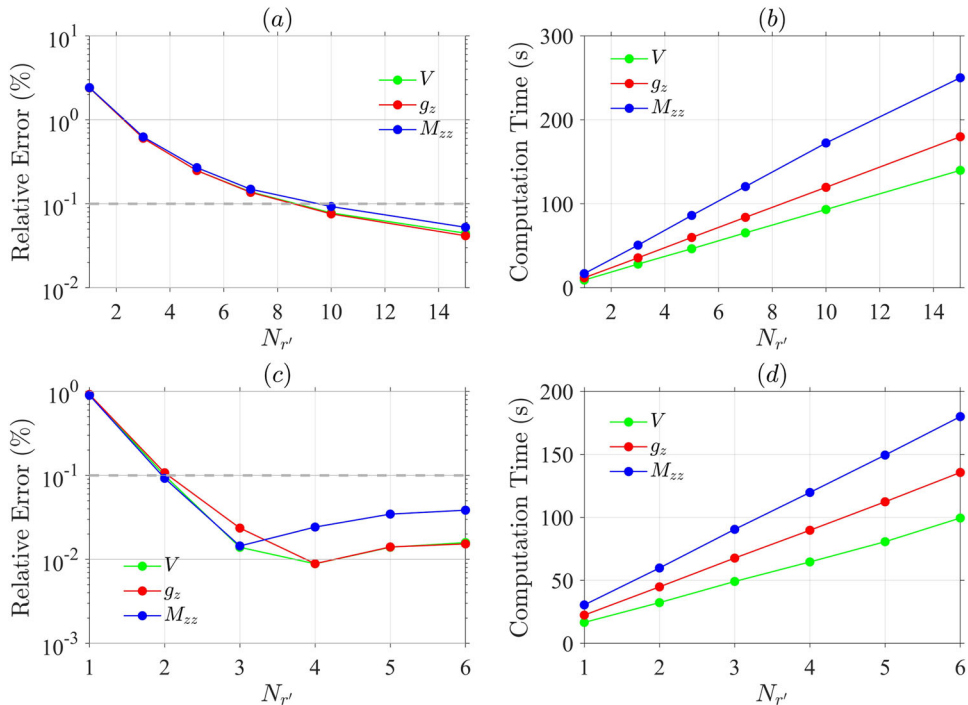
using a low-order polynomial function, that is,

$$\rho_B(r') = \sum_{n=0}^{N^m} a_n^m r'^n, \quad r_1'(m) \le r' \le r_2'(m) \tag{35}$$

where N^m is the order of the polynomial function in the mth layer, a_n^m is the corresponding polynomial coefficient, $r_1'(m)$ and $r_2'(m)$ are the radial coordinates of the bottom and top of the layer, respectively, and $m = 1, 2, \ldots, N_{r'}$. To investigate the computational accuracy and efficiency of the two approaches, we choose D equal to a fixed value, and only explore the errors caused by the radial integration. Here, the gravitational components V, g_z, and M_{zz} are used as examples to investigate the performance of the method in the case of a non-polynomial density.

By applying Nth-order polynomial interpolation to the exponential density, we obtain the relative errors and the com-

putation time of V, g_z, and M_{zz} for Method A (see Fig. 10). As can be seen, a low-degree polynomial function with $N \le 3$ is not sufficient to describe the exponential density variation well, and a polynomial of very high order also results in inaccurate solutions, due to its oscillation properties. To achieve an accuracy with a maximum error of 0.1% in this case, a fourth-order or fifth-order polynomial density assumption is the best for V and g_z, and a fourth-order assumption for M_{zz}. It is also shown that the computation time exhibits a linear increase with increasing N. The calculation of M_{zz} takes the longest time, then followed by g_z and V. Moreover, the difference in time consumption for the three gravity components becomes larger and larger with the increase of N.

Figure 11 shows the relative errors and the computation time as a function of $N_{r'}$ (the number of layers used in the radial direction) for Method B based on the same model. By assuming that the density is constant in each layer, a minimum number of $N_{r'} = 10$ is required to achieve an acceptable

Fig. 12 Comparison of **a** the maximum relative errors at different observation heights and **b** the relative computation time of g_z with different mesh intervals. The computation time is normalized by the slowest calculation time (890,964.72 s)
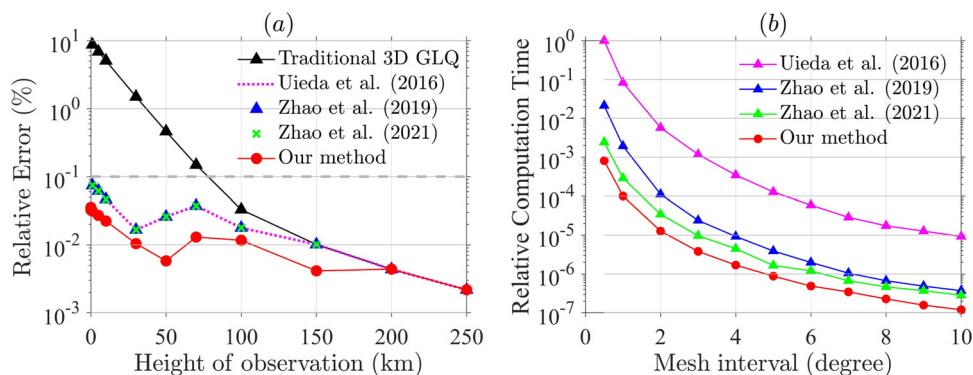
Table 2 Comparison of the computation time and the memory occupation of the proposed method and the 3D GLQ algorithms developed by Uieda et al. (2016), Zhao et al. (2019), and Zhao et al. (2021)

Methods	Computation time (s)	Memory occupation (GB)
Uieda et al. (2016)	890,964.72	537.48
Zhao et al. (2019)	16,243.24	0.75
Zhao et al. (2021)	1965.03	0.75
Our method	722.15	0.1

All tests are carried out on a PC with i7-7700hq CPU and 16 GB memory, but note that the memory occupation for the method Uieda et al. (2016) is an estimated value by Zhao et al. (2021). In the forward modelling by the software "Tesseroids" (Uieda et al. 2016), it does not need to store the vast kernel matrix. However, the storage of the dense kernel matrix is often required in an inversion

accuracy (0.1%) and the corresponding computation time is 93.21 s for V, 119.6 s for g_z and 172.4 s for M_{zz}. For a quadratic density assumption, however, $N_{r'} = 2$ is adequate. In comparison with the constant density model, less calculation time is required in this case, that is, 32.33 s for V, 44.84 s for g_z and 59.82 s for M_{zz}. Therefore, the quadratic density assumption is more appropriate than the constant one in terms of both accuracy and efficiency for the exponential density variation considered.

