SRBF.py

Python based regional gravity field determination software using spherical radial basis functions (SRBF)

User Manual

Rasit Ulug

Middle East Technical University

Ankara, Turkey

June 2020

1. Introduction

This work introduces a first open-source Python software package for the high-resolution regional gravity field determination using various spherical radial basis functions (SBRFs) in terms of point mass, Poisson, and Poisson wavelet kernel. This package will allow researchers in Earth Sciences and related fields to retrace the implementation of the algorithms used to compare and evaluate different methods in local and regional gravity field recovery. The modeling approach considers residual gravity field functionals that are generated (derived) by the well-known remove-compute-restore (RCR) technique where the long and short wavelength parts of the gravity signal are provided by a global geopotential model (GGM) and a digital terrain model (DTM), respectively. Reuter grid is employed to find out the number and center position of SRBFs. Optimal model parameters such as the Reuter grid control parameter and depth or bandwidth of SRBFs can be determined by the minimization of root-mean-square (RMS) of the differences between estimated and observed values at the selected control points. The unknown SRBF coefficients are estimated by applying the least-squares method to the extended Gauss Markov Model (GMM) with additional prior information to overcome ill-conditioned normal equations. The optimal regularization parameter is found by variance component estimation (VCE). By utilizing parallel processing, the computation time for the construction of the design matrix is remarkably decreased relative to the number of cores used in the process.

2. Installation

SRBF.py does not require any installation. In SRBF.py, we focus on standard Python libraries. Thus, we aimed to prevent the problems encountered while installing extra packaging. To avoid any trouble, we recommend using Anaconda distribution that collected all necessary packages and also including an integrated development environment (IDE), which enables easy usage of programing language.

3. SRBF.py

SRBF.py is Python-based regional gravity field modeling software that uses remove-compute-restore methodology (RCR). In this approach, long and short wavelengths of the gravity signal are subtracted from the observed signal by using state-of-art Global Geopotential Model (GGM) and Digital Terrain Model (DTM), respectively. The remaining middle wavelength of the gravity signal is modeled using SRBF, and long and short wavelengths are restored in the final solution. The SRBF.py is capable of reading a standard GGM format defined by ICGEM (International Center for Global Gravity Field Models) in which the spherical harmonics coefficients sorted according to degrees. For the computation of residual terrain model (RTM), the TC (terrain correction) module of the GRAVSOFT package is used as a subprocess. The TC module, which uses rectangular prism for the terrain reduction, is the part of the GRAVSOFT package initially developed by (Forsberg, 1984). In Fig. 1, the flow chart of software is visualized. As seen in the figure, the design of the software is simple. This facilitates interpretation and allows easy modification if desired. The dashed line in the figure shows that it will not be allowed to pass that stage without control data.

