Software for Fast Harmonic Synthesis
`hsynth_fast.f90`

February 2017

Pencho Petrushev, Kamen Ivanov, L. Scott Johnson

University of South Carolina

Contents

Introduction

The FORTRAN 90 code

`hsynth_fast.f90`

is developed for very fast and accurate evaluation of gravimetric quantities represented in high degree (> 2000) solid spherical or ellipsoidal harmonics at many scattered points in the space exterior to surface of the Earth. The covered gravimetric quantities are:

- Disturbing potential T (in m^2/s^2);
- Height anomaly ζ (in *meters*);
- Gravity anomaly Δg (in *mGal*);
- Gravity disturbance δg (in *mGal*);
- North-south deflection of the vertical ξ (in *arcseconds*);
- East-west deflection of the vertical η (in *arcseconds*);
- Second radial derivative of the disturbing potential T_{rr} (in *Eötvös Units*).

The values of these quantities are derived from the official NGA Earth Gravitational Model EGM2008 [3].

For the range from the surface of the Earth up to 544 kilometers above that surface the current version of the software runs on a standard PC at speed between 20,000 and 40,000 point evaluations per second, depending on the gravimetric quantity being evaluated, while the relative error does not exceed 10^{-6} and the memory (RAM) use is 9.3 GB.

It should be pointed out that the coefficients of the spherical harmonics expansions of T and ζ decrease at a moderate rate with the degree, while the spherical harmonics expansion of T_{rr} contains "large" high degree coefficients, which makes T_{rr} one of the most challenging quantity for fast evaluation. The gravity anomaly, gravity disturbance and deflection of the vertical coefficients exhibit intermediate behavior.

The algorithm for fast and accurate evaluation of gravimetric quantities presented here is based on the tensor product trigonometric needlets developed in [1]. A detailed description of this algorithm is given in Section 4 below.

It is a general requirement that the user run the software described in this manual on a computer with at least 16 GB RAM.

In the following we provide a detailed description our algorithm, software, and files with careful analysis of the accuracy, speed, and all related issues. The results of several experiments are also given.

1. Files in the package

The package hsynth_fast_f.zip contains all files necessary for the execution of the code hsynth_fast.f90. These are the files:

- hsynth_fast.f90
- hsynth_standard.f90
- hsynth_init.f90
- hsynth.bat
- EGM2008_to2190_TideFree
- scattered_points1000.zip, scattered_points1000000.zip
- scattered_points_values1000.zip
- Manual_hsynth_fast.pdf

The file hsynth_fast.f90 contains the FORTRAN code of the main program. In order to get executable files the user has to compile hsynth_init.f90 and hsynth_fast.f90 codes on his/her computer. For every gravimetric quantity using the EGM2008 coefficients given in EGM2008_to2190_TideFree the code hsynth_init.f90 produces a binary data file values_*.bin (up to 9.3 GB, not included in the package) with its weighted values at regular points located on several confocal ellipsoids.

The code hsynth_fast.f90 uses as input files one of the data files values_*.bin and the file scattered_points.dat, defined by the user. The package contains two test files scattered_points.dat with 1,000 and 1,000,000 point coordinates, respectively, contained in scattered_points1000.zip, scattered_points1000000.zip. The expected test results of the 1,000 point file are collected in scattered_points_values1000.zip.

The purpose of the code hsynth_standard.f90 is to monitor the accuracy of the performance and to be a benchmark for the speed of the code hsynth_fast.f90. The code hsynth_standard.f90 uses the file scattered_points_values.dat as an input file and evaluates the gravimetric quantities applying methods from the "standard" NGA software on the base of the model coefficients included in EGM2008_to2190_TideFree.

Manual_hsynth_fast.pdf is the current file.

2. Input and output files

2.1. **Free format ASCII input file.** The user defined ASCII input file `scattered_points.dat` contains the geocentric geographical spherical coordinates of the scattered points, where the gravimetric quantity is to be evaluated. The file has one record for every point. Each record contains the geocentric latitude and the longitude in decimal degrees followed by the geocentric distance in meters of the point.

(decimal degrees)	(decimal degrees)	(meters)
geocentric latitude	longitude	geocentric distance

These data are read using free FORMAT. Geocentric coordinates should refer to the WGS84. The scattered points have to be located in the range from -125 meters to $544,000$ meters above the Earth reference ellipsoid.