Comparing Method A with Method B, the former is superior in computational efficiency in this example. It takes only 22.72 s for V, 31.99 s for g_z and 42.72 s for M_{zz} to reach an accuracy of 0.1%. However, for highly variable density variations, Method B will be more beneficial, since the use of high-degree polynomial functions may cause oscillation and result in unstable results. On the whole, although our method considers up only to polynomial density models, its application to non-polynomial density models is also straightforward.

4.3 Comparison with the 3D GLQ methods

This section is aimed at comparing the proposed method with the commonly used tesseroid-based 3D GLQ methods in terms of accuracy and efficiency. Based on the same homogeneous shell model used in Sect. 4.1, we calculate the vertical component of the gravity acceleration g_z using the traditional 3D GLQ method (no adaptive division), Uieda et al. (2016)'s method, Zhao et al. (2019)'s method, Zhao

et al. (2021)'s method, and the proposed method, respectively. A same distance–size ratio of $D = 1.5$ is used for the 3D GLQ methods and the proposed method in this comparison analysis. The observation surface is located on a regular grid at an elevation ranging from 1 to 250 km (observation height of GOCE satellite (Cesare et al. 2010)) above the top surface of the shell.

Figure 12a compares the maximum relative errors obtained by the 3D GLQ methods and the proposed method as a function of the height of observation. It is observed that the results obtained by Uieda et al. (2016), Zhao et al. (2019), and Zhao et al. (2021) are exactly the same. In contrast, the proposed method achieves better accuracy at the near zone as compared to the 3D GLQ methods. In the far zone, the relative errors for all methods are almost the same and less than 0.01%. Therefore, the advantage of the proposed method in accuracy is verified.

We also compare the computational efficiency of both methods for different horizontal mesh intervals ranging from 0.5° to 10°. We divide the spherical shell into 10 layers in depth, and place the observation surface at a height of 10 km above the top surface of the shell. The relative computation time normalized by the slowest calculation varying with the mesh interval is presented in Fig. 12b. The results show that at small mesh intervals less than 2°, the calculation speed of the proposed algorithm is sped up by 3 orders of magnitude compared with Uieda et al. (2016)'s method, 1.5 order of magnitude compared with Zhao et al. (2019)'s method, and more than half compared with Zhao et al. (2021)'s
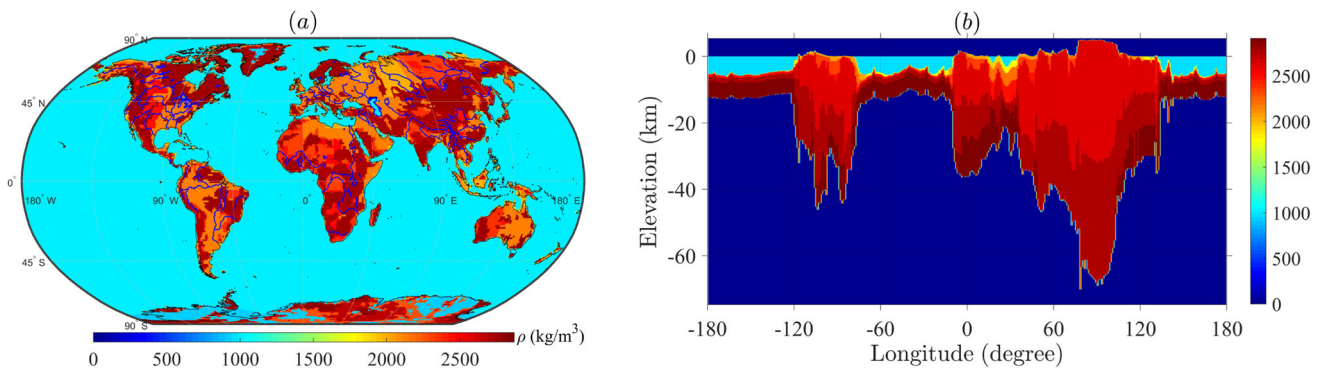
Fig. 13 CRUST1.0 gloabal crustal model. **a** Global density distribution at 10 m below mean Earth radius. **b** Density section along the latitude of 32.5° N

improved method. Table 2 lists the computation time and memory occupation for a discretization size of $360 \times 720 \times 10$ with $D = 1.5$. Compared with the fast algorithm developed by Zhao et al. (2021), the calculation time of the proposed method is further decreased by more than half, and the memory requirement is significantly reduced to 0.1 GB by seven times. This comparison further demonstrates the capability of the proposed method in fast calculation.

5 Application

In this section, we apply the new algorithm in the calculation of gravitational effects of the CRUST1.0, a global model of Earth crust on a 1 by 1 degree grid (Laske et al. 2013), so as to illustrate the reliability and flexibility of the method in practice. The CRUST1.0 incorporates the global sediment thickness and consists of eight-layered crustal profile, including water, ice, upper sediments, middle sediments, lower sediments, upper crust, middle crust, and lower crust. The model has the latitude ranging from $-89.5°$ to $89.5°$ and the longitude from $-179.5°$ to $179.5°$ with an equal interval of $1°$. Its elevation ranges from -74.81 km below mean Earth radius to 5.41 km above mean Earth radius. The density distribution of the CRUST1.0 global crustal model is shown in Fig. 13. Note that the influence of the mantle below the Moho is neglected here.

