Scripts to calculate the Vertical Gravity Gradients response of a 3D lithospheric model using spherical coordinates: the Caribbean oceanic domain as a case study

(http://doi.org/10.5880/GFZ.4.5.2019.002)

Ángela María Gómez-García[1,2], Christian Meeßen[3,4], Magdalena Scheck-Wenderoth[3] Gaspar Monsalve[1], Judith Bott[3], Anne Bernhardt[5], Gladys Bernal[1]

[1]Universidad Nacional de Colombia – Medellín. Grupo de investigación OCEANICOS.
[2]CEMarin – Corporation Center of Excellence in Marine Sciences.
[3]German Research Centre for Geosciences – GFZ, Potsdam, Germany.
[4]University of Potsdam, Institute for Earth and Environmental Sciences, Potsdam, Germany.
[5] Institute of Geological Sciences, Freie Universität Berlin, Berlin, Germany.

1. Licence

Scripts:

The "Scripts to calculate the Vertical Gravity Gradients response of a 3D lithospheric model using spherical coordinates" are published with a GNU General Public License, Version 3, 29 June 2007

Authors: Ángela María Gómez-García and Christian Meeßen.

Model input data:

In addition to the codes, we are providing the geophysical data used in the lithospheric starting model (model input data), except for the GEMMA model files. The sources of these data are described Section 4 of this README. Republication of subsets of these data are with permission of the authors or allowed by the licences of the input data.

Table of Contents

1. Licence 1

2. Citation 3

3. Purpose 3

4. Description of the geophysical data used in the lithospheric starting model 3

4.1 GEBCO 3

4.2 NOAA sediment thickness 4

4.3 GEMMA 4

4.4 SL2013sv tomography model 4

4.5 VGG from EIGEN-6C4 4

4.6 Earthquakes from the International Seismological Centre 5

5. Description of folders available in the data repository 6

6. Prerequisites 7

6.1 Setting up Python 8

7. Workflow 9

Step 1: Defining the SM (Starting Model) 10

Step 2: Assigning density solutions for lithospheric layers 11

Step 3: Modelling of VGG 13

Step 4: Applying Gaussian filter 15

Step 5: Calculating VGG residuals 16

Step 6: Repeat for all model configurations 17

Step 7: Sensitivity analysis: choosing best-fit starting model 17

Step 8: Identification of tectonic/terrain boundaries 17

Step 9: Inferring the crystalline crust density 17

8. Scripts to reproduce the figures of Gomez-García et al. (2019) 18

9. References 18

2. Citation

When using the code please cite:

Gómez-García, Á. M.; Meeßen, C.; Scheck-Wenderoth, M.; Monsalve, G.; Bott, J.; Bernhardt, A.; Bernal, G. (2019): Scripts to calculate the Vertical Gravity Gradients response of a 3D lithospheric model using spherical coordinates: the Caribbean oceanic domain as a case study. GFZ Data Services. http://doi.org/10.5880/GFZ.4.5.2019.002

When using the input datasets (described in Section 3 of this document), please cite the original sources provided in Section 4.

These scripts and the associated data provide the Workflow to reproduce the methodology of the paper:

Gómez-García, Á. M.; Meeßen, C.; Scheck-Wenderoth, M.; Monsalve, G.; Bott, J.; Bernhardt, A.; Bernal, G. (2019): 3D Modelling of Vertical Gravity Gradients and the delimitation of tectonic boundaries: The Caribbean oceanic domain as a case study. Geochemistry, Geophysics, Geosystems. https://doi.org/10.1029/2019GC008340

The modelling output data are published as:

Gómez-García, Á. M.; Meeßen, C.; Scheck-Wenderoth, M.; Monsalve, G.; Bott, J.; Bernhardt, A.; Bernal, G. (2019): Average crustal densities and main terrain boundaries of the Caribbean oceanic domain inferred from the modelling of Vertical Gravity Gradients. GFZ Data Services. http://doi.org/10.5880/GFZ.4.5.2019.001

3. Purpose

This workflow aims to help the users with the reproducibility of the step-by-step processes needed to obtain the residuals of the Vertical Gravity Gradients (VGG or Tzz), which allow the identification of tectonic/terrain boundaries, and the crustal density field. It also contains the scripts to reproduce the paper figures.

4. Description of the geophysical data used in the lithospheric starting model

In this research, we used different geophysical datasets to constrain the 3D lithospheric starting model. Hereafter, we summarise the sources of these data. It is important to mention that we included all the necessary files to execute the present workflow, except the GEMMA model files, which the user can download directly following the steps provided in Section 4.3. The GEMMA files must be storage on the folder "Data" of this repository.

4.1 GEBCO

We used "The GEBCO_2014 Grid, version 20141103, http://www.gebco.net", which has a 30 arc sec spatial resolution (Becker et al., 2009). In this case, we cut out the region around the Caribbean domain, between 6°S and 35°N, and 120°W and 50°W, and is available in the file "GEBCO_2014_2D_-120.0_-6.0_-50.0_35.0.nc".

4.2 NOAA sediment thickness

The file "sedthick_world_v2.grd.gz" can be downloaded from "https://www.ngdc.noaa.gov/mgg/sedthick/data/version2/gmt_netcdf/. [Accessed 30 August 2019]" (Whittaker et al., 2013), and after unzip it, you will get a .grd file which is the one we used in the present workflow.