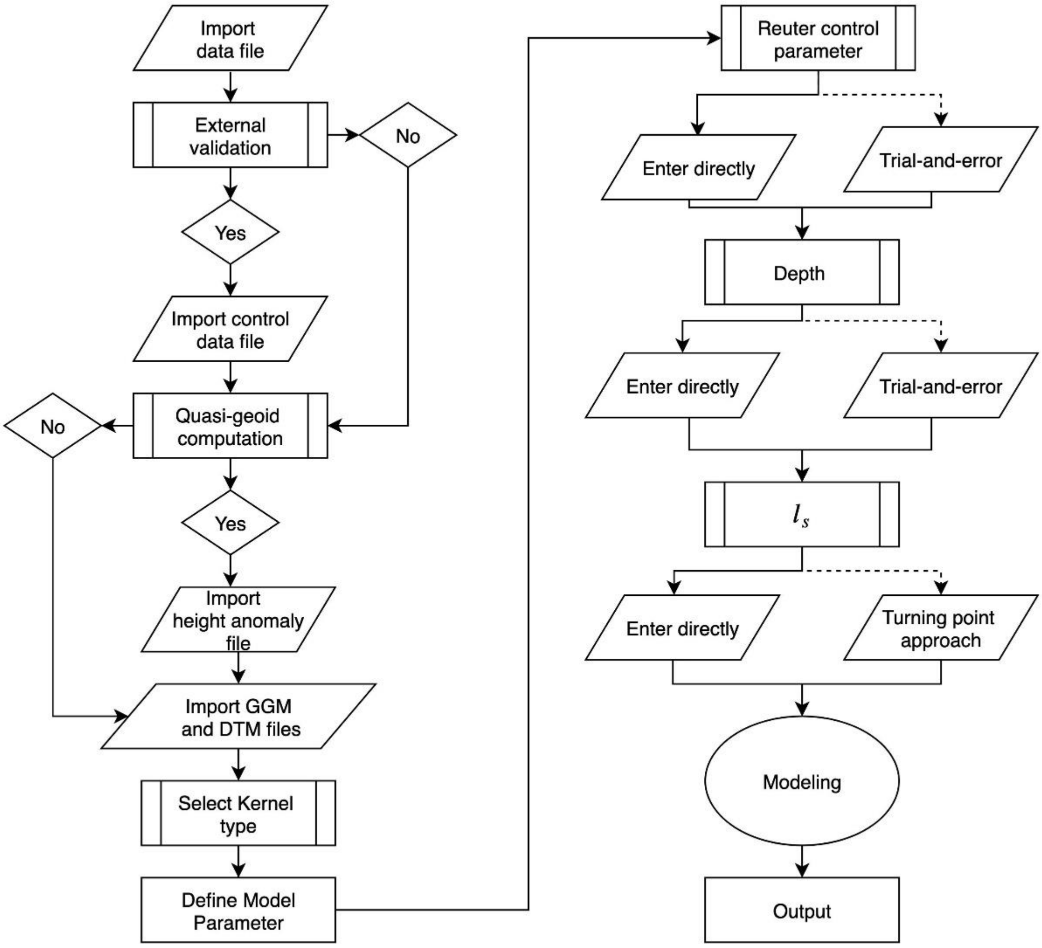


Figure. 1 Flow chart of SRBF.py

3.1. Basic scripts and files of SRBF.py

SRBF.py consists of 5 basic scripts. While they are an integral part of the program, they can also be used for other geodetic purposes. These scripts are summarized in Table 1. Among them, SRBF.py is the main script which imports others. All scripts must be stored in the same file. Besides, there are four folders: input, output, bin, and DEM to prevent clutter. The bin folder contains GRAVSOFT TC, SELECT, and TCGRID modules. While SRBF.py use TC directly, others are necessary to obtain coarse and reference DTM file, for detail see (Forsberg and Tscherning, 2008). DTM files should be put in the DEM folder; otherwise, the program will give an error.

Table 1 Script and contained functions

Script Name	Functions
SRBF.py	Main scripts
kernel_functions.py	Point mass, Poisson and Poisson wavelet kernel for disturbing potential, gravity anomaly and disturbance
coeff_matrix.py	Design matrix by using parallel processing algorithm
gravity_functions.py	Fully normalized associated Legendre polynomial, normal gravity (GRS80 ellipsoid), disturbing potential, gravity anomaly, and disturbance
other_functions.py	Spherical angle computation, geodetic to cartesian, geodetic to spherical, cartesian to spherical coordinate transformation, three functions to read input files, Monte-Carlo Variance Component Estimation (MCVCE), Reuter grid function

3.2. Data formats

3.2.1 Spherical harmonics coefficients

As already mentioned above, The SRBF.py is capable of reading a standard GGM format defined by ICGEM (International Center for Global Gravity Field Models) in which the spherical harmonics coefficients sorted according to degrees.

degree, order, Cnm, Snm

Example			
2	0	-4.841694588430318E-04	+0.000000000000000E+00
2	1	-3.402167402706640E-10	+1.449802542797210E-09
2	2	+2.439360719619100E-06	-1.400308587714660E-06
3	0	+9.571897944720510E-07	+0.000000000000000E+00

3.2.2 Point data format

The first column must always be integer identifier; the follows geographic (geodetic) latitude, longitude (degrees) and height (meter), and data (gravity anomaly (mGal)) anomaly or disturbance.

Identifier, lat, lon, H, Δ_g or δ_g

Example				
1	45.511567	0.956149	305.315	13.517
2	44.952336	0.896234	102.672	-7.726
3	45.408971	2.628698	806.036	42.691
4	45.798034	0.152363	72.101	-10.249
5	45.967748	5.646873	981.344	41.784

3.2.3 Height anomaly data format

The height anomaly data file is similar to point data format, but the data column is not necessary. The first column must always be integer identifier; the follows geographic (geodetic) latitude, longitude (degrees), and height (meter).

Identifier, lat, lon, H

Example

1	46.212787	1.895712	398.157
2	46.742402	1.824359	357.468
3	46.226090	2.663530	162.504
4	45.718828	3.016851	546.312
5	46.791084	3.226739	889.905

3.2.4 DTM data format

Since SRBF.py uses the GRAVSOFT TC module, the DTM file must be compatible with the GRAVSOFT format. GRAVSOFT grid data are stored row-wise from north to south like you would read the values if they were printed in an atlas. The grid values are initiated with the label of latitude (φ) and longitude (λ) limits and spacing; the follows the data in free format:

φ_1, φ_2, λ_1, λ_2, $\Delta\varphi$, $\Delta\lambda$

H_{11} H_{12} H_{13} H_{14} . . . . H_{1m}

. . . .

. . . .