In order to evaluate the gravitational effects of the CRUST1.0 model efficiently and accurately, we use different mesh grids at different latitudes (Fig. 14). According to the CRUST1.0 model, the latitude ranges from $-89.5°$ to $89.5°$ and the longitude ranges from $-179.5°$ to $179.5°$ with $\Delta\varphi' = \Delta\lambda' = 1°$. Therefore, only the radial subdivision needs to be considered. To exactly describe the density variation in the radial direction, we obtain the sampling points for each $r - \lambda$ plane using the union of the layer boundaries given by the CRUST1.0 model (as shown in Fig. 15a). Note that for this model, the density is constant within each layer. Because
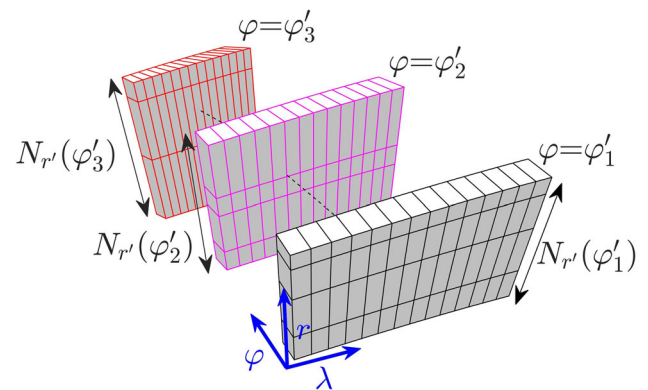

Fig. 14 Illustration of the model discretization at different latitudes. $N_{r'}$ denotes the number of elements in the radial direction and varies with latitudes

the density distribution of the CRUST1.0 model is quite different on different $r - \lambda$ planes, the number of layers ($N_{r'}$) varies with latitudes. Figure 15b shows the $N_{r'}$ we used along the vertical direction at different latitudes for the CRUST1.0 model. As can be seen, $360 \times \sum_{I=1}^{180} N_{r'}(\varphi'_I)$ tesseroids are required to build up the global crustal model.

The observation points are located at a height of 10 km above mean Earth radius and align with the geometric centres of the top horizontal surface of the tesseroids. Figures 16 and 17 show the numerical results for the gravitational vector fields and the Marussi tensor components calculated by the proposed method. To verify the correctness of the results, we also compare our solutions with those obtained by the 3D GLQ method based on a refined regular discretization of $180 \times 360 \times 8022$ (with a interval of 10m in depth). In the GLQ method, the distance–size ratio D is set to be 8 for all gravitational components, and in the proposed algorithm, $D = 4$ is used. The statistics of the forward results calculated by the two methods are listed in Table 3. It is observed that the maximum RRMS error between the two methods is $1.3 \times 10^{-5}\%$ for V, 0.001% for the gravitational vector, and 0.022% for the gravitational gradient. This result further

Fig. 15 Radial discretization for the CRUST1.0 model. **a** Determination of the sampling points in depth. **b** Number of radial elements ($N_{r'}$) varying with latitude
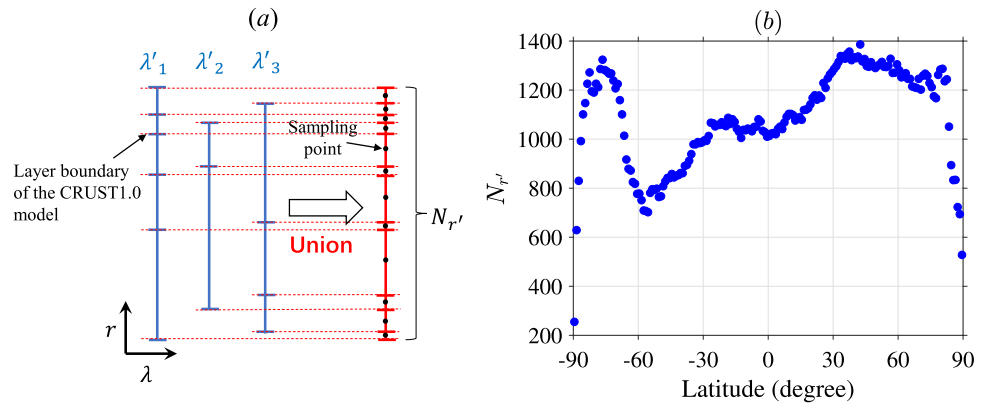
Table 3 Statistics of the forward results of the gravitational potential (in s^2/m^2), vector (in mGal) and Marussi tensor (in E) for the CRUST1.0 global crustal model

