

RELATÓRIO 2015

MODELAGEM E INVERSÃO DE CAMPOS GRAVITACIONAIS EM COORDENADAS ESFÉRICAS

NOME DO ALUNO: Leonardo Uieda

NÍVEL: Doutorado

ENDEREÇO RESIDENCIAL: Rua Aguiar, 49 ap 104. 20261-120 Tijuca, Rio de Janeiro - RJ

TELEFONES FIXO e CELULAR: (21) 983636761

EMAIL NÃO INSTITUCIONAL: leouieda@gmail.com

DATA DE INÍCIO NA PÓS-GRADUAÇÃO: 11/2011

DATA DE INÍCIO DO PROJETO DE PESQUISA: 11/2011

NOME DO ORIENTADOR: Valéria C. F. Barbosa

PERÍODO PREVISTO DE BOLSA DE ESTUDOS: 11/2011 - 10/2015

PERÍODO A QUE SE REFERE O RELATÓRIO: 03/2014 - 02/2015

Este relatório foi entregue ao orientador no dia _____

ASSINATURA DO ORIENTADOR

O orientador emitiu parecer no dia _____

ASSINATURA DO ORIENTADOR

Este relatório e o parecer do orientador foi recebido na SECRETARIA DA
PÓS-GRADUAÇÃO no dia:

NOME E ASSINATURA DO FUNCIONÁRIO DA SECRETARIA

INFORMAÇÕES CURRICULARES, CONTENDO OS SEGUINTE ITENS:

(1) TOTAL DE CRÉDITOS CURSADOS EM DISCIPLINAS: 12

(2) LISTA DE TODAS AS DISCIPLINAS CURSADAS ATÉ O MOMENTO PELO ALUNO E OS CONCEITOS OBTIDOS

Tópicos de interpretação de dados gravimétricos e magnéticos - A

Minicurso: “Electromagnetic methods in applied geophysics” - (curso sem conceito)

Fenômenos críticos em geociências - A

Inversão em métodos potenciais - A

Sísmica aplicada (ênfase em exploração de petróleo e gás) - A

(3) SITUAÇÃO DO ALUNO QUANTO AOS CRÉDITOS OBTIDOS NOS SEMINÁRIOS ANUAIS

Aprovado nos anos de 2012 e 2014.

(4) SITUAÇÃO DO ALUNO QUANTO AO EXAME DE PROFICIÊNCIA

Aprovado.

(5) SITUAÇÃO DO ALUNO DE DOUTORADO QUANTO AO EXAME DE QUALIFICAÇÃO

Aprovado no ano 2013.

(6) LISTA DAS REUNIÕES CIENTÍFICAS EM QUE PARTICIPOU NO PERÍODO A QUE SE REFERE O RELATÓRIO, COM O TÍTULO E AUTORES DOS TRABALHOS APRESENTADOS

Evento: Python in Science Conference 2014

Autores: Uieda, L., V. C. Oliveira Jr and V. C. F. Barbosa

Título: Using Fatiando a Terra to solve inverse problems in geophysics

Material disponível em <https://github.com/leouieda/scipy2014>

Evento: EGU General Assembly 2014

Autores: Uieda, L. and V. C. F. Barbosa

Título: Gravity inversion in spherical coordinates using tessieroids

Material disponível em <https://github.com/leouieda/egu2014>

(7) LISTA DOS ARTIGOS PUBLICADOS, ACEITOS OU SUBMETIDOS

(Submetido) Carlos, D. U., Uieda, L., and V. C. F. Barbosa (2015), How two gravity-gradient inversion methods can be used to reveal different geologic features of ore deposit - a case study from the Quadrilátero Ferrífero (Brazil), Journal of Applied Geophysics.

(Em revisão) Oliveira Jr, V. C., D. P. Sales, V. C. F. Barbosa, and L. Uieda (2014), Estimation of the total magnetization direction of approximately spherical bodies, Nonlinear Processes in Geophysics.

(Publicado) Uieda, L., V. C. Oliveira Jr, and V. C. F. Barbosa (2014), Geophysical tutorial: Euler deconvolution of potential-field data, The Leading Edge, 33(4), 448-450, doi:10.1190/tle33040448.1.

(Publicado) Carlos, D. U., L. Uieda, and V. C. F. Barbosa (2014), Imaging iron ore from the Quadrilátero Ferrífero (Brazil) using geophysical inversion and drill hole data, Ore Geology Reviews, 61, 268-285, doi:10.1016/j.oregeorev.2014.02.011.

(8) OUTRAS ATIVIDADES RELEVANTES NO PERÍODO (PARTICIPAÇÃO EM TRABALHOS DE CAMPO, ESCOLAS ETC)

Ministrei o curso "Tópicos de inversão em geofísica" com duração de dois dias na III Semana de Geofísica da UNB. O material didático e informações sobre o curso estão disponíveis em <https://github.com/pinga-lab/inversao-unb-2014>

PROJETO ORIGINAL

Nesta parte do relatório o estudante deve incluir o projeto de pesquisa tal como apresentado à Comissão de Pós-Graduação em Geofísica do Observatório Nacional na época da inscrição.

OBSERVATÓRIO NACIONAL

CURSO DE PÓS-GRADUAÇÃO EM GEOFÍSICA

PROPOSTA DE TESE DE DOUTORADO

**Inversão 3D de campos gravitacionais para determinação
de corpos e interfaces geológicas em coordenadas
esféricas**

ALUNO: Leonardo Uieda

ORIENTADORA: Valéria Cristina Ferreira Barbosa

31 de Junho de 2011

Objetivos

Este projeto propõe o desenvolvimento de métodos de inversão 3D de dados de altura geoidal, gravimétricos e de gradiometria gravimétrica. Presumiremos que tanto os dados como os resultados da inversão estão definidos em coordenadas esféricas, tornando possível levar em consideração a curvatura da Terra. Desta forma, é viável realizar a interpretação de dados gravimétricos que cobrem áreas extensas, como dados de satélite e do BNDG (Banco Nacional de Dados Gravimétricos). Para tanto, a subsuperfície será discretizada em prismas esféricos, conhecidos como tesseroides, ao invés de prismas retos retangulares. Neste projeto, serão desenvolvidos dois métodos distintos para a inversão.

Um dos métodos será utilizado para a determinação de corpos geológicos 3D compactos. Este método buscará estimar a distribuição de contrastes de densidade em subsuperfície. Para tanto, a subsuperfície será discretizada em tesseroides justapostos horizontalmente e verticalmente. Presumiremos que cada tesseróide possui um contraste de densidade homogêneo porém desconhecido. Os parâmetros a serem estimados na inversão são os valores de contraste de densidade de cada tesseróide. O algoritmo de inversão utilizado será uma adaptação para tesseroides do método desenvolvido por Uieda e Barbosa (2011). Exemplos de estruturas geológicas que este método possivelmente será capaz de interpretar incluem diques e soleiras, corpos salinos, intrusões, câmaras magmáticas, entre outros. Além disto, o método desenvolvido poderá ser utilizado para avaliar se estruturas detectadas pela tomografia sísmica em coordenadas esféricas, como placas em subducção e plumas mantélicas, são capazes de explicar os dados gravimétricos.

O segundo método proposto buscará determinar o relevo de uma interface geológica separando dois meios. A interface será discretizada em tesseroides justapostos horizontalmente e com contraste de densidade conhecido. Os parâmetros a serem estimados na inversão são a espessura de cada tesseróide. A estabilidade das soluções será garantida utilizando regularizadores de Tikhonov (Tikhonov, 1963) e de Variação Total (Martins et al., 2011). Este método poderá ser utilizado na interpretação do embasamento de bacias sedimentares extensas, como as bacias intracratônicas brasileiras, e da descontinuidade de Mohorovičić.

Metodologia

Modelagem direta

O potencial gravitacional e suas primeiras e segundas derivadas causados por um tesseróide são dados em Wild-Pfeiffer (2008). Porém, não há solução analítica para estas relações e portanto métodos numéricos de integração devem ser utilizados. Asgharzadeh et al. (2007) propõe a utilização da Quadratura Gauss-Legendre para a integração numérica. Este método consiste em discretizar o integrando em pontos correspondentes a raízes de polinômios de Legendre e aproximar a integração por uma soma ponderada (Hildebrand, 1987).