H_{n1} H_{n2} H_{n3} H_{n4} . . . . H_{nm}

Example

31.5	35	-108	-105	0.00833333	0.00833333	
2363.60	2355.78	2388.88	2341.38	2244.60	....	2352.52
2290.53	2234.55	2176.45	2058.27	2035.60	....	2011.33
2008.73	2004.10	2004.25	2003.23	2005.98	....	2052.42
2054.38	2067.73	2082.23	2074.82	2068.03	....	2036.77

"Each east-west row must be starting on a new line. Unknown data may be signaled by "9999". The grid label defines the exact latitude and longitude of the grid points, irrespectively whether the grids point values or average values over grid cells. The first data value in a grid file is thus the NW-corner (φ_2, λ_1) and the last the SE-corner (φ_1, λ_2)" (Forsberg and Tscherning, 2008).

4. Auvergne example

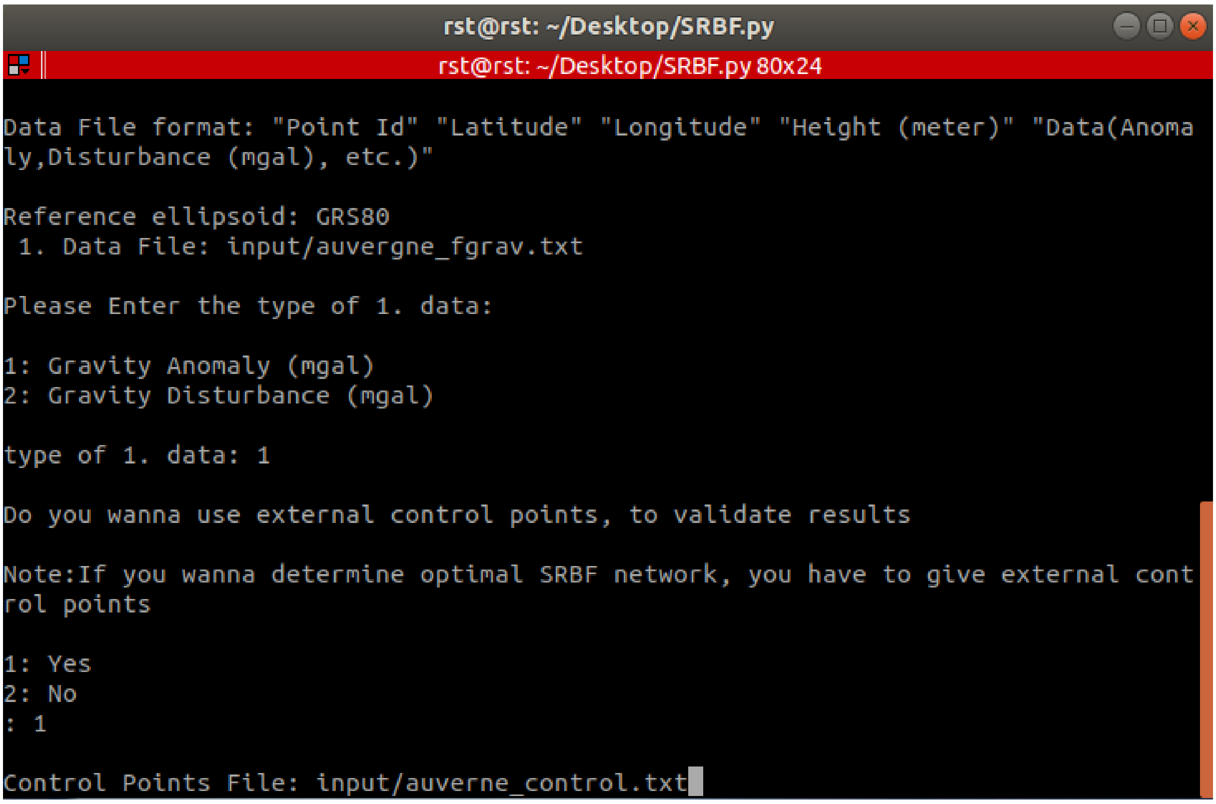
In the beginning, the user has to define the number of different data types which use in the process.

```
rst@rst: ~/Desktop/SRBF.py
rst@rst: ~/Desktop/SRBF.py 80x24
rst@rst:~/Desktop/SRBF.py$ python3 SRBF.py


Welcome Regional Gravity Field Modeling Software
In this version, you can only use gravity anomaly and disturbance
Please enter how many different type of data will use in modeling

Please enter how many different type of data you will use for modeling: 1
```