	Algorithm	Max	Min	Mean	RMS	RRMS (%)
V	Our method	391,878.52	244,680.49	303,607.65	305,823.68	
	3D GLQ	391,878.62	244,680.51	303,607.68	305,823.70	1.0×10^{-5}
	Difference	0.10	0.02	0.03	0.02	
g_x	Our method	4257.028	-3103.93	165.8723	1101.919	
	3D-GLQ	4257.034	-3103.94	165.8723	1101.922	4.8×10^{-4}
	Difference	0.006	-0.01	9.4×10^{-6}	0.003	
g_y	Our method	5183.82	-3448.489	-6.9×10^{-14}	966.635	
	3D-GLQ	5183.84	-3448.495	-8.4×10^{-14}	966.636	2.9×10^{-4}
	Difference	0.02	-0.006	-1.5×10^{-14}	0.001	
g_z	Our method	-2904.31	$-11,066.44$	-4915.63	5220.15	
	3D-GLQ	-2904.32	$-11,066.52$	-4915.65	5220.18	5.9×10^{-4}
	Difference	-0.01	-0.08	-0.02	0.03	
M_{xx}	Our method	166.059	-158.998	-7.695	23.1688	
	3D-GLQ	166.057	-158.997	-7.697	23.1700	0.009
	Difference	-0.002	0.001	-0.002	0.0012	
M_{yy}	Our method	185.859	-191.534	-8.110	24.107	
	3D-GLQ	185.857	-191.539	-8.111	24.108	0.008
	Difference	-0.002	-0.005	-0.001	0.001	
M_{zz}	Our method	279.163	-238.818	15.805	40.5799	
	3D-GLQ	279.174	-238.816	15.808	40.5821	0.009
	Difference	0.011	0.002	0.003	0.0022	
M_{xy}	Our method	65.6905	-82.498	7.4×10^{-13}	9.3633	
	3D-GLQ	65.6909	-82.497	-6.5×10^{-13}	9.3632	0.002
	Difference	0.0004	0.001	-1.4×10^{-12}	-0.0001	
M_{xz}	Our method	85.190	-144.980	0.61124	14.787	
	3D-GLQ	85.192	-144.983	0.61125	14.788	0.005
	Difference	0.002	-0.003	1×10^{-5}	0.001	
M_{yz}	Our method	191.9138	-189.346	-7.7×10^{-13}	19.3019	
	3D-GLQ	191.9142	-189.345	-1.0×10^{-12}	19.3023	0.003
	Difference	0.0004	0.001	-2.3×10^{-13}	0.0004	
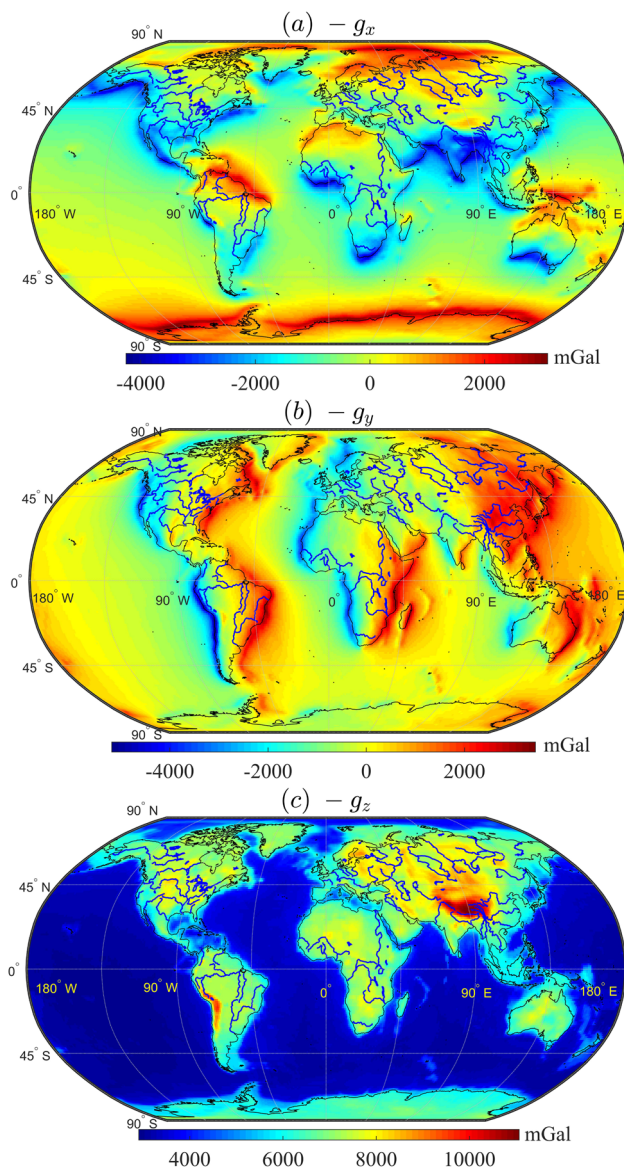
Fig. 16 Gravitational vector calculated by the proposed method for the CRUST1.0 global crustal model. Since the positive direction of the z axis is pointing upward, we use $-\mathbf{g}$ instead of $\mathbf{g}$

demonstrates the reliability and practicability of the proposed method.

6 Conclusions

An accurate and efficient method for 3D large-scale gravitational forward modelling is developed on the basis of tesseroids with a polynomial density up to an arbitrary order. The new method achieves high computational accuracy and efficiency by taking advantage of the recursive analytical expressions for the radial integration with a polynomial density and a fast discrete convolution algorithm based on the

general Toeplitz form of the weight coefficient matrix. The proposed method first transformed the triple integral into a surface integral through analytical integration along the radial direction and then numerically evaluated the surface integral by GLQ with a 2D adaptive discretization. A stop condition controlled by a scalar α is introduced into the adaptive strategy to avoid it from entering a dead circulation in the case of observation points located on the surface of the tesseroid. This also allows more control over errors when too many divisions are necessary for surface observation points. Next, the FFT-based fast algorithm is adopted to speed up the forward modelling and to reduce the memory in the computation of the discrete convolution between the density and the weight coefficient along the longitudinal dimension. Comparing with existing FFT-based convolution methods used in gravity modelling, this algorithm has a more general form that can be applicable for symmetric or non-symmetric and square or non-square weight coefficient matrices. It only requires $\Delta\lambda = \Delta\lambda'$ and puts no other constraints on the position and number of observation points, which means that one can take $\lambda \neq \lambda'$ and $N_\lambda \neq N_\lambda'$. However, $\Delta\lambda = \Delta\lambda'$ is also to some extent a limitation resulting from the fast calculation. Another limitation of the proposed method is that the explicit forms of the gravitational vector and tensor suffer from the polar singularity induced by the spherical coordinate system, and hence, our method cannot calculate the gravitational effects at observation points with $\varphi = \pm 90°$.

In the proposed method, the numerical accuracy and efficiency are mainly determined by three parameters: the number of GLQ nodes $\bar{M}/\bar{N}$, the distance–size ratio D, and the parameter α. To guarantee a satisfying accuracy while maintaining the computational efficiency, we carried out an optimal selection on $\bar{M}/\bar{N}$, D, and α based on a homogeneous shell model. The numerical experiments showed that to achieve a maximum relative error of 0.1%, $\bar{M}/\bar{N} = 2/2$ is often sufficient, while the optimal value of D is 0 for the gravitational potential, 1.5 for the gravitational vector, and 4 for the Marussi tensor. For the parameter α, which controls the accuracy of surface observation points, 10^3 is generally a good choose.