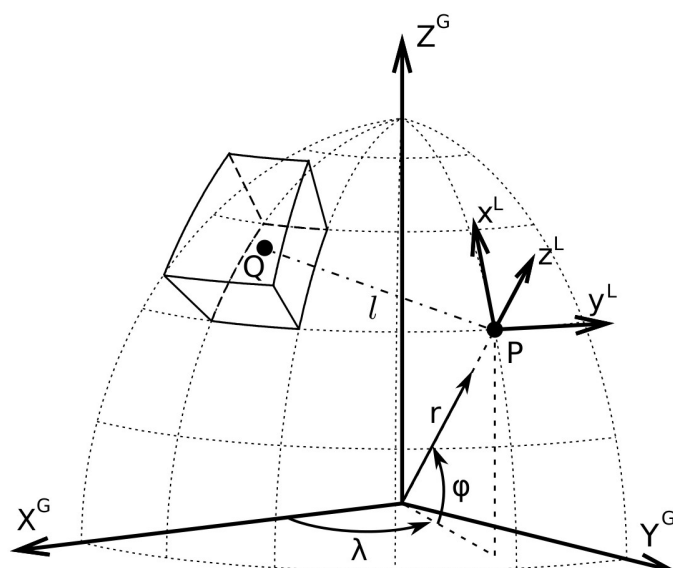


Figura 1: Representação esquemática de um tesseróide no sistema de coordenadas geocêntrico (G) e o ponto de observação P com o sistema local (L) correspondente.

Modelagem inversa de corpos geológicos

Se discretizarmos a subsuperfície em tesseróides justapostos horizontalmente e verticalmente, podemos aproximar o efeito gravitacional de corpos geológicos complexos

pela soma dos efeitos gravitacionais de todos os tesseroides. Considerando que cada tesseróide possui um contraste de densidade homogêneo, a relação entre o efeito gravitacional e os contrastes de densidade é linear e pode ser escrita em notação matricial como

$$\mathbf{d} = \mathbf{G} \mathbf{p} \quad , \quad (1)$$

em que \mathbf{d} é o vetor de dados preditos, \mathbf{G} é a matriz Jacobiana da relação funcional entre os dados preditos e os contrastes de densidade e \mathbf{p} é o vetor de parâmetros que contém os contrastes de densidade de cada tesseróide.

O algoritmo de inversão utilizado será uma adaptação para tesseróides do método desenvolvido por Uieda e Barbosa (2011). Este método consiste em crescer a solução em torno de tesseróides previamente definidos denominados “sementes”. O crescimento é feito incorporando novos tesseróides em torno das sementes de forma a obtermos corpos geológicos compactos (sem buracos em seu interior). Matematicamente, este problema consiste em minimizar a função objetivo

$$\Gamma(\mathbf{p}) = \phi(\mathbf{p}) + \mu \theta(\mathbf{p}) \quad , \quad (2)$$

em que a função $\phi(\mathbf{p})$ é uma medida do desajuste entre os dados observados e preditos. Esta função $\phi(\mathbf{p})$ pode ser a norma L1 ou L2 da diferença entre os dados observados e preditos (resíduos). A compacidade da solução é garantida através da utilização da função regularizadora

$$\theta(\mathbf{p}) = \sum_{i=1}^M \frac{p_i}{p_i + \epsilon} l_i^\beta \quad , \quad (3)$$

em que l_i é a distância entre o i-ésimo tesseróide do modelo interpretativo e a semente a qual ele será incorporado e ϵ é um número positivo pequeno.

Modelagem inversa do relevo de uma interface geológica

Seja o relevo de uma interface geológica definido como o desvio desta interface de um

nível de referência, é possível aproximar o efeito gravitacional deste relevo discretizando-o em M tesseroides justapostos horizontalmente. Nos referiremos a este conjunto de tesseroides como “modelo interpretativo”. Se fixarmos a base destes tesseroides no nível de referência e assumirmos um valor de contraste de densidade fixo, o efeito gravitacional d_i no i -ésimo ponto de observação produzido pelo conjunto de M tesseroides será função somente da espessura de cada tesseróide

$$d_i = \sum_{j=1}^M f_i(p_j) \quad i=1, \dots, N, \quad (4)$$

em que p_j é a espessura do j -ésimo tesseróide e $f_i(p_j)$ é a função não linear que relaciona o efeito gravitacional no i -ésimo ponto de medição com a espessura do j -ésimo tesseróide. Neste problema inverso os parâmetros a serem estimados são as espessuras dos M tesseroides, i.e., p_1, \dots, p_M .

Formulamos o problema inverso de estimar o relevo da interface geológica a partir de dados gravimétricos como a minimização da função objetivo

$$\Gamma(\mathbf{p}) = \|\mathbf{d}^o - \mathbf{d}\|^2 + \mu \sum_k \lambda_k \theta_k(\mathbf{p}), \quad (5)$$

em que \mathbf{p} é o vetor de parâmetros a ser estimado, \mathbf{d}^o é o vetor de dados observados, μ é um parâmetro de regularização, θ_k é uma função regularizadora e λ_k controla o peso da k -ésima função regularizadora. As funções regularizadoras que serão utilizadas incluem a de Tikhonov de ordem 1, que impõe suavidade a interface (Barbosa et al., 1997), e de variação total (Martins et al., 2011), que impõe descontinuidades a interface. Desta forma é possível interpretar interfaces tanto com topografia suave como descontínua.

Viabilização da Pesquisa

A viabilização desta pesquisa consistirá nos seguintes itens:

1. **Equipamento:** Neste projeto de pesquisa desenvolveremos e aplicaremos algoritmos

que podem ser escritos em qualquer linguagem de programação científica. Os algoritmos desenvolvidos poderão ser executados em Linux ou outros sistemas operacionais (Windows, por exemplo). Em termos de equipamento, este projeto será desenvolvido usando os computadores disponíveis na própria instituição.

2. **Dados:** Os dados gravimétricos serão dados de satélite e do BNDG (Banco Nacional de Dados Gravimétricos).
3. **Experiência no exterior:** Em 2013, solicitaremos uma bolsa de doutorado sanduíche via CNPq ou CAPES. O local de execução deste doutorado sanduíche será definido até o final 2012.

Cronograma de execução

A Tabela 1 mostra um cronograma de execução deste projeto de tese que será desenvolvido de Agosto de 2011 a Julho de 2015 através das seguintes etapas de trabalho:

Etapas 1 - Elaboração do modelo direto

Será implementado computacionalmente a modelagem direta do potencial gravitacional e suas derivadas primeiras segundas utilizando tesseroides. A acurácia desta implementação será testada com base em soluções analíticas para a casca esférica. Será também analisado o erro de aproximação da Quadratura Gauss-Legendre e buscaremos formas de minimizá-lo.

Etapas 2 - Desenvolvimento da inversão de corpos

Nesta etapa adaptaremos a metodologia de Uieda e Barbosa (2011) para utilizar tesseroides e as demais componentes do campo gravitacional. Possíveis otimizações serão investigadas.

Etapas 2.1 - Testes com dados sintéticos

Utilizando o método desenvolvido na etapa 2 realizaremos testes com dados

simulados, contaminados com ruído, que sejam produzidos por estruturas semelhantes a soleiras, corpos salinos, intrusões, câmaras magmáticas, entre outros. Nesta etapa analisaremos numericamente a estabilidade da solução estimada considerando a influência dos seguintes fatores: nível de ruído nos dados; posição das sementes; presença de fontes com efeito gravitacional interferente; utilização de diferentes componentes do campo gravitacional, sozinhas ou em conjunto.

Etapa 2.2 - Aplicação a dados reais

O método desenvolvido será aplicado um conjunto de dados reais objetivando caracterizar um ou mais corpos geológicos de grande extensão, como soleiras ou grandes intrusões. Este conjunto de dados ainda não está definido, porém muito provavelmente será composto de dados de satélite e/ou do BNDG (Banco Nacional de Dados Gravimétricos).

Etapa 2.3 - Elaboração do primeiro artigo científico

Nesta etapa será elaborado um artigo científico com os resultados obtidos na etapa 2 do projeto. Além disto, os resultados serão divulgados em congresso nacional ou internacional.