4.3 GEMMA

The current version of GEMMA model (Reguzzoni & Sampietro, 2015) is available at "http://gocedata.como.polimi.it/. [Accessed 30 August 2019]". There, the user has to click on "GEMMA data" option, then scroll down and click on "submit" button. This will charge a new window with the option "Download Moho (with error estimate)". Once you click there, the "moho.zip" file will be downloaded. It contains two files: the Moho depths named "t6.asc", and the error of the Moho estimations in "moho_err.asc". The user should rename the "t6.asc" as "mantleUP_top.asc", so the corresponding scripts can recognise the correct file. **Both files must be storage on the folder "Data" of this repository**.

4.4 SL2013sv tomography model

Here we include a cut out region of the original SL2013sv tomography model, with permission from the authors (Schaeffer & Lebedev, 2013). The complete model can be obtained from "https://schaeffer.ca/tomography/sl2013sv/. [Accessed 30 August 2019]". The selected region corresponds to the Caribbean oceanic domain (95°W, 55°W and 5°N, 25°N), and the mantle layers range from 25 km to 200 km depth.

4.5 VGG from EIGEN-6C4

We used the ICGEM calculation service (Ince et al., 2019) web page "http://icgem.gfz-potsdam.de/calcgrid. [Accessed 30 August 2019]" to obtain the second derivatives of the disturbance potential, namely Vertical Gravity Gradients. Figure 1 shows a screenshot of the web page with the selected parameters that were used for the calculated gradients based on EIGEN-6C4 dataset (Förste et al., 2014). Once all the parameters have been set, click on "start computation" button and wait until the system finishes. If you want to use this generated file in the next steps, you must rename it as "EIGENTzz_0_05.gdf".

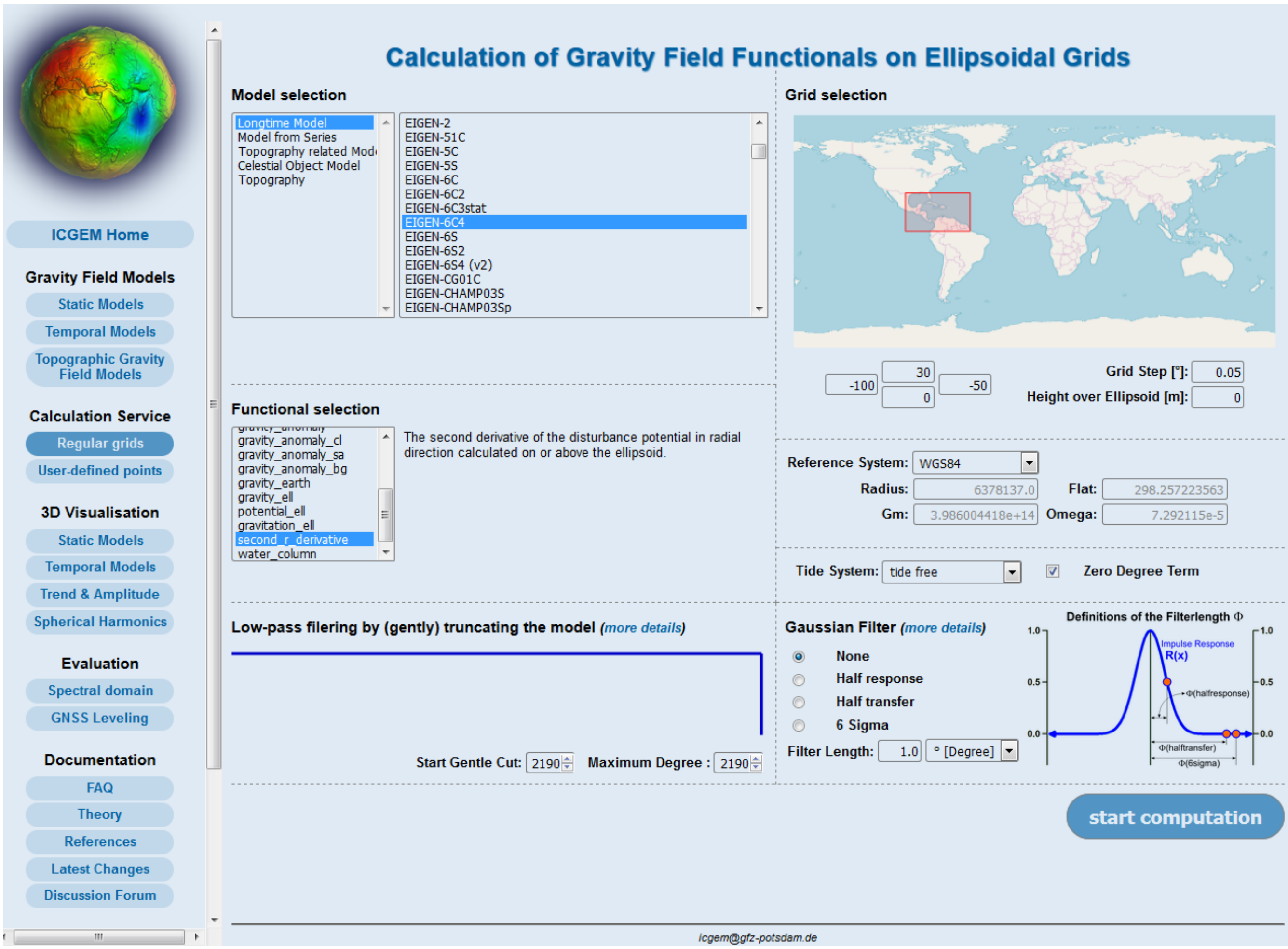

Figure 1. *ICGEM Calculation Service web page showing the selected parameters for the calculation of the VGG from EIGEN-6C4 model.*