2.2. **Free format ASCII output files.** The file `scattered_points_values.dat` is an ASCII output file containing the gravimetric quantity values at the scattered points from the input. The file has one record for every point. Each record has 4 entries: the WGS84 geocentric geographical latitude and longitude in decimal degrees, the geocentric distance in meters, followed by the computed by `hsynth_fast.f90` value at this point. This file serves as input for `hsynth_standard.f90`.

The ASCII output file `scattered_points_values_standard.dat` has the same structure as `scattered_points_values.dat`, where the last entry of every record is the value computed by `hsynth_standard.f90` (instead of the value computed by `hsynth_fast.f90`).

2.3. **The EGM2008 coefficients input file.** `EGM2008_to2190_TideFree` is the standard EGM2008 coefficients file. It is required as input to `hsynth_init.f90` and `hsynth_standard.f90`. The file `EGM2008_to2190_TideFree` has been downloaded from

`http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm2008/EGM2008_to2190_TideFree.gz`.

2.4. **Binary data files.** The FORTRAN data files `values_*.bin` are not included in the package. They are produced by `hsynth_init.f90` in order to initialize `hsynth_fast.f90`. Every file contains several records – one record for every confocal ellipsoid used by the code (see §4.3). Every unformatted record contains double precision numbers with little-endian byte ordering. These numbers are the weighted values of the gravimetric quantity at regular points on the corresponding ellipsoid.

The binary data files are

- `values_t_e_38.bin` (9.2 GB, values of T);
- `values_zeta_e_38.bin` (9.2 GB, values of ζ);
- `values_ddg_e_53.bin` (9.0 GB, values of Δg).
- `values_dg_e_53.bin` (9.0 GB, values of δg).
- `values_xi_e_54.bin` (9.1 GB, values of ξ);
- `values_eta_e_55.bin` (9.3 GB, values of η);
- `values_trr_e_55.bin` (9.3 GB, values of T_{rr}).

Their names contain indication of the gravimetric quantity and the number of confocal ellipsoids with regular grid points.

3. How to run the software

To use the software described in this manual proceed as follows:

1. Compile the codes hsynth_init.f90, hsynth_fast.f90 and hsynth_standard.f90 using hsynth.bat.
2. When any of the three executables is run it prompts you to select from the keyboard the gravimetric quantity whose values you want to compute, indicated by *the case sensitive* options:
   - T for the disturbing potential;
   - zeta for the height anomaly;
   - Dg for the gravity anomaly;
   - dg for the gravity disturbance;
   - xi for the north-south deflection of the vertical;
   - eta for the east-west deflection of the vertical;
   - Trr for the second radial derivative of the disturbing potential.

   The code hsynth_standard.exe accepts the additional option eta1 (see §6).
3. If the binary data file values_*.bin corresponding to the selected gravimetric quantity does not exist, then generate it by running hsynth_init.exe.
4. Create a free format ASCII input file scattered_points.dat with the coordinates of the scattered points, where the values of the selected by you quantity are to be computed.
5. Run hsynth_fast.exe with the created by you file scattered_points.dat in order to obtain the ASCII output file scattered_points_values.dat with quantity values. You may repeat many times the previous and the current steps to evaluate the selected quantity on different sets of scattered points without performing step 3.
6. If you want to check the accuracy of hsynth_fast.exe, then run hsynth_standard.exe on scattered_points_values.dat, which is the output of hsynth_fast.exe.

Remarks:

- All data files are located in the executable files folder.
- Every data file values_*.bin is up to 9.3 GB.
- hsynth_init.exe works approximately 2 hours for xi or eta and 1 hour for the other quantities! [1]
- It is advisable to run hsynth_standard.exe on data file scattered_points_values.dat with no more than 10,000 points due to the low speed of the "standard" code (see the execution times as reported by the tests in §7.2).
- The time for selecting a gravimetric quantity is not included in the reported Total time.

[1] The execution times were measured on the computer configuration described in §7.

4. Key components of the algorithm of code `hsynth_fast.f90`

4.1. **Gravimetric quantities represented in solid spherical harmonics.** In this subsection we describe in detail the gravimetric quantities that the code `hsynth_init.f90` is designed to evaluate. All of them are represented as a product of a *slowly varying* function κ and a high degree *harmonic* expansion F (or a sum of such products).