Etapa 3 - Desenvolvimento da inversão de interfaces

Nesta etapa desenvolveremos um método de inversão para estimar o relevo de interfaces geológicas em coordenadas esféricas baseado na solução da equação 5. Nesta etapa serão investigadas otimizações computacionais que viabilizem a interpretação de conjuntos de dados extensos.

Etapa 3.1 - Testes com dados sintéticos

Utilizando o método desenvolvido na etapa 3 realizaremos testes com dados simulados, contaminados com ruído, que sejam produzidos por grandes interfaces geológicas, suaves ou abruptas, como embasamentos de bacias sedimentares ou a

descontinuidade de Mohorovičić. Nesta etapa analisaremos numericamente a estabilidade da solução estimada considerando a influência dos seguintes fatores: nível de ruído nos dados; estimativa inicial; profundidade da interface; utilização de diferentes componentes do campo gravitacional, sozinhas ou em conjunto.

Etapa 3.2 - Aplicação a dados reais

O método desenvolvido será aplicado um conjunto de dados reais objetivando caracterizar uma interface geológica de grande extensão, como o embasamento de uma bacia sedimentar continental ou a descontinuidade de Mohorovičić sob o território brasileiro. Este conjunto de dados ainda não está definido, porém muito provavelmente será composto de dados de satélite e/ou da rede fundamental.

Etapa 3.3 - Elaboração do segundo artigo científico

Nesta etapa será elaborado um artigo científico com os resultados obtidos na etapa 3 do projeto. Além disto, os resultados serão divulgados em congresso nacional ou internacional.

Etapa 4 - Período no exterior com bolsa de doutorado sanduíche a ser solicitada

Etapa 5 - Elaboração da versão escrita da tese de doutorado

Etapa 6 - Defesa da tese de doutorado

Etapas		2011	2012		2013		2014		2015
		2º Sem	1º Sem	2º Sem	1º Sem	2º Sem	1º Sem	2º Sem	1º Sem
	Disciplinas	X	X	X					
E1	Elaboração do modelo direto	X							
E2	Desenvolvimento da inversão de corpos	X	X						
E2.1	Testes com dados sintéticos		X	X					
E2.2	Aplicação a dados reais			X					
E2.3	Elaboração do primeiro artigo científico			X	X				
E3	Desenvolvimento da inversão de interfaces				X				
E3.1	Testes com dados sintéticos				X	X			
E3.2	Aplicação a dados reais					X			
E3.3	Elaboração do segundo artigo científico					X	X		
E4	Período no Exterior comn Bolsa sanduiche					X	X		
E5	Elaboração da tese						X	X	X
E6	Defesa da tese								X

Tabela 1: Cronograma de execução.

Referências

- Asgharzadeh, M.F., Von Frese, R.R.B., Kim, H.R., Leftwich, T.E., Kim, J.W. , 2007, Spherical prism gravity effects by Gauss-Legendre quadrature integration. *Geophysical Journal International*, 169, 1-11.
- Barbosa VCF, Silva JBC, Medeiros W. E., 1997, [Gravity inversion of basement relief using approximate equality constraints on depths](#). *Geophysics*, 62(6), 1745-1757.
- Hildebrand. F.B., 1987, Introduction to numerical analysis. Courier Dover Publications, 2. ed.
- Martins, C.M., Lima, W.A., Barbosa, V.C.F., Silva, J.B.C., 2011, Total variation regularization for depth-to-basement estimate: Part 1 - Mathematical details and applications. *Geophysics*, 76 (1), I1-I12.
- Tikhonov, A. N., 1963, Regularization of ill-posed problems: Doklady Akad. Nauk, SSSR, 153 , 49-52.
- Uieda L., Barbosa, V.C.F., 2011, 3D gravity gradient inversion by planting density anomalies. In: 73th EAGE Conference & Exhibition incorporating SPE EUROPEC 2010, 2011, Vienna. 73th EAGE - 2011, 2011. v. 1. p. 1-5.
- Wild-Pfeiffer, F., 2008, A comparison of different mass elements for use in gravity gradiometry. *Journal of Geodesy*, 82 (10), 637-653.

RELATO DO DESENVOLVIMENTO DO PROJETO DE PESQUISA NO ÚLTIMO RELATÓRIO

No período entre Março de 2013 e Fevereiro de 2014, eu desenvolvi a infraestrutura computacional que seria necessária para a implementação da inversão de dados gravimétricos em coordenadas esféricas.

A primeira etapa foi o desenvolvimento de um conjunto de classes e funções em linguagem Python para automatizar a criação de métodos de inversão. Esse código foi incluído na biblioteca Fatiando a Terra (<http://fatiando.org/>), onde é utilizado para implementar os diversos métodos de inversão já incluídos na biblioteca. Esse novo módulo de inversão automatiza as tarefas de (1) construção da função objetivo, (2) implementação das diversas formas de regularização, (3) minimização da função objetivo por diferentes métodos, (4) cálculo do vetor gradiente e matriz Hessiana da função objetivo e (5) determinação do parâmetro de regularização através da análise da curva L.

A segunda etapa foi a implementação da modelagem direta em coordenadas esféricas utilizando tesseroides (prismas esféricos). Fiz uma implementação da modelagem em linguagem Python para comparar com a implementação anterior em linguagem C do programa Tesseroids (<http://tesseroids.leouieda.com>). Obtive resultados preliminares da análise do erro cometido na integração numérica do campo gravitacional de um tesseróide. Existem poucos modelos em coordenadas esféricas com soluções analíticas que possam ser utilizados na quantificação do erro. Utilizei o modelo de meia casca esférica, para a qual existe uma solução analítica para o ponto de observação localizado no polo. Outro modelo utilizado foi o de um prisma retangular reto. Este não é um modelo ideal pois a aproximação de um tesseróide por um prisma retangular piora quanto maior o tesseróide ou quanto mais longe da região equatorial.

DESCRIÇÃO DETALHADA DO TRABALHO DE PESQUISA DESENVOLVIDO NO PERÍODO DO RELATÓRIO

Metodologia aplicada ou desenvolvida e Resultados parciais já obtidos

Abaixo o manuscrito do artigo sobre a modelagem direta utilizando tesseroides que será submetido para a revista *Geophysics*. Este trabalho descreve a metodologia para realizar a modelagem e sua implementação computacional. A metodologia sofreu diversas modificações durante a escrita do artigo no ano de 2014.

Também realizei uma implementação inicial da inversão de dados gravimétricos para determinar o relevo de uma interface. A parametrização da interface é feita com tesseroides, possibilitando a inversão em coordenadas esféricas de grandes áreas. A inversão é feita com uma adaptação do método de [Silva et al. \(2014\)](#). A implementação foi feita utilizando o módulo de inversão do software *Fatiando a Terra* (descrito no relatório de 2014). Essa infraestrutura computacional permite a implementação de uma inversão relativamente complexa em poucas (≈ 30) linhas de código. A implementação desta inversão ainda necessita ser testada extensivamente para verificar sua eficácia.

Dificuldades encontradas e como elas estão sendo superadas

A metodologia para modelagem direta sofreu diversas alterações. O algoritmo de discretização adaptativa dos tesseroides foi convertido de um algoritmo recursivo para um algoritmo utilizando uma estrutura de dados denominada “pilha” (mais detalhes em “Implementation” abaixo). Essa mudança aumenta a velocidade do algoritmo pois evita o tempo gasto com chamadas a funções.

A análise do erro da integração numérica passou a utilizar como referência uma casca esférica no lugar da meia casca esférica que foi utilizada anteriormente. Fizemos essa modificação pois a meia casca esférica possui solução analítica em um único ponto, no polo. A figura 2 mostra que o maior erro na integração numérica está em uma área acima do tesseróide. Logo, o cálculo em um único ponto pode não registrar o maior erro da integração, subestimando o tamanho do erro (mais detalhes em “Evaluation of the accuracy” abaixo).