4.6 Earthquakes from the International Seismological Centre

Earthquake data were downloaded from the ISC Bulletin (International Seismological Centre, 2019). This is regarded as the definitive record of the Earth's seismicity, based on data provided by numerous agencies, and reviewed by ISC.

In the online search form "http://www.isc.ac.uk/iscbulletin/search/bulletin/. [Accessed 30 August 2019]", the selected database was "Reviewed ISC Bulletin", in "ISF Bulletin" output format (an ASCII format later converted into rows and columns before processing). A "rectangular search" was made in the region 95°W, 55°W and 5°N, 25°N, since January 1st, 1980, 00:00:00 until December 1st, 2016, 00:00:00 (the latest date for which the catalogue had been reviewed by ISC at the time of download). Earthquakes with unknown depths and/or magnitudes were excluded.

A minimum magnitude of 4.0 was chosen, for any magnitude type, calculated by the "prime author", the agency defined by ISC as the preferred one for each earthquake. As output options, only prime hypocentres were chosen, accompanied by magnitudes.

5. Description of folders available in the data repository

This data and software publication includes different folders and files which description can be found in Table 1.

Table 1. *General information of the folders and files available on this repository.* [1] *GEMMA model files that* *<u>the user must download</u>* *following the instructions provided in Section 4.3.*

Folder	Content	Type		Description
		Folder	File	
Data	dVsMean	x		Tomography data from Schaeffer & Lebedev (2013)
	Shapefiles	x		Selected shapefiles from different sources. See details on the Supplemenary information of the paper.
	VsMean	x		Tomography data from Schaeffer & Lebedev (2013)
	EIGENTzz_0_05.gdf		x	Vertical Gravity Gradients processed and downloaded on the ICGEM Calculation Service (Ince et al., 2019) with a spatial resolution of 0.05°.
	GEBCO_2014_2D_-120.0_-6.0_-50.0_35.0.nc		x	Bathymetry and topography grid as originally downloaded from GEBCO web page (Becker et al., 2009).
	mantleUP_top.asc[1]		x	Moho depth from the GEMMA model (Reguzzoni & Sampietro, 2015).
	moho_err.asc[1]		x	Moho depth error from the GEMMA model (Reguzzoni & Sampietro, 2015).
	sedthick_world_v2.grd		x	Sediment thickness as originally downloaded from the NOAA web page (Whittaker et al., 2013).
	Terremotos_Caribe_ISC.xlsx		x	Seismicity of the Caribbean region from the ISC Catalog (International Seismological Centre, 2019).
Output	Precomputed	x		Folder containing the results of all the precomputed starting models (SM1 to SM-F).

	Tomography	x		Initially it is an empty folder, but it will contain the results of the tomography after running the code "SL2013_200km.py" on step 1.
	Velocity2Density.bash		x	Bash code to convert mantle tomography velocity anomalies to mantle densities according to Meeßen (2017).
	Velocity2Density_v1.1.bash		x	Updated bash code to convert mantle tomography velocity anomalies to mantle densities according to Meeßen (2017).
	CaribbeanMineralogy.csv		x	Mineralogy of the upper mantle used to obtain mantle densities according to Meeßen (2017).
Scripts	N/A	x		This folder contains all the scripts necessary to reproduce the workflow of the paper. In this document we explain how to use them step by step.
Setup	N/A	x		It has the "environment.yml" file which will be used to create an anaconda environment to run all the Python codes available in this repository.

6. Prerequisites

The workflow presented here was tested on Linux. It is important to note that if you wish to follow this workflow and run the scripts, you will need:

- **Python v2.7** (Van Rossum & Drake, 1995)
- **GMT v5.1.1**(Wessel et al., 2013)
- **Tesseroids v1.2.1** (Uieda et al., 2016)
- **VeloDT v1.0.1** (Meeßen, 2017)

The **VeloDT v1.0.1** package should be installed in the "Output" folder of this repository.

Additionally the Python libraries:

- **Matplotlib** (Droettboom et al., 2018)
- **Numpy** (Oliphant & Millma, 2006; Van Der Walt et al., 2011)
- **Pandas** (Mckinney, 2010)
- **netCDF4** (Whitaker et al., 2019)
- **Scipy** (Jones et al., 2001)

- **scikit-learn** (Pedregosa et al., 2011)

6.1 Setting up Python

We recommend to use Miniconda (https://docs.conda.io/en/latest/miniconda.html) to install all required packages. Once you have downloaded Miniconda, navigate to the "Setup" folder and run

```
conda env create –f environment.yml
```

This will create an anaconda environment named "gomezetal" that contains all necessary Python packages. Prior to executing Python scripts make sure to activate the environment with

```
conda activate gomezetal
```

It can be closed with

```
conda deactivate
```

7. Workflow

Figure 2 shows the methodological scheme including the step-by-step (in blue) that will be described with more detail in this document.