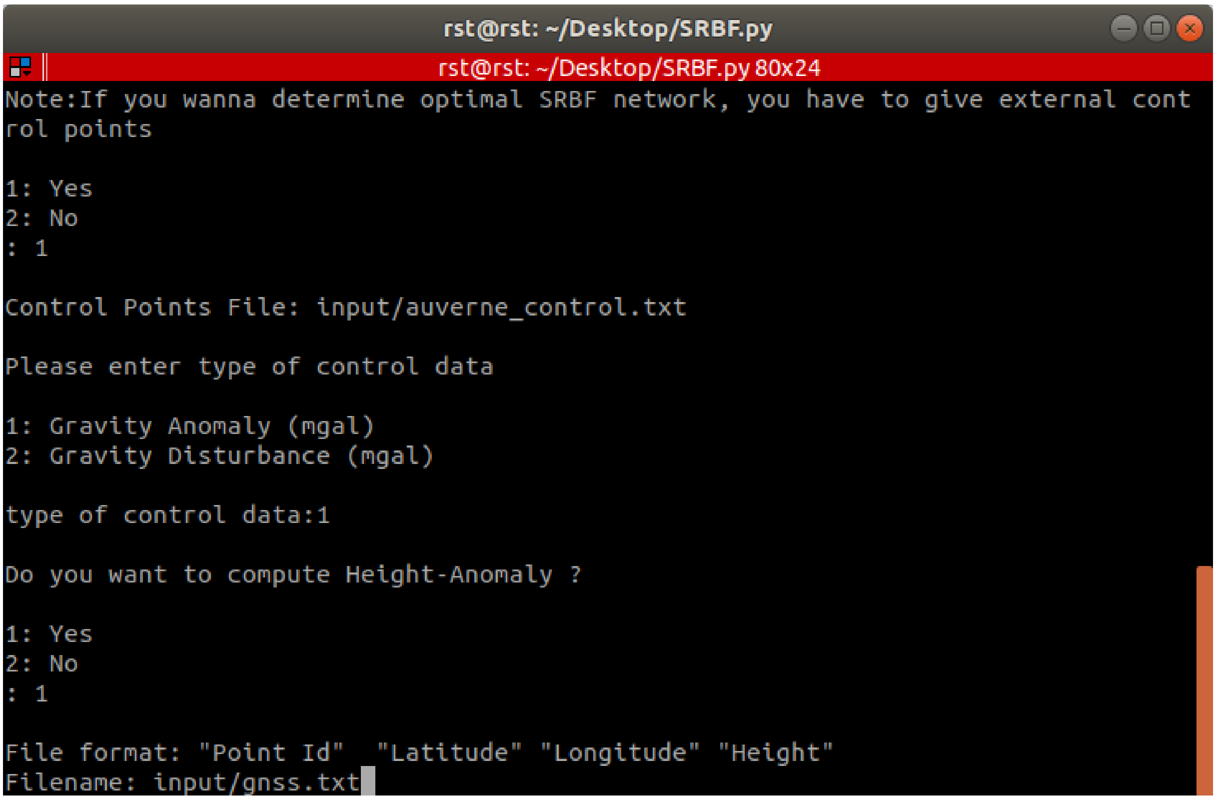
Then the software will ask filename and data type for the number of data