Manuscrito do artigo sobre modelagem direta com tesseroides

Tesseroids: forward modeling gravitational fields in spherical coordinates
by Leonardo Uieda, Valéria C. F. Barbosa, and Carla Braitenberg

Abstract

We present the open-source software *Tesseroids*, a set of command-line programs to perform the forward modeling of gravitational fields in spherical coordinates. The fields are calculated numerically using the Gauss-Legendre Quadrature (GLQ). We have improved upon an adaptive discretization algorithm to guarantee the accuracy of the GLQ integration. Our implementation of adaptive discretization uses a “stack” based algorithm instead of recursion to achieve higher performance and more control over execution errors and corner cases. The algorithm is controlled by a scalar called the distance-size ratio (D) that determines the accuracy of the integration as well as the computation time. We

determined optimal values of D for the gravitational potential, gravitational acceleration, and gravity gradient tensor by comparing the computed tesserooids effects with those of a homogeneous spherical shell. Contrary to previous assumptions, our results show that the potential and its first and second derivatives require different values of D to achieve a target accuracy of 0.1%. These values were incorporated as defaults in the software.

1 Introduction

Satellite missions dedicated to measuring the Earths gravity field (like CHAMP, GRACE, and GOCE) have provided geophysicists with almost uniform and global data coverage. These new data have enabled interpretations on regional and global scales (e.g. [Reguzzoni et al., 2013](#); [Braitenberg, 2015](#)). Modeling at such scales requires taking into account the curvature of the Earth and calculating gravity gradients as well as the traditional gravitational acceleration. A common approach to achieve this is to discretize the Earth into tesserooids (Figure 1) instead of rectangular prisms. However, the integral formula for the gravitational effects of a tesserooid must be solved numerically. Approaches to this numerical integration include Taylor series expansion ([Heck and Seitz, 2007](#); [Grombein et al., 2013](#)) and the Gauss-Legendre Quadrature ([Asgharzadeh et al., 2007](#)). Taylor series expansion produces accurate results at low latitudes but presents a decrease in accuracy towards the polar regions. This is attributed to tesserooids degenerating into an approximately triangular shape at the poles. The Gauss-Legendre Quadrature (GLQ) integration consists of approximating the volume integral by a weighted sum of the effect of point masses. An advantage of the GLQ approach is that it can be controlled by the number of point masses used. The larger the number of point masses, the better the accuracy of GLQ integration. A disadvantage is the increased computation time as the number of point masses increases. Thus, there is a trade-off between accuracy and computation time. This is a common theme in numerical methods. [Wild-Pfeiffer \(2008\)](#) investigated the use of different mass elements, including tesserooids, to compute the gravitational effects of topographic masses. The author concludes that using tesserooids with GLQ integration gives the best results for near-zone computations. However, the question of how to determine the optimal parameters for GLQ integration remained open.

Previous work by [Ku \(1977\)](#) investigated the use of the GLQ in gravity forward modeling. [Ku \(1977\)](#) numerically integrated the vertical component of the gravitational acceleration of right rectangular prisms. The author suggested that the accuracy of the GLQ integration depends on the ratio between distance to the computation point and the distance between adjacent point masses. Based on this, [Ku \(1977\)](#) proposed an empirical criteria that the distance between adjacent point masses should be less than the distance to the computation point. [Asgharzadeh et al. \(2007\)](#) used this criteria for the GLQ integration of the gravity gradient tensor of tesserooids. To our knowledge, an analysis of how well this ad hoc criteria of [Ku \(1977\)](#) works for gravity gradient components or for tesserooids has never been done before. There has also been no attempt to quantify the error committed in the GLQ integration when applying the criteria of [Ku \(1977\)](#).

[Li et al. \(2011\)](#) devised an algorithm to automatically enforce the criteria of [Ku \(1977\)](#). Their algorithm divides the tesserooid into smaller ones instead of increasing the number of point masses per tesserooid. This division effectively increases the total number of point masses because the number of points masses per tesserooid is fixed. A tesserooid is divided if the minimum distance to the computation point is smaller than the largest dimension of the tesserooid. This division is repeated recursively until all tesserooids obey the criterion. Then, GLQ integration is performed for each of the smaller tesserooids using the specified number of point masses. The advantage of this adaptive discretization over

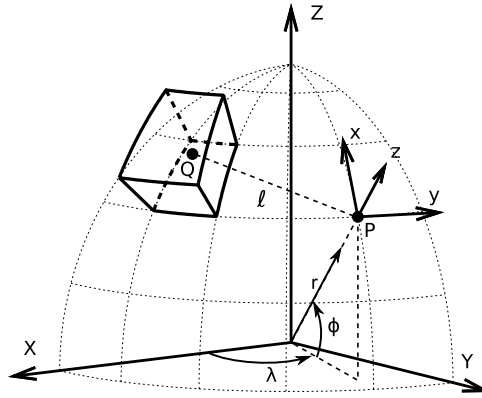


Figura 1: View of a tesseroïd, the integration point Q inside the tesseroïd, a geocentric coordinate system (X, Y, Z) , the computation P and its local coordinate system (x, y, z) . r , ϕ , λ are the radius, latitude, and longitude, respectively, of point P , and ℓ is the Cartesian distance between P and Q .

increasing the number of points masses is that the total distribution of point masses will be greater only close to the computation point. This makes the adaptive discretization more computationally efficient.

Grombein et al. (2013) developed optimized formula for the gravitational fields of tesseroïds using Cartesian integral kernels. These formulas are faster to compute and do not have singularities at the poles like their spherical counterparts. The Cartesian formula are numerically integrated using a Taylor series expansion as per Heck and Seitz (2007). Grombein et al. (2013) use a near-zone separation to mitigate the increased error at high latitudes. In the so called “near-zone” of the computation point they use a finer discretization composed by smaller tesseroïds. This is accomplished by dividing the tesseroïds along their horizontal dimensions. However, the determination of an optimal size of the near-zone remains an open question (Grombein et al., 2013).

We have implemented a modified version of the adaptive discretion of Li et al. (2011) into the open-source software package *Tesseroïds*. The software uses the Cartesian formula of Grombein et al. (2013) for improved performance and robustness. Previous versions of the software have been used by, e.g., Álvarez et al. (2012); Bouman et al. (2013a,b); Mariani et al. (2013); Braitenberg et al. (2014, 2011); Fullea et al. (2014).

This article describes the software design and the implementation of our modified adaptive discretization algorithm. We also present a numerical investigation of the error committed in the computations. These results allow us calibrate the adaptive discretization algorithm separately for the gravitational potential, gravitational acceleration, as well as the gravity gradient tensor components.

2 Theory

A tesseroïd is a mass element defined in geocentric spherical coordinates (Figure 1). It is bounded by two meridians, two parallels, and two concentric circles. The gravitational fields of a tesseroïd at a point $P = (r, \phi, \lambda)$ are determined with respect to the local North-oriented coordinate system at P (x, y, z in Figure 1). Grombein et al. (2013) formulated Cartesian kernels for the volume integrals that define the tesseroïd gravitational potential,

gravitational acceleration, and Marussi tensor, respectively,

$$V(r, \phi, \lambda) = G\rho \int_{\lambda_1}^{\lambda_2} \int_{\phi_1}^{\phi_2} \int_{r_1}^{r_2} \frac{1}{\ell} \kappa \, dr' d\phi' d\lambda', \quad (1)$$

$$g_\alpha(r, \phi, \lambda) = G\rho \int_{\lambda_1}^{\lambda_2} \int_{\phi_1}^{\phi_2} \int_{r_1}^{r_2} \frac{\Delta_\alpha}{\ell^3} \kappa \, dr' d\phi' d\lambda', \quad (2)$$

and

$$g_{\alpha\beta}(r, \phi, \lambda) = G\rho \int_{\lambda_1}^{\lambda_2} \int_{\phi_1}^{\phi_2} \int_{r_1}^{r_2} I_{\alpha\beta} \kappa \, dr' d\phi' d\lambda'. \quad (3)$$

$$I_{\alpha\beta} = \left(\frac{3\Delta_\alpha\Delta_\beta}{\ell^5} - \frac{\delta_{\alpha\beta}}{\ell^3} \right) \quad (4)$$

where $\alpha, \beta \in \{x, y, z\}$, ρ is the density, $G = 6.673 \times 10^{-11} \, m^3 kg^{-1} s^{-1}$ is the gravitational constant, and

$$\Delta_x = r'(\cos \phi \sin \phi' - \sin \phi \cos \phi' \cos(\lambda' - \lambda)), \quad (5)$$

$$\Delta_y = r' \cos \phi' \sin(\lambda' - \lambda), \quad (6)$$

$$\Delta_z = r' \cos \psi - r, \quad (7)$$

$$\kappa = r'^2 \cos \phi', \quad (8)$$

$$\ell = \sqrt{r'^2 + r^2 - 2r'r \cos \psi}, \quad (9)$$

$$\cos \psi = \sin \phi \sin \phi' + \cos \phi \cos \phi' \cos(\lambda' - \lambda). \quad (10)$$