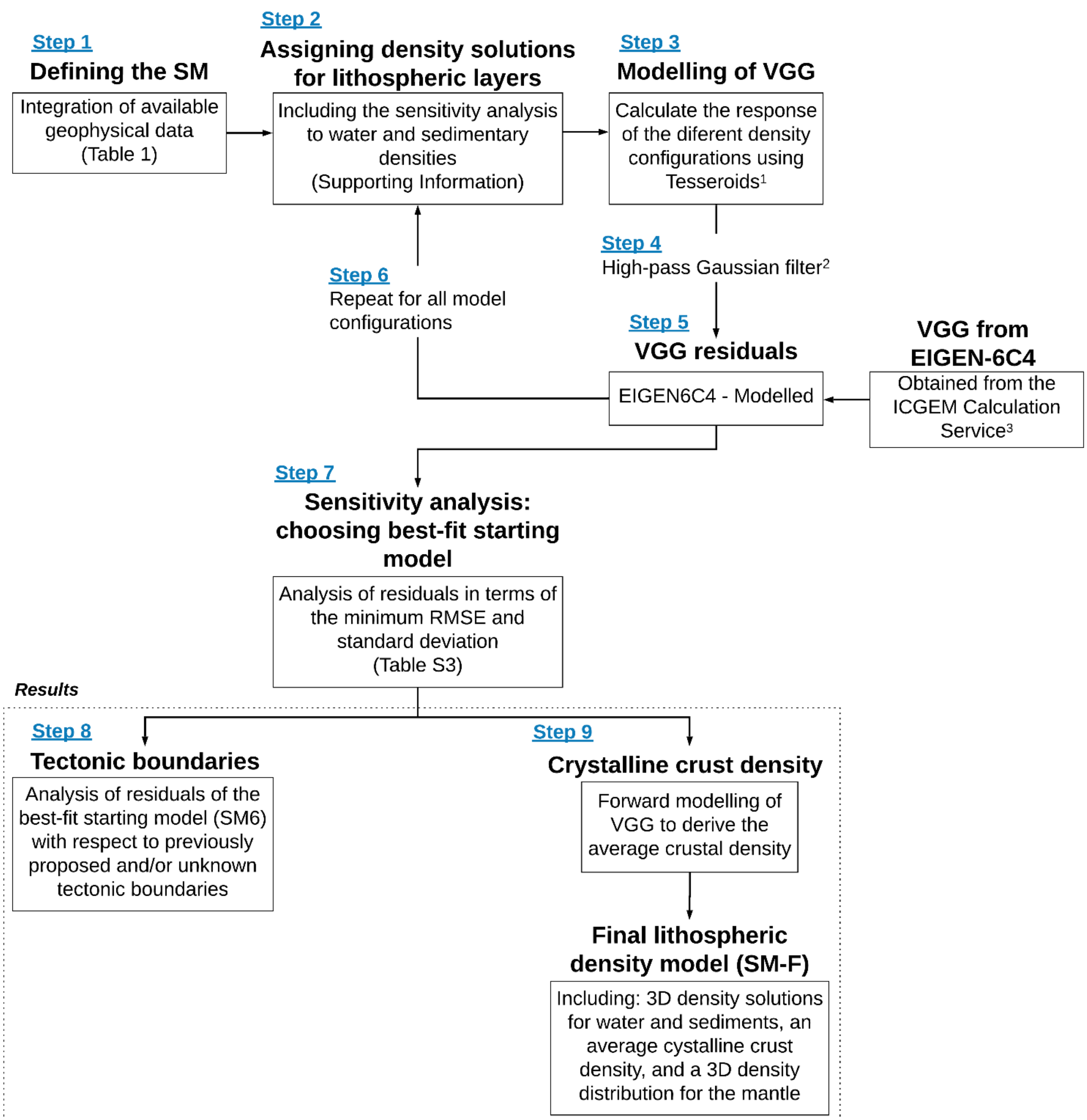

Figure 2. *General methodological scheme used in the research. [1]Uieda et al. (2016). [2]Wessel et al. (2013). [3]ICGEM (2018). SM = Starting Model.*

Step 1: Defining the SM (Starting Model)

The first step is to download all the geophysical information available in the different datasets (please refer to Section 4 of this Readme file). Here, we provide the reader with the downloaded files from each dataset (except GEMMA, see Section 4.3). A synthesis of the datasets used to constrain the lithospheric model is presented in Table 2.

Table 2. *Datasets used to define the lithospheric starting model.*

Layer / Interface	Dataset	File name in "Data" folder	Reference
Bathymetry	GEBCO bathymetry (30 arc sec)	GEBCO_2014_2D_-120.0_-6.0_-50.0_35.0.nc	Weatherall et al. (2015)
Sediments	NOAA sediment thickness (Version 2)	sedthick_world_v2.grd	Whittaker et al. (2013)
Moho depth	GEMMA	mantleUP_top.asc	Reguzzoni & Sampietro (2015)
Mantle (down to 200km depth)	SL2013sv (Version v2.1)	Folders "dVsMean" and "VsMean"	Schaeffer & Lebedev (2013)

Due to the fact that these datasets have different spatial resolutions, it is necessary to interpolate them to a homogeneous cell size. In our case we worked with 0.05° x 0.05°. Additionally to these datasets, we used the Vertical Gravity Gradients from EIGEN-6C4 model (Förste et al., 2014) calculated from the ICGEM Calculation Service web page (Ince et al., 2019).