One significant advantage of the proposed method is its ability to deal with a polynomial density up to an arbitrary degree. It can obtain accurate results for a tesseroid with a Nth-order polynomial density using only one layer in the radial direction. Our numerical examples showed that for the PREM model with a polynomial density of different order in different layers, the proposed method can achieve an accuracy with a relative error lower than 0.01% for an observation height of 1 km. For a shell model with a non-polynomial exponential density, the method still works well. The results showed that by using a fourth-order polynomial function to fit the exponential density, the relative errors between the numerical and analytical solutions are about 0.05%. It also
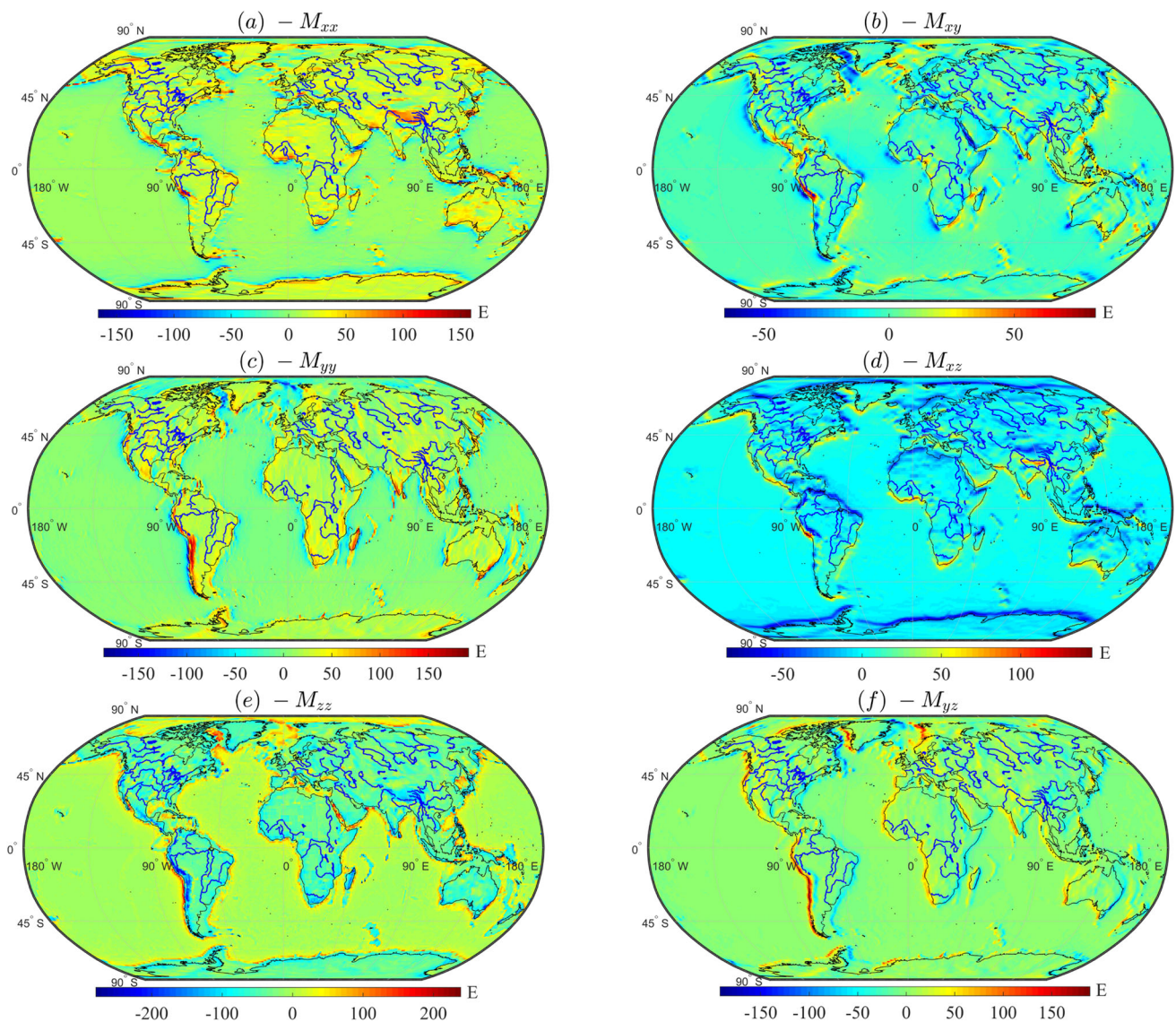
Fig. 17 Gravitational gradient calculated by the proposed method for the CRUST1.0 global crustal model. Since the positive direction of the z axis is pointing upward, we use $-\mathbf{M}$ instead of $\mathbf{M}$

showed that increasing the number of layers in the radial dimension and using a lower-degree polynomial density in each layer can also achieve good results, but require more calculation time. The application to the global CRUST1.0 model with undulating topography further demonstrated the reliability and applicability of the new method in real cases. For the CRUST1.0 model, the numerical results obtained from the proposed method and the 3D GLQ method with a dense mesh are nearly the same and the RRMS error between the two methods is lower than 0.002%.

The comparison analysis of the proposed method and the existing tesseroid-based 3D GLQ methods also verified the good accuracy and efficiency of the new algorithm. The numerical results showed that the new method has higher precision than the 3D GLQ methods at the near zone and

exhibits the same accuracy at the far zone. In addition, the computational efficiency of the proposed algorithm is significantly improved by 3 orders of magnitude as compared to Uieda et al. (2016)'s method, by 1.5 order of magnitude as compared to Zhao et al. (2019), and by more than half as compared to Zhao et al. (2021)'s improved method. Further, the memory occupation for the proposed algorithm is also substantially decreased by seven times.

In conclusion, the newly developed method is more advantageous over the tesseroid-based 3D GLQ methods in both accuracy and efficiency, and also in its capability in dealing with polynomial and non-polynomial density models. Therefore, we believe that our method could have more potentials in large-scale gravity forward and inverse problems.