We will follow [Asgharzadeh et al. \(2007\)](#) and perform the numerical integration using the Gauss-Legendre Quadrature (GLQ). The GLQ consists of approximating the integral by a weighted sum of the integration kernel ([Hildebrand, 1987](#)),

$$\int_a^b f(x) dx \approx \frac{b-a}{2} \sum_{i=1}^N W_i f(x_i), \quad (11)$$

in which N is the order of the quadrature, i.e. the number of points used in the GLQ. The points x_i are called the quadrature nodes. They are the roots of the N^{th} order Legendre polynomial $P_N(x)$. For a second order polynomial ($P_2(x)$), the roots are $x = \pm 0.577350269$. Roots for larger order polynomials can be determined by a root finder algorithm. Roots of Legendre polynomials will be within the range $[-1, 1]$. Before being used for GLQ integration, the roots must be scaled to the integration limits $[a, b]$ using

$$x_i^{scaled} = \frac{b-a}{2} x_i + \frac{b+a}{2}. \quad (12)$$

The weights of the GLQ are given by (Hildebrand, 1987),

$$W_i = \frac{2}{(1 - x_i^2)(P'_N(x_i))^2}. \quad (13)$$

The values of $P_N(x)$ and its first derivative $P'_N(x)$ can be calculated with recursive relations.

The Gauss-Legendre Quadrature for three-dimensional volume integrals, like equations 1-3, becomes (Asgharzadeh et al., 2007)

$$\iiint_{\Omega} f(r', \lambda', \phi') d\Omega \approx A \sum_{i=1}^{N^r} \sum_{j=1}^{N^\phi} \sum_{k=1}^{N^\lambda} W_i^r W_j^\phi W_k^\lambda f(r_i, \phi_j, \lambda_k), \quad (14)$$

where

$$A = \frac{(\lambda_2 - \lambda_1)(\phi_2 - \phi_1)(r_2 - r_1)}{8}. \quad (15)$$

Comparing equation 14 with equations 1-3, we see that $f(r_i, \phi_j, \lambda_k)$ is the effect of a point mass located on the quadrature nodes. Thus, it can be said that the GLQ integration approximates the volume integrals by a weighted sum of point mass effects.

The accuracy of the integration depends on the number of point masses used in the summation. Ku (1977) showed that it also depends on the ratio between the distance to the computation point and the distance between adjacent nodes. Figure 2 illustrates this effect on the g_{xy} gravity gradient component. The g_{xy} component was produced by a $7^\circ \times 7^\circ \times 20$ km tesseroid with 2.67 g.cm^{-3} density and top at $z = 0$ km. The maps were calculated on a regular grid with 100×100 points. Figure 2a shows the g_{xy} component calculated at 400 km height using GLQ with order two ($2 \times 2 \times 2 = 8$ point masses). Figure 2b shows g_{xy} computed with order two GLQ as well but at 150 km height. Notice that the computed effect is concentrated around each point mass of the GLQ (black dots) and does not resemble the effect of a tesseroid. Ku (1977) determined an ad hoc criterion that the distance between point masses (quadrature nodes) should be smaller than the minimum distance to the computation point. Thus, if a computation point is too close to the tesseroid one would have to decrease the distance between the point masses in order to obtain an accurate result. One way to accomplish this would be increase the order of the quadrature N in all three directions. Figure 2c shows the g_{xy} component calculated at 150km height but with a GLQ order of 30 ($30 \times 30 \times 30 = 27,000$ point masses). The computed g_{xy} component more closely resembles the expect results for a single tesseroid (Asgharzadeh et al., 2007).

2.1 Adaptive discretization

Li et al. (2011) proposed an alternative method for decreasing the distance between point masses on the quadrature nodes aiming at achieving an accurate integration. Instead of increasing the GLQ order, they keep it fixed to a given number and divide the tesseroid into smaller volumes. The sum of the effects of the smaller tesseroids is equal to the gravitational effect of the larger tesseroid. This division effectively decreases the distance between nodes because of the smaller size of the tesseroids. The criterion for dividing a tesseroid is that the distance to the computation point should be smaller than a constant times the size of the tesseroid. This is analogous to the criterion proposed by Ku (1977) because the size of the tesseroid serves as a proxy for the distance between point masses. This procedure is repeated recursively until all tesseroids are within the acceptable ratio of distance and size or a minimum size is achieved.

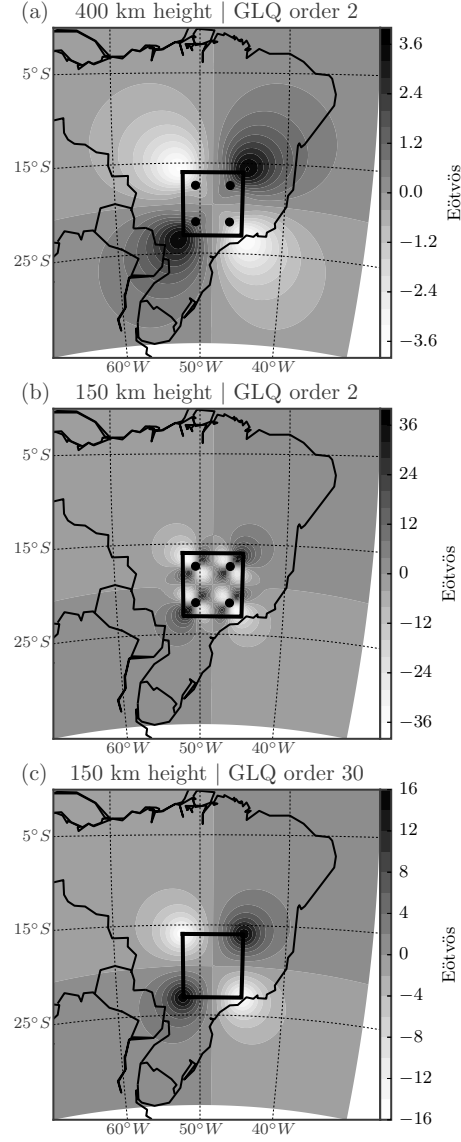


Figure 2: Example of the effect of varying the computation height and the number of point masses in the Gauss-Legendre Quadrature. Black circles represent the horizontal location of the point masses. a) g_{xy} calculated at 400 *km* height using GLQ order 2 ($2 \times 2 \times 2 = 8$ point masses). b) At 150 *km* height and GLQ order 2, the result resembles that of four point masses instead of a single tesseroïd. This effect was shown by [Ku \(1977\)](#). c) At 150 *km* but with a higher GLQ order of 30. In (c) the horizontal locations of the point masses were not shown. Notice that the results shown in (c) are similar to that expected for a single mass source.

The advantage of this adaptive discretization is that the number of point masses is only increased in parts of the tesseroid that are closer to the computation point. Notice that the alternative approach of simply increasing the order of the GLQ would increase the number of point masses evenly throughout the whole tesseroid.

3 Implementation

We have implemented the calculation of the tesseroid gravitational fields with adaptive discretization in version 1.2 of the open-source package *Tesseroids*. It is freely available online¹² under the BSD 3-clause open-source license. An archived version of the source code is also available as part of this article.

Tesseroids consists of command-line programs written in the C programming language. The package includes programs to calculate the gravitational fields of tesseroids and rectangular prisms (in both Cartesian and spherical coordinates). All programs receive input through command-line arguments and the standard input channel (“STDIN”) and output the results through the standard output channel (“STDOUT”). For example, the command to generate a regular grid with $NLON \times NLAT$ points, calculate on it g_z and g_{zz} caused by the tesseroids in a file “MODELFILE”, and save the results to a file called “OUTPUT” is:

```
tessgrd -rW/E/S/N -bNLON/NLAT -zHEIGHT | \
    tessgz MODELFILE | \
    tessgzz MODELFILE > OUTPUT
```

The *src* folder of the source code archive contains the C files that build the command-line programs (e.g., *tessgz.c*). The *src/lib* folder contains the source files that implement the numerical computations. We will not describe here the implementation of the input/output parsing and other miscellanea. Instead, we will focus on the details of the Gauss-Legendre Quadrature integration of equations 1-3 and the adaptive discretization of tesseroids.