The Python scripts listed on Table 3 will save the corresponding dataset as ASCII files (X,Y,Z), because latter on they will be used for setting up the Tesseroids files.

Table 3. *Scripts and associated outputs for processing the datasets.*

Script	Outputs	Output Description
GEBCO_bath.py	Gebco_30arcsec.png Gebco_0.05.png	Figures of GEBCO data in the original and interpolated resolution, respectively.
	GEBCO_stack_0.05.txt GEBCO_stack_0.05.npy	Files with three columns corresponding with longitude, latitude and elevation of the study area.
NOAA_sedth.py	SedimentThickness_0.5.png SedimentThickness_0.05.png	Figures of sediment thickness data in the original and interpolated resolution, respectively.
	NOAA_stack_0.05.txt NOAA_stack_0.05.npy	Files with three columns corresponding with longitude, latitude and sediment thickness of the study area.
GEMMA_moho.py	Moho_GEMMA_0.5.png Moho_GEMMA_0.05.png	Figures of Moho depths in the original and interpolated resolution, respectively.
	Moho_Gemma_0.05.txt Moho_Gemma_0.05.npy	Files containing three columns corresponding to longitude, latitude, and depth to Moho.
SL2013_200km.py	SL_stack_0_05.txt <depth>stack_0_05.txt	Files containing four columns corresponding to longitude, latitude, depth and velocity anomalies (in m/s). We will use the first file because it contains all the information from 25 to 200 km depth. These files are stored in the folder "Output/Tomography".
EIGEN6C4_VGG.py	EIGEN6C4_Tzz.png	Figure of the VGG (Tzz) from EIGEN-6C4 dataset in the original resolution.
	EIGEN-Tzz_0_05.txt	File containing three columns corresponding to longitude, latitude, and the value of Tzz (in Eötvös).

Execute the scripts by navigating to the "Scripts" folder, and running for example:

```
python GEBCO_bath.py
```

Be aware that the Current Working Directory (CWD) of your Python environment should be the folder called "Scripts", which is inside the main repository folder.

Step 2: Assigning density solutions for lithospheric layers

The next step is to assign density distributions to the different layers of the lithospheric model. Table S1 (see Supporting Information of the paper) presents the configurations we tested. The Python codes listed on Table 4 are useful for the calculation of the three dimensional (3D) approaches for the densities of the water, sediments and mantle.

For the mantle densities, we additionally used VelocityConversion v1.0.1, written by Meeßen (2017) (https://github.com/cmeessen/VelocityConversion/releases/tag/v1.0.1). It requires the mineralogy of the mantle, which in our case is in the file "Output/CaribbeanMineralogy.csv".

> **Note**: The conversion with v1.0.1 is very time consuming. Download the latest version of VelocityConversion by cloning the repository into the "Output" folder:
>
> ```
> cd ../Output
> git clone https://github.com/cmeessen/VelocityConversion.git
> ```
>
> and then execute "Velocity2Density_v1.1.bash" instead of "Velocity2Density.bash" (both files are in the "Output" folder).

Table 4. *Description of the codes for computing the three dimensional densities for water and sediments.*

Script	Output	Output Description
Water_3D_density.py	3D_water_density.npy	Density of the water calculated according to Gladkikh & Tenzer (2012).
	Th_water.npy	Water layers in which the density has been computed. We used a total of 84 layers, 100 m thick each.
	rhoVsDepth_3DWater.png	Profile of water densities with depth. It corresponds to the FigureS3 (a) of the Supporting Information of Gomez-Garcia et al. (2019)
Sediments_3D_density.py	3D_density_Tenzer.npy	Density of the sediments calculated according to Tenzer & Gladkikh (2014). We apply the bedrock density contrast correction of Chen et al. (2014), using the maximum density expected for shales (2750 kg m^3) according to Schön (2011).
	Th_sediments.npy	Sedimenary layers in which the density has been computed. We used a total of 181 layers, 100 m thick each.
	rhoVsDepth_3DSedim_2750.png	Profile of sedimentary densities with depth. It corresponds to the FigureS3 (b) of the Supporting Information of this paper.

Then, the steps to calculate the 3D mantle densities are:

1. Run “Velocity2Density.bash”, which will perform the velocity to density conversion of Meeßen (2017). This step will take a while.

```
cd ../Output
bash Velocity2Density.bash
```

2. Run “Mantle_3D_density.py” that will generate the files necessary for building the Tesseroids input.

```
cd ../Scripts
python Mantle_3D_density.py
```

To assign constant densities for the water and sediments, please refer to the script in step 3.

Step 3: Modelling of VGG

Tesseroids requires the thickness and the density for each point of the grid, as well as the calculation height at which the VGG will be computed. Initially, the crystalline crust density was defined as constant (2810 kg m^{-3} for the continents, and 2900 kg m^{-3} for the ocean). In this case, the script “MakeFile_Tesseroids_ConstantCrust.py” will generate the input files for running Tesseroids. In this code, it is possible to assign the 3D density fields calculated in step 2 for the water and the sediments, or define constant values. You just need to modify lines 20 to 23 directly on the script.