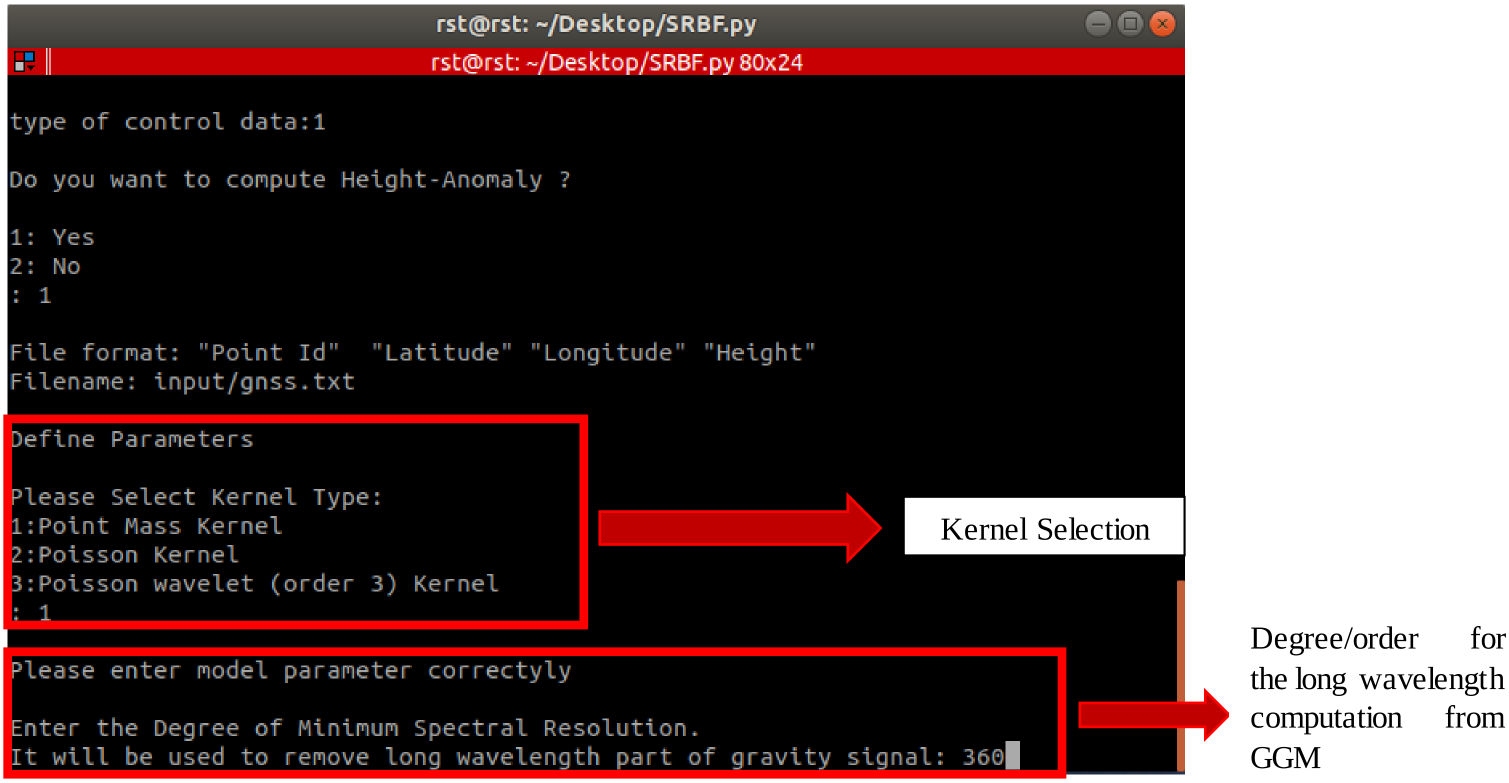
```
rst@rst: ~/Desktop/SRBF.py
rst@rst: ~/Desktop/SRBF.py 80x24
rst@rst:~/Desktop/SRBF.py$ python3 SRBF.py


Welcome Regional Gravity Field Modeling Software
In this version, you can only use gravity anomaly and disturbance
Please enter how many different type of data will use in modeling

Please enter how many different type of data you will use for modeling: 1

Please import 1. data file:

Data File format: "Point Id" "Latitude" "Longitude" "Height (meter)" "Data(Anoma
ly,Disturbance (mgal), etc.)"

Reference ellipsoid: GRS80
 1. Data File: input/auvergne_fgrav.txt

Please Enter the type of 1. data:

1: Gravity Anomaly (mgal)
2: Gravity Disturbance (mgal)

type of 1. data: 1
```

If you determine the optimal model parameter using root mean square (RMS) minimization approach, the control data points have to be added.