3.1 Numerical integration

The source file *src/lib/glq.c* contains the code necessary to perform a Gauss-Legendre Quadrature integration. The first step in the GLQ is to compute the locations of the discretization points (i.e., the point masses). These points are roots of Legendre polynomials. Precomputed values are available low order polynomials, typically up to order five. For flexibility and to compute higher order roots, we use the multiple root-finder algorithm of Barrera-Figueroa et al. (2006). The additional computational load is minimal because the root-finder algorithm must be run only once per program execution. The root-finder is implemented in functions *glq_nodes* and *glq_next_root*. The computed roots will be in the range $[-1, 1]$ and must be scaled to the integration limits (the physical boundaries of the tesseroid) using function *glq_set_limits* (see equation 12).

The GLQ weights (equation 13) are computed by function *glq_weights*. Both the computed roots and weights are stored in a data structure (a C *struct*) called *GLQ*. Function *glq_new* handles memory allocation, calculates the roots and weights, and returns the complete *GLQ* structure.

¹ <http://tesseroids.leouieda.com>

² <http://dx.doi.org/10.5281/zenodo.16033>

The numerical integration of the tesseroïd gravitational fields is performed by the functions in module *src/lib/grav_tess.c*. Functions *tess_pot*, *tess_gx*, *tess_gy*, and so on, compute the gravitational fields of a single tesseroïd on a single computation point. These functions require three *GLQ* structures, each containing the roots and weights for *GLQ* integration in the three dimensions. The roots must be scaled to the integration limits $[\lambda_1, \lambda_2], [\phi_1, \phi_2], [r_1, r_2]$ (see equations 1-3). The integration consists of three loops that sum the weighted kernel functions evaluated at each *GLQ* point mass (the scaled roots).

The biggest bottlenecks for the numerical integration are the number of point masses used and the evaluation of the trigonometric functions in equations 1-3 inside the inner loops. Better performance is achieved by pre-computing the sine and cosine of latitudes and moving some trigonometric function evaluations to the outer loops.

3.2 Implementation of adaptive discretization

Our implementation of the adaptive discretization algorithm differs in a few ways from one proposed by Li et al. (2011). In Li et al. (2011), a tesseroïd will be divided when the smallest distance between it and the computation point is smaller than a constant times the largest dimension of the tesseroïd. Instead of the smallest distance, we use the easier to calculate distance between the computation point (r, λ, ϕ) and the geometric center of the tesseroïd (r_t, λ_t, ϕ_t)

$$d = [r^2 + r_t^2 - 2rr_t \cos \psi_t]^{\frac{1}{2}}. \quad (16)$$

$$\cos \psi_t = \sin \phi \sin \phi_t + \cos \phi \cos \phi_t \cos(\lambda - \lambda_t) \quad (17)$$

Our definition of the dimensions of the tesseroïd (the “side lengths” of Li et al. (2011)) along longitude, latitude, and radius, respectively, are (Figure 3a)

$$L_\lambda = r_2 \arccos(\sin^2 \phi_t + \cos^2 \phi_t \cos(\lambda_2 - \lambda_1)), \quad (18)$$

$$L_\phi = r_2 \arccos(\sin \phi_2 \sin \phi_1 + \cos \phi_2 \cos \phi_1), \quad (19)$$

$$L_r = r_2 - r_1. \quad (20)$$

L_λ and L_ϕ are arc-distances measured along the top surface of the tesseroïd (Figure 3a). Specifically, L_λ is measured long the middle latitude of the tesseroïd (ϕ_t).

To determine if a tesseroïd must be divided, we check if

$$\frac{d}{L_i} \geq D, \quad (21)$$

for each $i \in (\lambda, \phi, r)$. D is a positive scalar hereafter referred to as the “distance-size ratio”. If the inequality holds for all three dimensions, the tesseroïd is not divided. Thus, the distance-size ratio determines how close the computation point can be before we must divide the tesseroïd. The value of D is indirectly responsible for the accuracy of the solution and the computation time. We will explore the relationship with the accuracy in the following section.

Figure 3 shows examples of the resulting tesseroïd models after adaptive discretization. Figure 3a shows the initial tesseroïd and computation point P. Figures 3b-d are the result of adaptive discretization using different values of the distance-size ratio D , respectively,

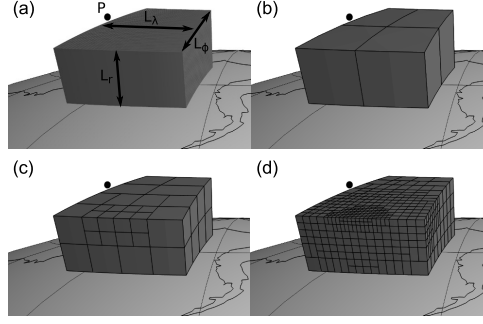


Figure 3: Adaptive discretization of the tesseract shown in (a) for a computation point P using the distance-size ratio D equal to (b) 1, (c) 2, and (d) 6. L_r , L_ϕ , and L_λ are the dimensions of the tesseract. Note that increasing D results in a fine division of the tesseract close the computation point and a coarser division further away.

$D = 1$, $D = 2$, and $D = 6$. The number of tesseroids in the resulting discretization is, respectively, 4, 38, and 936.

Instead of using recursive function calls, as originally proposed by [Li et al. \(2011\)](#), we use a stack-based implementation of the algorithm. Stacks are array-like data structures with a particular way of inserting and removing elements from it. In a stack, one can only insert elements to the top of the stack (the last empty position). Likewise, one can only remove the last element of the stack (commonly referred to as “popping” the stack). Stacks are also known as “Last-in-first-out”, or LIFO, data structures.

We will describe the stack-based algorithm for computing the effect of a single tesseract on a single computation point. The algorithm is implemented in function `calc_tess_model_adapt` of the file `src/lib/grav_tess.c`. The stack of tesseroids is represented by the `stack` variable, an array of `TESSEROID` structures. We must define a maximum size for the stack to allocate memory for it. Defining a maximum size allows us to avoid an infinite loop in case the computation point is on (or sufficiently close to) the surface of the tesseract. We use the integer `stktop` to keep track of the index of the last element in the stack (the top of the stack).

The algorithm starts by creating an empty stack of tesseroids. Then, the stack is initialized with the input tesseract. The initialization is done by copying the tesseract into the stack and setting `stktop` to zero (the first element). Once the stack is initialized, the steps of the algorithm are:

1. “Pop” the stack (i.e., take the last tesseract from it). This will cause `stktop` to be reduced by one. This tesseract is the one that will be evaluated in the following steps.
2. Compute the distance d (equation 16) between the geometric center of the tesseract and the computation point.
3. Compute the dimensions of the tesseract L_λ , L_ϕ , and L_r using equations 18-20.
4. Check the condition in equation 21 for each dimension of the tesseract.
5. If all dimensions hold the inequality 21, the tesseract is not divided and its gravitational effect is computed using the Gauss-Legendre Quadrature (equations 1-3 and 14). We use a GLQ order of two for all three dimensions ($2 \times 2 \times 2 = 8$ point masses) by default. This value can be changed using a command-line argument of the modeling programs.

6. If any of the dimensions fail the condition:
 - (a) Divide the tesseractoid in half along each dimension that failed the condition.
 - (b) Check if there is room in the stack for the new tesseractoids (i.e., the number of new elements plus *stktop* is smaller than the maximum stack size). If there isn't, warn the user of a "stack overflow" and compute the effect of the larger tesseractoid (step 5). If there is room in the stack, place the smaller tesseractoids into the stack.
7. Repeat the above steps until the stack is empty (*stktop* is equal to -1).

This stack-based implementation has some advantages over the original recursive implementation, namely: (1) It will be faster because it bypasses the overhead of function calls. (2) It gives the developer more control over the recursion step. In recursive implementations, the developer has no control over the maximum number of consecutive recursive calls (i.e., the "recursion depth"). This limit may vary with programming language, compiler, and operating system. Overflowing the maximum recursion depth may result in program crashes, typically with cryptic or inexistent error messages. In the stack-based implementation, the developer has complete control. Overflowing of the stack can be handled gracefully with an error message or even performing a suitable approximation of the result.