The given example has the configuration according with the starting model SM3, but it is possible to define other values, for example, for other density configurations as in Table 5.

It is also possible to obtain the average density of the crystalline crust by means of forward modelling of the VGG, which corresponds to the SM-F configuration (Table 5). This part of the workflow will be described in the step 9.

Table 5. *Different density (kg m^{-3}) configurations used in the starting models (SM) tested in this study. [†]Heterogeneous density of the crystalline crust obtained from the forward modelling of the VGG (see step 9).*

<table>
<tr><th><version></th><th>Water</th><th>Sediments</th><th>Crystalline crust</th><th>Mantle</th></tr>
<tr><td>SM1</td><td rowspan="4">1030</td><td>1700</td><td rowspan="6">2810 (continental)
2900 (oceanic)</td><td rowspan="7">3D α(P,T)</td></tr>
<tr><td>SM2</td><td>2350</td></tr>
<tr><td>SM3</td><td>2610</td></tr>
<tr><td>SM4</td><td>3D</td></tr>
<tr><td>SM5</td><td>3D</td><td>2350</td></tr>
<tr><td>SM6</td><td>3D</td><td>3D</td></tr>
<tr><td>SM-F</td><td>3D</td><td>3D</td><td>2700-3100[†]</td></tr>
</table>

The outputs of the make file script are listed on Table 6.

Table 6. *Outputs of the "MakeFile_Tesseroids_ConstantCrust.py" script.*

Output	Output Description
Caribbean_0_05_DfromT_**<version>**.txt Caribbean_0_05_DfromT_**<version>**.npy	Files with the information of layer thicknesses and densities as required by the Tesseroids "tesslayers" package.
xyz_calc_heights.xyz	Heights above the ellipsoid at which the VGG will be computed. In this case it is defined as 1 meter above the mean sea level or above the topography (depending on the location within the modelling domain).
densities_**<version>**.txt	This file contains the information about the densities used in the configuration of the corresponding starting model. The header has more details.

Where **<version>** is a string internally defined on the script (line 14), and should be written according with the first column of Table 5, depending on which density configuration you are assigning to the water and sedimentary layers.

Be aware that the "MakeFile_Tesseroids_ConstantCrust.py" script only works for <u>constant density values of the crystalline crust.</u> To assign density variations to this layer, please refer to step 9.

After you have built all the input files from the previous steps, now it is time for running Tesseroids in order to calculate the VGG (T_{zz}). For this, you will need to execute the bash codes listed on Table 7, **<u>in this strict order</u>**, otherwise you will not have the correct input files.

The third script on Table 7 will carry out a parallel computation of the gravity derivatives; therefore if you go to line 28 within the code, you will be able to assign the threads number you would like to use.

Note: For the scripts to run, the binaries of Tesseroids must be accessible. Make sure that the PATH variable contains the path to the Tesseroids binaries.

__Table 7.__ Steps to generate the spherical prisms grid and calculate the VGG using Tesseroids packages.

Script	Output (s)	Output Description
1. TessLayers.sh	Caribbean_0_05_tess.txt	Tesseroid model generated by the package "tesslayers" (see Tesseroids documentation)
2. remove_zero_tess.sh	Caribbean_tesslayers.dat	The tesseroid model after removing the zero thickness prisms
3. Run_Tesseroids_Parallel.sh	Tzz_**<version>**.dat	ASCII file with four rows: 1. Longitude, 2. Latitude, 3. Elevation (bathymetry of topography) and 4. Tzz (calculated with Tesseroids).
	log_**<machine-name>**.txt	Contains information about the start and finish times of the calculation

Please have a look at the script's header before executing it, some of them need few modifications depending on the starting model <version> that you have configured, or have important information regarding what they do.

The "Tzz_**<version>**.dat" file will have NaN values in the most external rows and columns (i.e: 5°, 25°, -95° and -55°). We fix it later because these regions are not part of our analysis, but belong to the area where the boundary effects take place (that is the reason why you should model a bigger region compared with what actually is going to be your study area).

Step 4: Applying Gaussian filter

Due to the fact that the EIGEN-6C4 dataset has a spherical harmonic solution up to degree and order Nmax=2190, the smallest topographic wavelength that we are able to resolve can be calculated as $\lambda_{\mathrm{min}} = 2\pi\mathrm{R}/\mathrm{N}_{\mathrm{max}} \cong 18\ \mathrm{km}$. Thus, it is important to be sure that the modelled results have the same wavelength. For this reason, the next step is to apply a Gaussian filter using GMT.

The code "HPass_Gaussian_Tzz.sh" will carry out this with a Gaussian high-pass filter set up at 18 km. Before running it, please modify the **<version>** internally, according to the model you are working with (see Table 5). This script will give as output the files listed on Table 8.