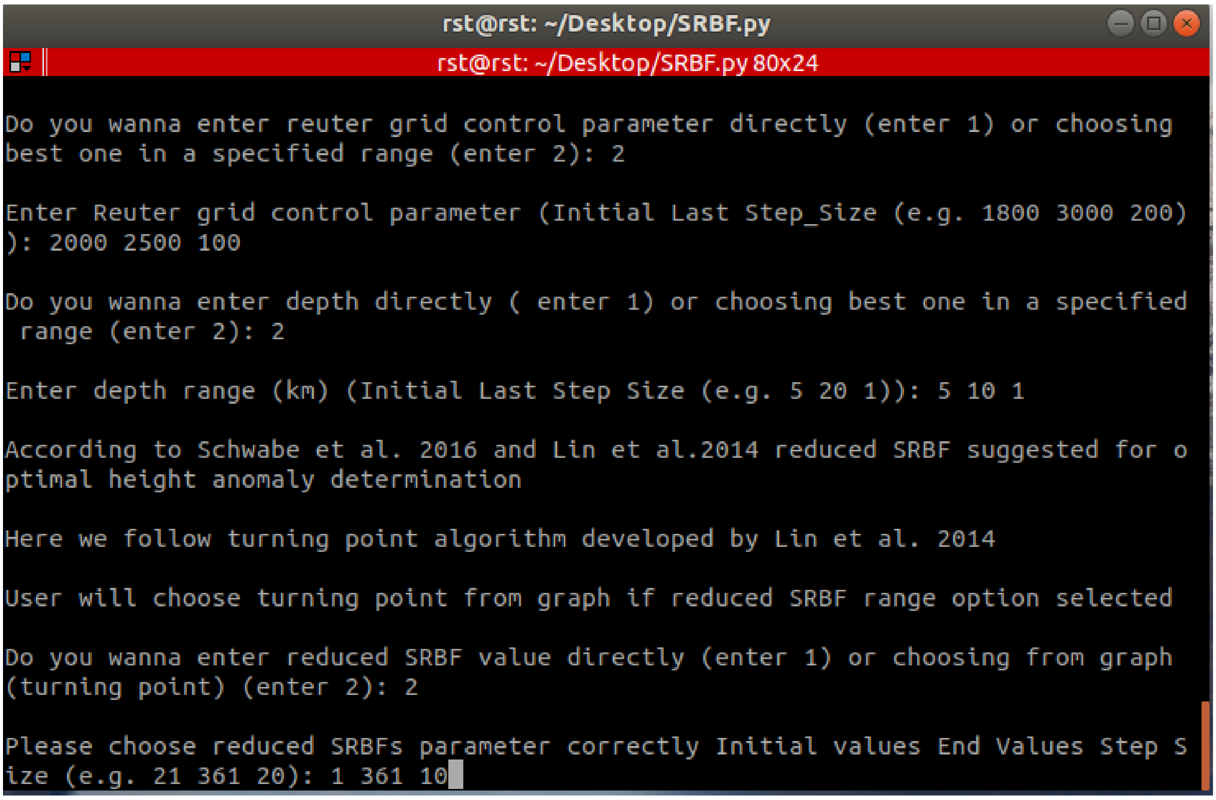


If you want to compute height anomaly in the interest area, you need to add height anomaly file.