3.3 Code for figures and error analysis

The error analysis and all figures in this article were produced in IPython notebooks (Pérez and Granger, 2007). The notebook files combine source code in various programming languages, program execution, text, equations, and the figures generated by the code into a single document. We used the following Python language libraries to perform the error analysis and generate figures: *pandas* by McKinney (2010), *matplotlib* by Hunter (2007) for 2D figures and maps, and *Mayavi* by Ramachandran and Varoquaux (2011) for 3D figures.

The IPython notebooks and the data generated for the error analysis, as well as instructions for installing the software and running the programs, are also included in the source code archive that accompanies this article. Alternatively, all accompanying material is available in online repositories³⁴.

4 Evaluation of the accuracy

The key controlling point of the adaptive discretization algorithm is the distance-size ratio D (equation 21). The specific value chosen for D determines how many divisions will be made (Figure 3). Thus, D indirectly controls both the accuracy of the integration and the computation time. In this section, we investigate the relationship between the distance-size ratio and the integration error. We perform the analysis for the gravitational potential, acceleration, and gradient tensor components to evaluate if the same value of D yields compatible error levels for different fields.

The reference against which we compare the computed tesseractoid fields is a homogeneous spherical shell. The shell has analytical solutions along the polar axis (Grombein et al., 2013) and can be perfectly discretized into tesseractoids. We chose a spherical shell

³ <https://github.com/pinga-lab/paper-tesseractoids>

⁴ <http://dx.doi.org/10.5281/zenodo.16032>

with a thickness of 1 km, density of 2670 kg.m^{-3} , bottom at height 0 km above the reference sphere, and top at 1 km height. We produced tesseroïd models of the shell by discretizing it along the horizontal dimensions into a regular mesh.

Figure 2 shows that the largest errors are spread over on top of the tesseroïd. Thus, calculating the tesseroïd fields at a single point might not capture the point of largest error. Instead, we calculate the effect of the tesseroïd model on a regular grid of 10×10 points at different geographic locations (see Table 1). Fortunately, the symmetry of the shell allows us to consider the computation point at any geocentric coordinate. Therefore, the effect of the shell will be same along the entire grid. We compute the differences between the effects of the shell and the tesseroïd model on the grid. However, we will consider only the largest error in our analysis.

We placed the grid on top of a particular tesseroïd to increase the chances of capturing the true largest integration error. We calculate the errors for values of the distance-size ratio D varying from 0 (i.e., no divisions) to 10 in 0.5 intervals. Furthermore, we repeated the error analysis in four different numerical experiments, each with computation grids at different locations and different tesseroïd model sizes. Table 1 describes the different numerical experiments and the corresponding parameters of the computation grid and tesseroïd model.

Figure 4 shows the maximum difference between the shell and tesseroïd fields as a function of D for the four experiments. The differences are given as a percentage of the shell value. We established a maximum tolerated error of 0.1%, represented by the horizontal solid lines in Figure 4. Only results for the gravitational potential, g_z , and g_{zz} are shown. The results for the other diagonal components of the gravity gradient tensor are similar to g_{zz} . Figures for these components can be found in the supplementary material (see section "Reproducing the analysis and results").

For the potential V , a distance-size ratio $D = 1$ guarantees that the curves for all experiments are below the 0.1% error threshold. For g_z , the same is achieved with $D = 1.5$. Conversely, g_{zz} requires a value of $D = 8$ to achieve an error level of 0.1%. For a computation height of 260 km, the error curve for g_{zz} intercepts the error threshold line at $D = 2.5$. This behavior suggests that the error curves for g_{zz} might depend on the computation height. To test this hypothesis, we computed the error curves for g_{zz} at heights 2, 10, 50, 150, and 260 km. Figure 5 shows the results for g_{zz} at varying computation heights. Notice that the distance-size ratio required to achieve 0.1% accuracy decreases as the computation height increases. For example, computation at 260 km height requires $D = 2.5$ whereas at 10 km height a value of $D = 5.5$ is required to achieve the same accuracy. One can take advantage of this behavior to reduce the distance-size ratio for computations of the gravity gradient tensor at high altitudes, saving computation time.

We have implemented the values of the distance-size ratio producing 0.1% accuracy determined above as defaults for the software *Tesseroïds*. We chose the conservative value of $D = 8$ for the gravity gradient components as a fail-safe alternative. Users can control the value of D used in the computations through command-line arguments to achieve greater performance at the cost of accuracy.

5 Conclusions

We have presented the open-source software *Tesseroïds*. It consists of command-line programs, written in the C programming language, to perform the forward modeling of gravitational fields in spherical coordinates. The fields are calculated from a mass model composed of spherical prisms, the so-called tesseroïds. The volume integrals of

	Grid location	Grid height	Tesseractoid size
Experiment 1 (pole)	89N–90N/0E–1E	2 km	$1^\circ \times 1^\circ$
Experiment 2 (equator)	0N–1N/0E–1E	2 km	$1^\circ \times 1^\circ$
Experiment 3 (260 km)	89N–90N/0E–1E	260 km	$1^\circ \times 1^\circ$
Experiment 4 (30° size)	60N–90N/0E–30E	2 km	$30^\circ \times 30^\circ$

Tabela 1: Parameters of the numerical experiments to quantify the accuracy of the numerical integration. All grids had 10×10 regularly spaced computation points at a constant height. Tesseroids used to discretize the spherical shell had 1 km thickness and the horizontal dimensions shown in the table.

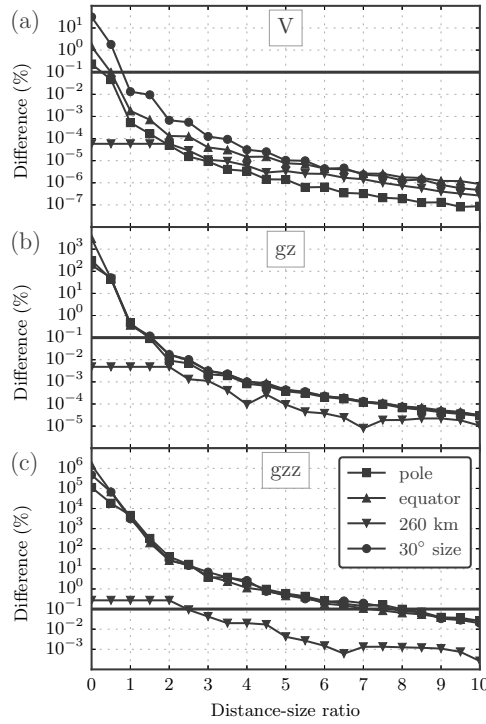


Figure 4: The maximum difference between the computed tesseroid and shell effects as a function of the distance-size ratio D for (a) the gravitational potential, (b) g_z , and (c) g_{zz} . The difference is given as a percentage of the shell effect. Curves correspond to the different tesseroid models and computation grids shown in Table 1. The horizontal solid black line marks the established error threshold of 0.1%. A value of $D = 0$ means that no divisions are made.

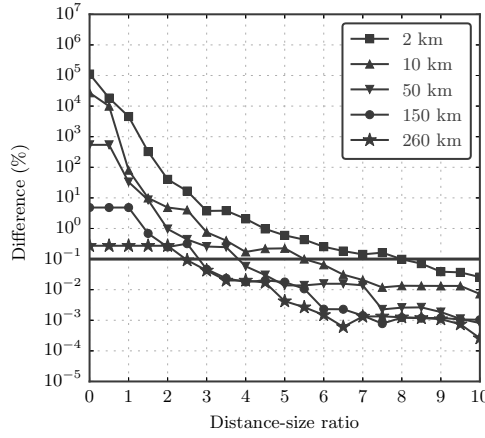


Figure 5: Difference between the computed g_{zz} for the spherical shell and the tesseroïd model at different heights. Curves show the maximum difference as a percentage of the shell value. The horizontal solid black line marks the established error threshold of 0.1%. A value of $D = 0$ means that no divisions are made.

the gravitational fields of a tesseroïd are solved numerically using the Gauss-Legendre Quadrature (GLQ). The GLQ approximates the volume integrals by weighted sums of point mass effects. The error of the GLQ integration increases as the computation point gets closer to the tesseroïd. To counter this effect, the accuracy of the GLQ integration can be increased by using more point masses or by dividing each tesseroïd into smaller ones.