Appendix A: Analytical expressions for ξ_p, $\partial_j \xi_p$, and $\partial_{ij}^2 \xi_p$

The radial integrals $\partial_j \xi_{n+2}$ $(n = 0, 1, ..., N, \ j = \varphi, \lambda, r)$ in Eq. (14) can be analytically expressed as

$$\partial_\varphi \xi_{n+2} = r \eta_{n+3} \partial_\varphi \cos \psi$$
$$\partial_\lambda \xi_{n+2} = r \eta_{n+3} \partial_\lambda \cos \psi$$
$$\partial_r \xi_{n+2} = \eta_{n+3} \cos \psi - \eta_{n+2} r \qquad (A1)$$

where

$$\eta_k = \int \frac{r'^k}{\ell^3} dr', \quad k = 0, 1, \ldots, n+3 \qquad (A2)$$

with

$$\eta_0 = \begin{cases} -\frac{\text{sign}(r'-r)}{2(r'-r)^2}, & \cos \psi = 1 \\ -0.5(r'+r)^{-2}, & \cos \psi = -1 \\ \frac{(r'r - \cos \psi)}{r(1-\cos^2\psi)\ell}, & \cos \psi \neq \pm 1 \end{cases}$$

$$\eta_2 = -\frac{r'}{\ell} + r \cos \psi \cdot \eta_1 + \xi_0$$

$$\eta_k = \frac{1}{k-2} \left(\frac{r'^{k-1}}{\ell} + (2k-3)r \cos \psi \eta_{k-1} \right.$$
$$\left. - (k-1)r^2 \eta_{k-2} \right)$$
$$k > 0 \quad \text{and} \quad k \neq 2 \qquad (A3)$$

The radial integrals $\partial_{ij}^2 \xi_{n+2}$ $(n = 0, 1, \ldots, N, \ i, j = \varphi, \lambda, r)$ in Eq. (18) have the following analytical form:

$$\partial_{\varphi\varphi}^2 \xi_{n+2} = r \partial_{\varphi\varphi}^2 \cos \psi \eta_{n+3} + 3r^2 (\partial_\varphi \cos \psi)^2 \zeta_{n+4}$$
$$\partial_{\lambda\lambda}^2 \xi_{n+2} = r \partial_{\lambda\lambda}^2 \cos \psi \eta_{n+3} + 3r^2 (\partial_\lambda \cos \psi)^2 \zeta_{n+4}$$
$$\partial_{rr}^2 \xi_{n+2} = 2\eta_{n+2} - 3 \left(1 - \cos^2 \psi \right) \zeta_{n+4}$$
$$\partial_{\varphi\lambda}^2 \xi_{n+2} = r \partial_{\varphi\lambda}^2 \cos \psi \eta_{n+3} + 3r^2 \left(\partial_\varphi \cos \psi \partial_\lambda \cos \psi \right) \zeta_{n+4}$$
$$\partial_{\varphi r}^2 \xi_{n+2} = \partial_\varphi \cos \psi \left(\eta_{n+3} - 3r(r\zeta_{n+3} - \cos \psi \zeta_{n+4}) \right)$$
$$\partial_{\lambda r}^2 \xi_{n+2} = \partial_\lambda \cos \psi \left(\eta_{n+3} - 3r(r\zeta_{n+3} - \cos \psi \zeta_{n+4}) \right)$$
$$(A4)$$

where

$$\zeta_k = \int \frac{r'^k}{\ell^5} dr', \quad k = 0, 1, \ldots, n+4 \qquad (A5)$$

with

$$\zeta_0 = \begin{cases} -\frac{\text{sign}(r'-r)}{4(r'-r)^4}, & \cos \psi = 1 \\ -0.25(r'+r)^{-4}, & \cos \psi = -1 \\ \frac{r'r - \cos \psi}{3r(1-\cos^2\psi)} \left(\frac{1}{\ell^3} + \frac{2}{\ell(1-\cos^2\psi)r^2} \right), & \cos \psi \neq \pm 1 \end{cases}$$

$$\zeta_4 = -\frac{r'^3}{3\ell^3} + r \cos \psi \zeta_3 + \eta_2$$

$$\zeta_k = \frac{1}{k-4} \left(\frac{r'^{k-1}}{\ell^3} + (2k-5)r \cos \psi \zeta_{k-1} \right.$$
$$\left. - (k-1)r^2 \zeta_{k-2} \right)$$
$$k > 0 \quad \text{and} \quad k \neq 4 \qquad (A6)$$

and

$$\partial_\varphi \cos \psi = \cos \varphi \sin \varphi' - \sin \varphi \cos \varphi' \cos(\lambda' - \lambda)$$
$$\partial_\lambda \cos \psi = \cos \varphi \cos \varphi' \sin(\lambda' - \lambda)$$
$$\partial_{\varphi\varphi}^2 \cos \psi = -\cos \psi$$
$$\partial_{\lambda\lambda}^2 \cos \psi = -\cos \varphi \cos \varphi' \cos(\lambda' - \lambda)$$
$$\partial_{\varphi\lambda}^2 \cos \psi = -\sin \varphi \cos \varphi' \sin(\lambda' - \lambda) \qquad (A7)$$