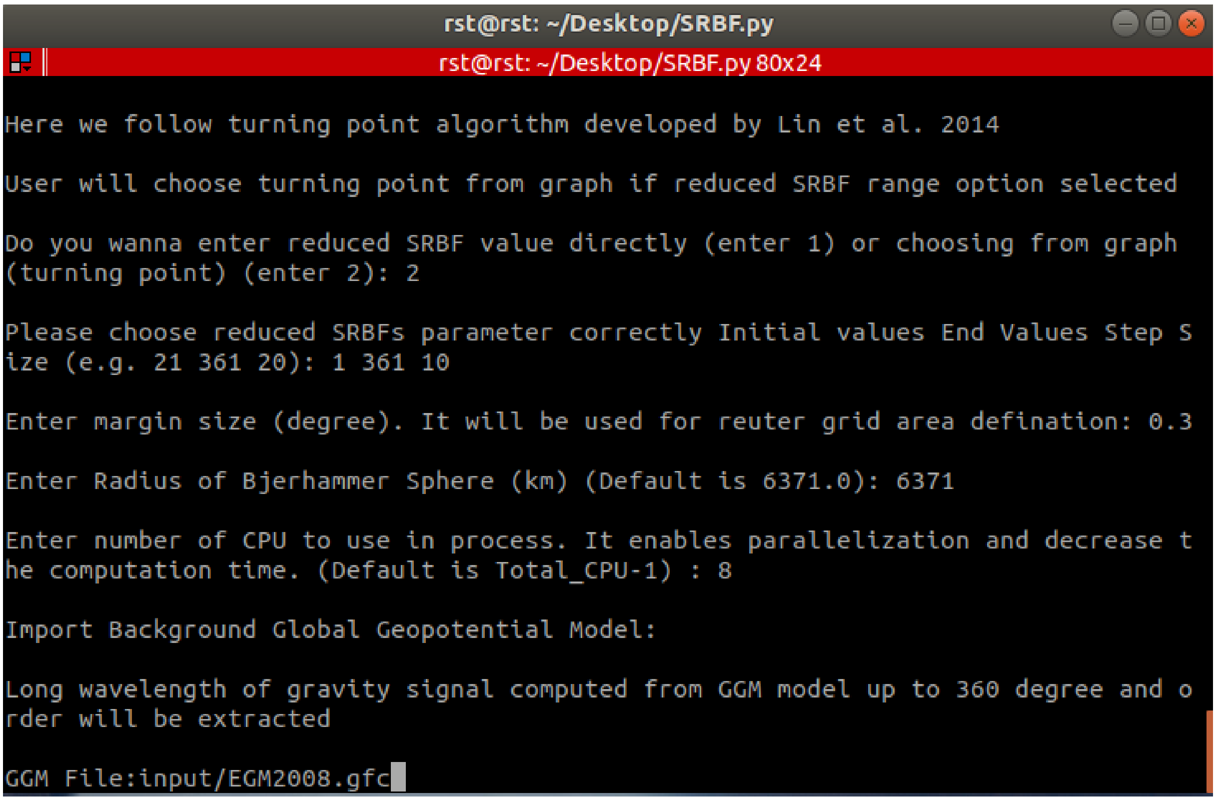




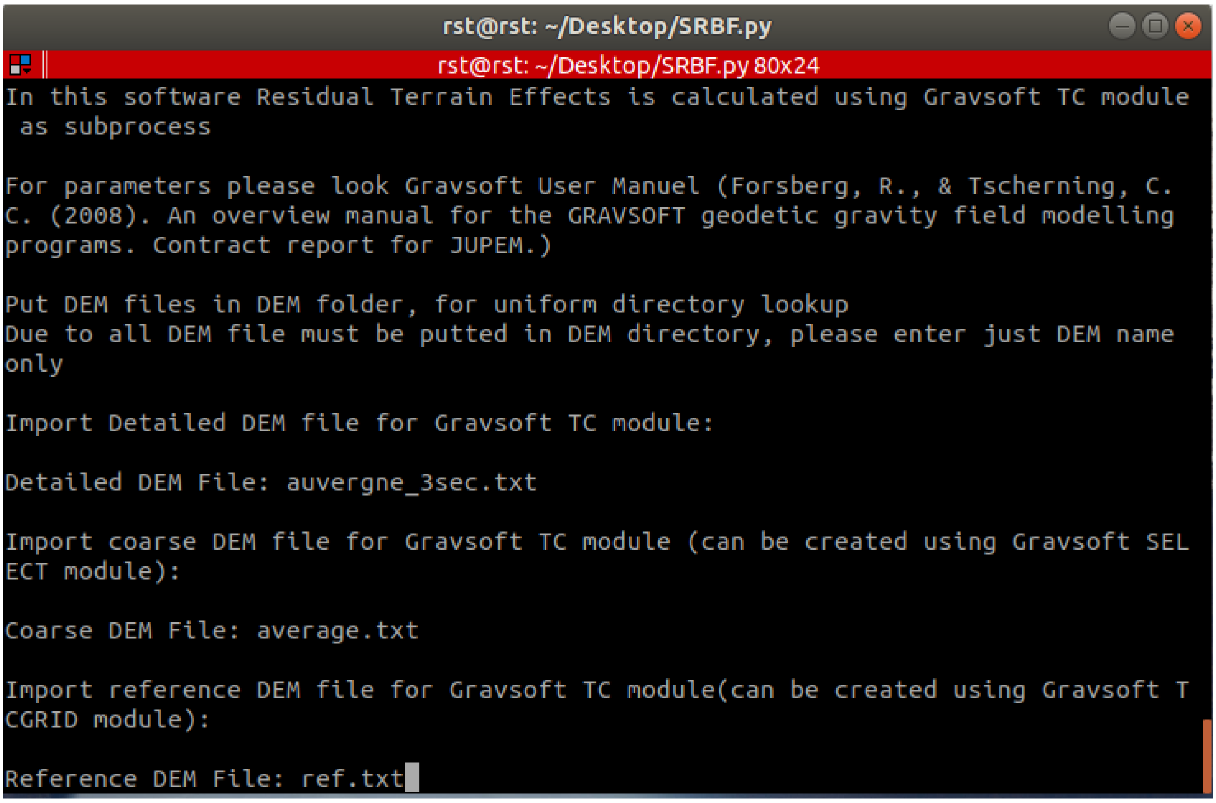
The Reuter grid control parameter, depth, and l_s should be an integer value. In SRBF.py, we use turning point approach for the selection of the l_s. For more detail, see (Lin et al., 2019)