Table 8. *Output files of the "HPass_Gaussian_Tzz.sh" script.*

Output	Output Description
PowerSpectrum_**<version>**.ps	Figure of the radial power spectrum of modelled Tzz (before filtering).
Tzz_PowerSpectrum_**<version>**.txt	ASCII file with the power spectrum of the Tzz before filtering.
PowerSpectrum_18km_**<version>**.ps	Power spectrum of filtered Tzz.
Tzz_PowerSpectrum_18km_**<version>**.txt	ASCII file with the power spectrum of the Tzz after filtering.
Tzz_HP_18km_**<version>**.nc	Contains the modelled Tzz grid after applying the Gaussian filter. **This is the file that will be used in the next steps of the workflow.**

Step 5: Calculating VGG residuals

The Python script "Residuals_VGG_FigS2.py" calculates and plots the residuals of the VGG. To use this code, it is important that you modify the **<version>** and additional variables (described on script lines 28 and 29) that change based on the starting model you are working with. The outputs of this script are listed on Table 9.

Considering that the calculation of the VGG might take a lot of time, we provide the results of all starting models in the "Output/Precomputed" folder. In this script you can select whether or not calculate the residuals using the results of this folder. For this you must modify line 40 of the code, and set the "precomputed" variable as `True` or `False`.

This script can also be used to generate the Figure S2 of the paper (Gómez-García et al. 2019).

Table 9. *Output files of the "Residuals_VGG_FigS2.py" script.*

Output	Output Description
Tzz_Filtered_18km _v**<version>**.txt	ASCII file containing three columns: Lon, Lat and Residuals. Cropped to: **Longitude: -94°, -56°** **Latitude: 6°, 24°** This region is smaller than the modelled domain aiming to avoid boundary effects. The file has been masked in the continental areas.
Residuals_**<version>**.png	Figure of the VGG residuals including the tectonic boundaries from Bird (2003) and Ahlenius (2014).

Statistics_**<version>**.txt	ASCII file with the five statistics of the residuals that will be used for comparing the different starting models (see steps 6 and 7): **Standard deviation** **Root Mean Squared Error (RMSE)** **Mean** **Minimum** **Maximum**

Step 6: Repeat for all model configurations

The sensitivity analysis consists in changing the density solutions (or distributions) of the starting model. For this, you should perform steps 2 to 5 for all model configuration in Table 5.

❗ **Please, be aware of changing the <version> and other parameters inside the codes (when required), otherwise you will rewrite files with the wrong information inside.**

Step 7: Sensitivity analysis: choosing best-fit starting model

Using the file "Statistics_**<version>**.txt" generated for each starting model on step 5 (see Table 9), it is possible to build a table such as the Table S2 of the Supporting Information of the paper.

This table will allow the comparison of the different models in terms of the statistics relative to EIGEN-6C4 dataset.

Step 8: Identification of tectonic/terrain boundaries

The code "Tzz_Residuals_Contours.py" plots the residuals of the starting model SM6, which has been used for the identification of terrain boundaries, following the contours where Tzz=0. This is our main hypothesis, so please refer to the paper in case you need additional information regarding the methodology.

The output of the script is a figure named "Tzzres_vSM6 _Contours.png". We used it to digitalise the main boundaries in ArcGIS, but one can use an open source GIS software as well.

Please notice that if you modify the starting model in the script, you will end with different contour lines because the residuals are different for each density model.

Step 9: Inferring the crystalline crust density

All the previous starting models have assumed a constant density of the crystalline crust. However, it is possible to forward model the VGG, in order to obtain a crustal density solution which minimise de residuals, and therefore, that will be gravity validated.

The script "MakeFile_Tesseroids_VariableCrust.py" is designed to generate a Tesseroids input file, like the one of the step 3, but iterating the density of the crust based on the residuals from previous iterations. Here, we give two files from a previous iteration that should be used as inputs for the script. These files are:

“../Precomputed/Tzz_Filtered_18km_iteration.txt” and “../Precomputed/Rho_crust_iteration.npy”, which are the residuals and the crustal densities of an iteration previous to the final one (that you will calculate now).

After running this script, you should calculate the VGG response of the new stating model (in this case SM-F), following the steps described on Table 7 (step 3). Steps 4 and 5 are also required to compare how the residuals decreased with the use of the inferred crustal densities.

The output of the “MakeFile_Tesseroids_VariableCrust.py” is the same as the one explained on step 3, but it also includes the figure of the inferred crustal densities, as well as the “Crustal_Densities_**<version>**.txt”, which is a three columns file with the corresponding longitude, latitude and density value for the cropped area. This code will also generate the figure of the inferred crustal densities, which corresponds to the data plotted on figure 6 of the paper.

8. Scripts to reproduce the figures of Gomez-García et al. (2019)

Additionally to the previous described codes, we provide some scripts to generate Figures 3, 5, 7, 8 and 9 of the paper. These python codes are:

- Fig3_LayersThicknesses.py
- Fig5_Eigen_Model_Resid.py
- Fig7_ZoomLesserAntilles.py
- Fig8_ZoomYucatan_Cayman.py
- Fig9_InferredNewTerrains.py

9. References

Ahlenius, H.: Tectonic Plates. GitHub repository, [online] Available from: https://github.com/fraxen/tectonicplates/ (Accessed 10 May 2018), 2014.