Appendix B: Analytical solutions for spherical shell

For a spherical shell with a radially varying density $\rho(r')$, the analytical expressions for the gravitational potential V, vector **g**, and Marussi tensor **M** on any point with a radial distance r above the shell can be expressed as

$$V = \frac{4\pi G}{\ell} \gamma \qquad (B8)$$

$$g_i = 4\pi G\gamma \frac{\Delta_i}{\ell^3} \tag{B9}$$

$$M_{ij} = 4\pi G\gamma \left(\frac{3\Delta_i \Delta_j}{\ell^5} - \frac{\delta_{ij}}{\ell^3} \right) \tag{B10}$$

with

$$\ell = \sqrt{\Delta_x^2 + \Delta_y^2 + \Delta_z^2}$$

$$\Delta_x = r_0 \left[\cos\varphi \sin\varphi_0 - \sin\varphi \cos\varphi_0 \cos(\lambda_0 - \lambda) \right]$$

$$\Delta_y = r_0 \cos\varphi_0 \sin(\lambda_0 - \lambda)$$

$$\Delta_z = r_0 \cos\psi - r$$

$$\cos\psi = \sin\varphi \sin\varphi_0 + \cos\varphi \cos\varphi_0 \cos(\lambda_0 - \lambda) \tag{B11}$$

where $i, j = x, y, z$, δ_{ij} is the Kronecker delta function, $(r_0, \varphi_0, \lambda_0)$ is the geometrical centre of the spherical shell, and

$$\gamma = \int_{R_1}^{R_2} r'^2 \rho(r') \mathrm{d}r' \tag{B12}$$

where R_1 and R_2 are the inner and outer radius of the shell. When the geometrical centre of the shell is located at the origin, i.e. $(r_0, \varphi_0, \lambda_0) = (0, 0, 0)$, Eqs (B8) to (B10) further reduce to

$$V = \frac{4\pi G}{\ell}\gamma \tag{B13}$$

$$g_z = -\frac{4\pi G}{\ell^2}\gamma \tag{B14}$$

$$M_{xx} = M_{yy} = -\frac{4\pi G}{\ell^3}\gamma, \quad M_{zz} = \frac{8\pi G}{\ell^3}\gamma \tag{B15}$$

with $g_x = g_y = 0$, and $M_{xy} = M_{xz} = M_{yz} = 0$.

For a polynomial density function of N degree, $\rho(r') = \sum_{n=0}^{N} A_n r'^n$, γ is given by

$$\gamma = \sum_{n=0}^{N} \frac{A_n}{n+3} \left(R_2^{n+3} - R_1^{n+3} \right) \tag{B16}$$

For a exponential density function $\rho(r') = Ae^{-B(r'-R_1)} + C$, γ is as follows:

$$\gamma = \frac{C}{3}(R_2^3 - R_1^3) + \frac{A}{B^3} \left[\left(R_1^2 B^2 + 2R_1 B + 2 \right) - \left(R_2^2 B^2 + 2R_2 B + 2 \right) e^{-B(R_2 - R_1)} \right] \tag{B17}$$

with

$$A = \frac{\rho(R_1) - \rho(R_2)}{1 - e^{-b}}, \quad B = \frac{b}{R_2 - R_1},$$
$$C = \rho(R_1) - A \tag{B18}$$

where R_1 is the inner radius of the shell and R_2 the outer radius.

The analytical solutions for the gravitational effects of PREM are obtained by summing the contributions of the spherical shells that form it.

References

Anderson EG (1976) The effect of topography on solutions of Stokes' problem. School of Surveying, University of New South Wales. ISBN 100858390213

Benedek J, Papp G, Kalmár J (2018) Generalization techniques to reduce the number of volume elements for terrain effect calculations in fully analytical gravitational modelling. J Geod 92(4):361–381. https://doi.org/10.1007/s00190-017-1067-1

Cesare S, Aguirre M, Allasio A et al (2010) The measurement of Earth's gravity field after the GOCE mission. Acta Astronaut 67:702–712. https://doi.org/10.1016/j.actaastro.2010.06.021

Chai Y, Hinze W (1988) Gravity inversion of an interface above which the density contrast varies exponentially with depth. Geophysics 53(6):837–845. https://doi.org/10.1190/1.1442518

Chen L, Liu L (2019) Fast and accurate forward modelling of gravity field using prismatic grids. Geophys J Int 216(2):1062–1071. https://doi.org/10.1093/gji/ggy480

Conway JT (2015) Analytical solution from vector potentials for the gravitational field of a general polyhedron. Celest Mech Dyn Astron 121(1):17–38. https://doi.org/10.1007/s10569-014-9588-x

D'Urso MG (2014) Analytical computation of gravity effects for polyhedral bodies. J Geod 88(1):13–29. https://doi.org/10.1007/s00190-013-0664-x

D'Urso MG (2014) Gravity effects of polyhedral bodies with linearly varying density. Celest Mech Dyn Astron 120(4):349–372. https://doi.org/10.1007/s10569-014-9578-z

D'Urso MG, Trotta S (2017) Gravity anomaly of polyhedral bodies having a polynomial density contrast. Surv Geophys 38(4):781–832. https://doi.org/10.1007/s10712-017-9411-9

Dziewonski AM, Anderson DL (1981) Preliminary reference Earth model. Phys Earth Planet Inter 25(4):297–356. https://doi.org/10.1016/0031-9201(81)90046-7

Fukushima T (2018) Accurate computation of gravitational field of a tesseroid. J Geod 92(12):1371–1386. https://doi.org/10.1007/s00190-018-1126-2

Grombein T, Seitz K, Heck B (2013) Optimized formulas for the gravitational field of a tesseroid. J Geod 87(7):645–660. https://doi.org/10.1007/s00190-013-0636-1

Hamayun Prutkin I, Tenzer R (2009) The optimum expression for the gravitational potential of polyhedral bodies having a linearly varying density distribution. J Geod 83(12):1163–1170. https://doi.org/10.1007/s00190-009-0334-1

Heck B, Seitz K (2007) A comparison of the tesseroid, prism and point-mass approaches for mass reductions in gravity field modelling. J Geod 81(2):121–136. https://doi.org/10.1007/s00190-006-0094-0

Heiskanen WA, Moritz H (1967) Physical geodesy. Bull Geod 41(4):491–492. https://doi.org/10.1007/BF02525647

Holstein H (2003) Gravimagnetic anomaly formulas for polyhedra of spatially linear media. Geophysics 68(1):157–167. https://doi.org/10.1190/1.1543203

Holstein H, Ketteridge B (1996) Gravimetric analysis of uniform polyhedra. Geophysics 61(2):357–364. https://doi.org/10.1190/1.1443964

Kennett BLN (1998) On the density distribution within the Earth. Geophys J Int 132(2):374–382. https://doi.org/10.1046/j.1365-246X.1998.00451.x