We have implemented and improved upon an adaptive discretization algorithm to achieve an optimal division of tesseroïds. Tesseroïds are divided into more parts closer to the computation point, where more point masses are needed. Our implementation of the adaptive discretization uses a “stack” data structure in place of the originally proposed recursive implementation. The stack-based algorithm is computationally faster and allows more control over errors when too many divisions are necessary. The adaptive discretization is controlled by a scalar called the distance-size ratio (D). The algorithm ensures that all tesseroïds will have dimensions smaller than D times the distance to the computation point. The value of D indirectly controls the accuracy of the integration as well as the computation time.

We performed an error analysis to determine the optimal value of D required to achieve a target accuracy. We used a spherical shell as a reference to calculate the computation error of our algorithm for different values of D . Our results show that the values of D required to achieve a maximum error of 0.1% of the shell values are 1 for the gravitational potential, 1.5 for the gravitational acceleration, and 8 for the gravity gradients. Previous assumptions in the literature were that accurate results are guaranteed if the distance to the tesseroïd is larger than the distance between point masses. This condition was previously applied indiscriminately to both the gravitational acceleration and the gravity gradients. That assumption is equivalent to using $D = 1.5$ for all fields. Our results show that this is valid for the gravitational acceleration and results in a 0.1% computation error. This is expected because the original study that determined the above condition was performed on the vertical component of gravitational acceleration. However, applying the same condition to the gravity gradients produces an error of the order of 10²%.

For the gravity gradients in particular, the distance-size ratio required for 0.1% error decreases with height. This can be attributed to the higher sensitivity of gravity gradients to height variations because they decay with the distance cubed. This behavior causes the integration kernel to have more high frequency components, thus requiring more point

masses for numerical integration.

The values of the distance-size ratio determined above were incorporated as defaults in the software *Tesseroids*. We chose the value $D = 8$ for the gravity gradients as a conservative default. If the user desires, the value of D used can be controlled by a command-line argument.

In situations that require many tesseroid divisions, the stack used in the algorithm will overflow and further divisions become impossible. The current implementation warns the user that the overflow occurred and proceeds with the GLQ integration without division. Future improvements to the algorithm include a better way to handle such situations as they arise. An alternative would be to replace the tesseroid by an equivalent right rectangular prism and compute its effects instead. This would allow accurate computations at smaller distances. Furthermore, the computation time increases drastically as the computation point gets closer to the tesseroid. This effect can be prohibitive for computing the gravity gradients at relatively low heights (e.g., for terrain corrections of ground or airborne surveys). Further investigation of different criteria for dividing the tesseroids could yield better performance through a reduced number of divisions.

6 Acknowledgments

The authors were supported in this research by a fellowship (VCFB) from Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and a scholarship (LU) from Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Brazil. Additional support for the authors was provided by the Brazilian agency FAPERJ (grant E-26/103.175/2011).

Referências

- Álvarez, O., M. Gimenez, C. Braitenberg, and A. Folguera, 2012, GOCE satellite derived gravity and gravity gradient corrected for topographic effect in the south central andes region: GOCE derivatives in the south central andes: *Geophysical Journal International*, **190**, 941–959.
- Asgharzadeh, M. F., R. R. B. von Frese, H. R. Kim, T. E. Leftwich, and J. W. Kim, 2007, Spherical prism gravity effects by Gauss-legendre quadrature integration: *Geophysical Journal International*, **169**, 1–11.
- Barrera-Figueroa, V., J. Sosa-Pedroza, and J. López-Bonilla, 2006, Multiple root finder algorithm for legendre and Chebyshev polynomials via Newton’s method: *Annales Mathematicae et Informaticae*, 3–13.
- Bouman, J., J. Ebbing, and M. Fuchs, 2013a, Reference frame transformation of satellite gravity gradients and topographic mass reduction: *Journal of Geophysical Research: Solid Earth*, **118**, 759–774.
- Bouman, J., J. Ebbing, S. Meekes, R. Abdul Fattah, M. Fuchs, S. Gradmann, R. Hagmanns, V. Lieb, M. Schmidt, D. Dettmering, and W. Bosch, 2013b, GOCE gravity gradient data for lithospheric modeling: *International Journal of Applied Earth Observation and Geoinformation*.
- Braitenberg, C., 2015, Exploration of tectonic structures with GOCE in Africa and across-continent: *International Journal of Applied Earth Observation and Geoinformation*, **35**, Part A, 88–95.
- Braitenberg, C., P. Mariani, and A. De Min, 2014, The European alps and nearby orogenic belts sensed by GOCE: *Bollettino di Geofisica Teorica e Applicata*.

- Braitenberg, C., P. Mariani, J. Ebbing, and M. Sprlak, 2011, The enigmatic Chad lineament revisited with global gravity and gravity-gradient fields: Geological Society, London, Special Publications, **357**, 329–341.
- Fullea, J., J. Rodríguez-González, M. Charco, Z. Martinec, A. Negredo, and A. Villaseñor, 2014, Perturbing effects of sub-lithospheric mass anomalies in GOCE gravity gradient and other gravity data modelling: Application to the atlantic-mediterranean transition zone: International Journal of Applied Earth Observation and Geoinformation.
- Grombein, T., K. Seitz, and B. Heck, 2013, Optimized formulas for the gravitational field of a tesseroid: Journal of Geodesy, **87**, 645–660.
- Heck, B., and K. Seitz, 2007, A comparison of the tesseroid, prism and point-mass approaches for mass reductions in gravity field modelling: Journal of Geodesy, **81**, 121–136.
- Hildebrand, F. B., 1987, Introduction to numerical analysis: Dover Publications.
- Hunter, J. D., 2007, Matplotlib: A 2d graphics environment: Computing in Science & Engineering, **9**, 90–95.
- Ku, C. C., 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**, 610–622.
- Li, Z., T. Hao, Y. Xu, and Y. Xu, 2011, An efficient and adaptive approach for modeling gravity effects in spherical coordinates: Journal of Applied Geophysics, **73**, 221–231.
- Mariani, P., C. Braitenberg, and N. Ussami, 2013, Explaining the thick crust in paraná basin, Brazil, with satellite GOCE gravity observations: Journal of South American Earth Sciences, **45**, 209–223.
- McKinney, W., 2010, Data structures for statistical computing in python: Proceedings of the 9th Python in Science Conference, 51 – 56.
- Pérez, F., and B. E. Granger, 2007, IPython: a system for interactive scientific computing: Computing in Science and Engineering, **9**, 21–29.
- Ramachandran, P., and G. Varoquaux, 2011, Mayavi: 3d visualization of scientific data: Computing in Science & Engineering, **13**, 40–51.
- Reguzzoni, M., D. Sampietro, and F. Sanso, 2013, Global Moho from the combination of the CRUST2.0 model and GOCE data: Geophysical Journal International.
- Silva, J., D. Santos, and K. Gomes, 2014, Fast gravity inversion of basement relief: GEOPHYSICS, G79–G91.
- Wild-Pfeiffer, F., 2008, A comparison of different mass elements for use in gravity gradiometry: Journal of Geodesy, **82**, 637–653.

PRÓXIMAS ETAPAS DO TRABALHO DE PESQUISA

Atividades de pesquisa previstas para o próximo período:

1. Submissão do artigo sobre a modelagem direta utilizando tesseroides para a revista Geophysics.
2. Aplicação a dados sintéticos e dados reais da inversão para determinação do relevo de interfaces em coordenadas esféricas.
3. Escrita e submissão do artigo sobre a inversão de interfaces.
4. Elaboração da tese baseada nos dois artigos.
5. Defesa.

Atividades acadêmicas previstas para o próximo período:

Nenhuma.

Cronograma detalhado das atividades:

Atividade	Período de execução
Submissão do artigo de modelagem direta	Março
Finalização da implementação da inversão de interfaces	Março
Aplicação da inversão de interfaces	Abril-Maio
Escrita do artigo de inversão de interfaces	Abril-Agosto
Finalização da tese	Agosto-Setembro
Defesa	Outubro

Data prevista de conclusão do mestrado ou doutorado: Outubro/2015