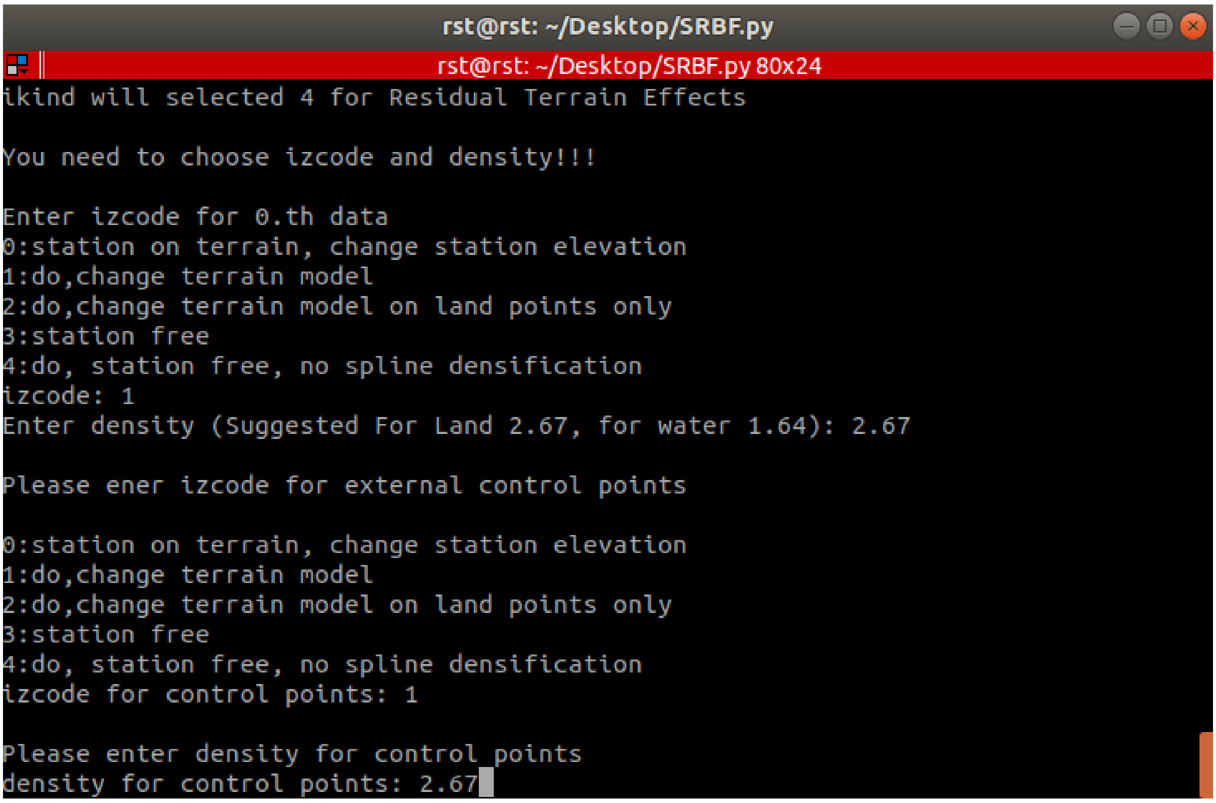


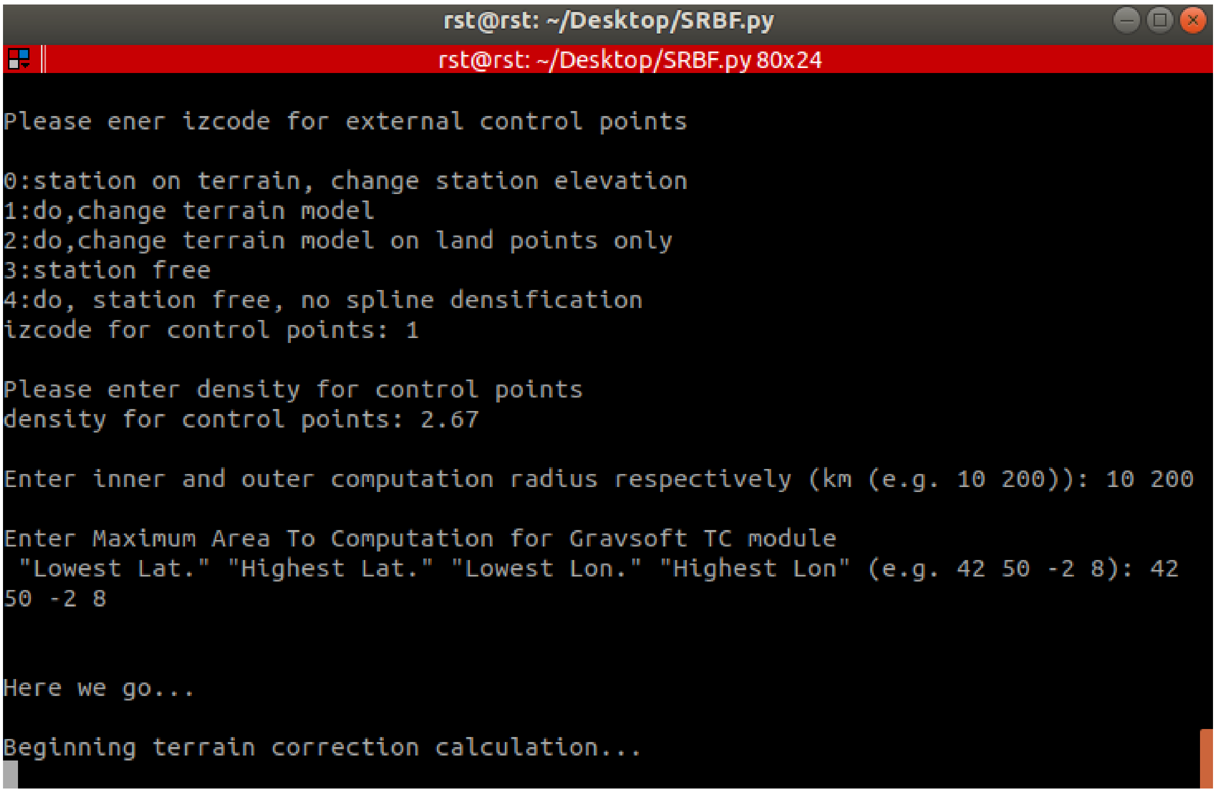
SRBF.py read Earth gravitation constant and radius of reference ellipsoid from GGM; however, somehow, this information might be missing. In such a case, the software will ask you to complete this information



The rest is related to the GRAVSOFT TC module.







References

Forsberg, R., 1984. A study of terrain reductions, density anomalies and geophysical inversion methods in gravity field modelling. Ohio State Univ Columbus Dept Of Geodetic Science and Surveying.

Forsberg, R., Tscherning, C.C., 2008. An overview manual for the GRAVSOFT geodetic gravity field modelling programs. Contract Rep. JUPEM.

Lin, M., Denker, H., Müller, J., 2019. A comparison of fixed-and free-positioned point mass methods for regional gravity field modeling. J. Geodyn. 125, 32–47.