Ku CC (1977) A direct computation of gravity and magnetic anomalies caused by 2- and 3-Dimensional bodies of arbitrary shape and arbitrary magnetic polarization by equivalent-point method and a simplified cubic spline. Geophysics 42(3):610. https://doi.org/10.1190/1.1440732

Kumagai N (1933) Density distribution and compressibility in the Earth's crust and compensating density after Pratt's hypothesis on isostasy. Part I. Jpn J Astron Geophys 11:117

Laske G, Masters G, Ma Z, et al (2013) Update on CRUST1.0-A 1-degree global model of Earth's crust. In: EGU general assembly conference abstracts, EGU general assembly conference abstracts, pp EGU2013–2658

Li Z, Hao T, Xu Y et al (2011) An efficient and adaptive approach for modeling gravity effects in spherical coordinates. J Appl Geophys 73(3):221–231. https://doi.org/10.1016/j.jappgeo.2011.01.004

Lin M, Denker H (2019) On the computation of gravitational effects for tesseroids with constant and linearly varying density. J Geod 93(5):723–747. https://doi.org/10.1007/s00190-018-1193-4

Lin M, Denker H, Müller J (2020) Gravity field modeling using tesseroids with variable density in the vertical direction. Surv Geophys 41(4):723–765. https://doi.org/10.1007/s10712-020-09585-6

Marotta AM, Seitz K, Barzaghi R et al (2019) Comparison of two different approaches for computing the gravitational effect of a tesseroid. Stud Geophys Geod 63(3):321–344. https://doi.org/10.1007/s11200-018-0454-2

Maxant J (1980) Variation of density with rock type, depth, and formation in the Western Canada basin from density logs. Geophysics 45(6):1061. https://doi.org/10.1190/1.1441107

Nagy D, Papp G, Benedek J (2000) The gravitational potential and its derivatives for the prism. J Geod 74(7):552–560. https://doi.org/10.1007/s001900000116

Pohanka V (1988) Optimum expression for computation of the gravity field of a homogeneous polyhedral BODY1. Geophys Prospect 36(7):733–751. https://doi.org/10.1111/j.1365-2478.1988.tb02190.x

Qiu L, Chen Z (2020) Gravity field of a tesseroid by variable-order Gauss–Legendre quadrature. J Geod 94(12):114. https://doi.org/10.1007/s00190-020-01440-1

Rao CV, Chakravarthi V, Raju ML (1993) Parabolic density function in sedimentary basin modelling. Pure Appl Geophys 140(3):493–501. https://doi.org/10.1007/BF00876967

Ren Z, Chen C, Pan K et al (2017) Gravity anomalies of arbitrary 3D polyhedral bodies with horizontal and vertical mass contrasts. Surv Geophys 38(2):479–502. https://doi.org/10.1007/s10712-016-9395-x

Ren Z, Zhong Y, Chen C et al (2018) Gravity anomalies of arbitrary 3D polyhedral bodies with horizontal and vertical mass contrasts up to cubic order. Geophysics 83(1):G1–G13. https://doi.org/10.1190/geo2017-0219.1

Sansó F, Sampietro D (2022) Analysis of the gravity field: direct and inverse problems. Birkhäuser, Cham

Smith DA (2002) Computing components of the gravity field induced by distant topographic masses and condensed masses over the entire Earth using the 1-D FFT approach. J Geod 76(3):150–168. https://doi.org/10.1007/s00190-001-0227-4

Soler SR, Pesce A, Gimenez ME et al (2019) Gravitational field calculation in spherical coordinates using variable densities in depth. Geophys J Int 218(3):2150–2164. https://doi.org/10.1093/gji/ggz277

Tscherning CC (1976) Computation of second-order derivatives of the normal potential based on a representation by a Legendre series. Maunuscr Geod 1(1):71–92

Tsoulis D (2012) Analytical computation of the full gravity tensor of a homogeneous arbitrarily shaped polyhedral source using line integrals. Geophysics 77(2):F1–F11. https://doi.org/10.1190/geo2010-0334.1

Uieda L, Barbosa VCF, Braitenberg C (2016) Tesseroids: forward-modeling gravitational fields in spherical coordinates. Geophysics 81(5):F41–F48. https://doi.org/10.1190/geo2015-0204.1

Vogel CR (2002) Computational methods for inverse problems. SIAM, Philadelphia

Wild-Pfeiffer F (2008) A comparison of different mass elements for use in gravity gradiometry. J Geod 82(10):637–653. https://doi.org/10.1007/s00190-008-0219-8

Wu L, Chen L (2016) Fourier forward modeling of vector and tensor gravity fields due to prismatic bodies with variable density contrast. Geophysics 81(1):G13–G26. https://doi.org/10.1190/geo2014-0559.1

Wu L, Chen L, Wu B et al (2019) Improved Fourier modeling of gravity fields caused by polyhedral bodies: with applications to asteroid Bennu and comet 67P/Churyumov–Gerasimenko. J Geod 93(10):1963–1984. https://doi.org/10.1007/s00190-019-01294-2

Zhang Y, Chen C (2018) Forward calculation of gravity and its gradient using polyhedral representation of density interfaces: an application of spherical or ellipsoidal topographic gravity effect. J Geod 92(2):205–218. https://doi.org/10.1007/s00190-017-1057-3

Zhao G, Chen B, Uieda L et al (2019) Efficient 3-D large-scale forward modeling and inversion of gravitational fields in spherical coordinates with application to lunar mascons. J Geophys Res (Solid Earth) 124(4):4157–4173. https://doi.org/10.1029/2019JB017691

Zhao G, Liu J, Chen B et al (2021) 3D density structure of the lunar mascon basins revealed by a high efficient gravity inversion of the GRAIL data. J Geophys Res (Planets) 126(5):e06841. https://doi.org/10.1029/2021JE006841

Zhong Y, Ren Z, Chen C et al (2019) A new method for gravity modeling using tesseroids and 2D Gauss–Legendre quadrature rule. J Appl Geophys 164:53–64. https://doi.org/10.1016/j.jappgeo.2019.03.003