Becker, J. J., Sandwell, D., Smith, W. H. F., Braud, J., Binder, B., Depner, J., et al. (2009). Global Bathymetry and Elevation Data at 30 Arc Seconds Resolution: SRTM30_PLUS. Marine Geodesy, 32(4), 355–371. https://doi.org/10.1080/01490410903297766

Bird, P. (2003). An updated digital model of plate boundaries. Geochemistry, Geophysics, Geosystems, 4(3), 52. https://doi.org/10.1029/2001GC000252

Chen, W., Tenzer, R., & Gu, X. (2014). Sediment Stripping Correction to Marine Gravity Data. Marine Geodesy, 37(4), 419–439. https://doi.org/10.1080/01490419.2014.932870

Droettboom, M., Caswell, T. A., Hunter, J., Firing, E., Nielsen, J. H., Varoquaux, N., et al. (2018, January 18). matplotlib/matplotlib v2.1.2. https://doi.org/10.5281/ZENODO.1154287

Förste, C., Bruinsma, S., Abrikosov, O., Lemoine, J.-M., Marty, J.-C., Flechtner, F., et al. (2014). EIGEN-6C4 The latest combined global gravity field model including GOCE data up to degree and order 2190 of GFZ Potsdam and GRGS Toulouse. GFZ Data Services. https://doi.org/http://dx.doi.org/10.5880/icgem.2015.1

Gladkikh, V., & Tenzer, R. (2012). A mathematical model of the global ocean saltwater density distribution. Pure and Applied Geophysics, 169(1–2), 249–257. https://doi.org/10.1007/s00024-011-0275-5

Gómez-García, Á. M.; Meeßen, C.; Scheck-Wenderoth, M.; Monsalve, G.; Bott, J.; Bernhardt, A.; Bernal, G. (2019). 3D Modelling of Vertical Gravity Gradients and the delimitation of tectonic boundaries: The Caribbean oceanic domain as a case study. Geochemistry, Geophysics, Geosystems. https://doi.org/10.1029/2019GC008340

ICGEM. (2018). Calculation service, GFZ Potsdam. Retrieved June 30, 2019, from http://icgem.gfz-potsdam.de/calcgrid

Ince, E. S., Barthelmes, F., Reißland, S., Elger, K., Förste, C., Flechtner, F., & Schuh, H. (2019). ICGEM - 15 years of successful collection and distribution of global gravitational models, associated services, and future plans. Earth System Science Data, 11, 647–674. https://doi.org/10.5194/essd-2019-17

International Seismological Centre. (2019). On-line Bulletin, https://doi.org/http://doi.org/10.31905/D808B830

Jones, E., Oliphant, T., & Peterson, P. (2001). SciPy: Open source scientific tools for Python -- Reference Guide, 0.14. (online) http://www.scipy.org/ (

Mckinney, W. (2010). Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference. (online) https://conference.scipy.org/proceedings/scipy2010/pdfs/mckinney.pdf

Meeßen, C. (2017). VelocityConversion. GFZ Data Services. https://doi.org/http://doi.org/10.5880/GFZ.6.1.2017.001

Oliphant, T., & Millma, J. k. (2006). A guide to NumPy. Trelgol Publishing. USA: Trelgol Publishing. https://doi.org/DOI:10.1109/MCSE.2007.58

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine Learning in Python. Journal Of Machine Learning Research, 12, 2825–2830.

Reguzzoni, M., & Sampietro, D. (2015). GEMMA: An Earth crustal model based on GOCE satellite data. International Journal of Applied Earth Observation and Geoinformation, 35(PA), 31–43. https://doi.org/10.1016/j.jag.2014.04.002

Van Rossum, G., & Drake, F. (1995). Python tutorial. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands.

Schaeffer, A. J., & Lebedev, S. (2013). Global shear speed structure of the upper mantle and transition zone. Geophysical Journal International, 194(1), 417–449. https://doi.org/10.1093/gji/ggt095

Schön, J. H. (2011). Physical Properties of Rocks: A Workbook. (J. Cubbit, Ed.). Elsevier.

Tenzer, R., & Gladkikh, V. (2014). Assessment of density variations of marine sediments with ocean and sediment depths. The Scientific World Journal, 2014. https://doi.org/10.1155/2014/823296

Uieda, L., Barbosa, V. C. F., & Braitenberg, C. (2016). Tesseroids: Forward-modeling gravitational fields in spherical coordinates. Geophysics, 81(5). https://doi.org/10.1190/geo2015-0204.1

Van Der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: A structure for efficient numerical computation. Computing in Science and Engineering, 13(2), 22–30. https://doi.org/10.1109/MCSE.2011.37

Weatherall, P., Marks, K. M., Jakobsson, M., Schmitt, T., Tani, S., Arndt, J. E., et al. (2015). A new digital bathymetric model of the world's oceans. Earth and Space Science, 2, 331–345. https://doi.org/10.1002/2015EA000107

Wessel, P., Smith, W. H. F., Scharroo, R., Luis, J., & Wobbe, F. (2013). Generic mapping tools: Improved version released. Eos, 94(45), 409–410. https://doi.org/10.1002/2013EO450001

Whitaker, J., Khrulev, C., Huard, D., Paulik, C., Hoyer, S., Filipe, et al. (2019, May 6). Unidata/netcdf4-python: version 1.5.1.2 release. https://doi.org/10.5281/ZENODO.2669496

Whittaker, J. M., Goncharov, A., Williams, S. E., Muller, R. D., & Leitchenkov, G. (2013). Global sediment thickness data set updated for the Australian-Antarctic Southern Ocean. Geochemistry, Geophysics, Geosystems, 14(8), 3297–3305. https://doi.org/10.1002/ggge.20181