4.1.1. *Disturbing potential.* The disturbing potential T is represented in spherical coordinates in EGM2008 by

$$T(r,\theta,\lambda) = \frac{GM}{a}\sum_{n=0}^{N}\left(\frac{a}{r}\right)^{n+1}\sum_{m=0}^{n}(\tilde{c}_{nm}\cos m\lambda + \tilde{s}_{nm}\sin m\lambda)\bar{P}_{nm}(\cos\theta),$$

where GM is the gravitational constant of the earth, a is the Earth semi-major axis, $N = 2190$, $\bar{P}_{nm}$ are the fully-normalized associated Legendre functions and $\tilde{c}_{nm}$, $\tilde{s}_{nm}$ are the fully-normalized spherical harmonic coefficients of T. The coefficients $\tilde{c}_{nm}$, $\tilde{s}_{nm}$ are stored in `EGM2008_to2190_TideFree` for $n \geq 2$ and $\tilde{c}_{nm} = \tilde{s}_{nm} = 0$ for $n = 0, 1$.

In this case the product representation takes the form

$$T(r,\theta,\lambda) = \kappa(r,\theta)F(r,\theta,\lambda), \quad 0 \leq \theta \leq \pi,\ 0 \leq \lambda < 2\pi,\ r > 0, \tag{1}$$

where

$$F(r,\theta,\lambda) = \sum_{n=0}^{N}\sum_{m=0}^{n}\left(\frac{a}{r}\right)^{n+1}(c_{nm}\cos m\lambda + s_{nm}\sin m\lambda)\bar{P}_{nm}(\cos\theta), \tag{2}$$

$$c_{nm} = \frac{GM}{a}\tilde{c}_{nm},\ s_{nm} = \frac{GM}{a}\tilde{s}_{nm}, \quad \kappa(r,\theta) = 1. \tag{3}$$

4.1.2. *Height anomaly.* The height anomaly $\zeta = T/\gamma$ with $\gamma = \gamma(r,\theta)$ being the normal gravity has a representation similar to (1)–(3) with ζ instead of T in (1) and $\kappa(r,\theta) = 1/\gamma(r,\theta)$ in (3).

4.1.3. *Gravity anomaly.* The gravity anomaly $\Delta g = -\dfrac{\partial T}{\partial r} - \dfrac{2}{r}T$ is represented just as in (1)–(2) with Δg instead of T in (1) and

$$c_{nm} = \frac{GM}{a^2}(n-1)\tilde{c}_{nm},\ s_{nm} = \frac{GM}{a^2}(n-1)\tilde{s}_{nm},\ \kappa(r,\theta) = \frac{a}{r}. \tag{4}$$

4.1.4. *Gravity disturbance.* The gravity disturbance $\delta g = -\dfrac{\partial T}{\partial r}$ is represented just as in (1)–(2) with δg instead of T in (1) and

$$c_{nm} = \frac{GM}{a^2}(n+1)\tilde{c}_{nm},\ s_{nm} = \frac{GM}{a^2}(n+1)\tilde{s}_{nm},\ \kappa(r,\theta) = \frac{a}{r}. \tag{5}$$

4.1.5. *Second radial derivative of the disturbing potential.* The second radial derivative of the disturbing potential $T_{rr} = \dfrac{\partial^2 T}{\partial r^2}$ is represented just as in (1)–(2) with T_{rr} instead of T in (1) and

$$c_{nm} = \frac{GM}{a^3}(n+1)(n+2)\tilde{c}_{nm},\ s_{nm} = \frac{GM}{a^3}(n+1)(n+2)\tilde{s}_{nm},\ \kappa(r,\theta) = \frac{a^2}{r^2}. \tag{6}$$

4.1.6. *North-south deflection of the vertical.* The spherically approximated north-south deflection of the vertical $\xi = \dfrac{1}{r\gamma}\dfrac{\partial T}{\partial \theta}$ has the representation

$$\xi(r,\theta,\lambda) = \kappa(r,\theta)\left(F_1(r,\theta,\lambda)\cos\lambda + F_2(r,\theta,\lambda)\sin\lambda\right), \tag{7}$$

where

$$F_j(r,\theta,\lambda) = \sum_{n=2}^{N}\sum_{m=0}^{n}\left(\frac{a}{r}\right)^{n+1}\frac{GM}{a^2}(c^j_{nm}\cos m\lambda + s^j_{nm}\sin m\lambda)\bar{P}_{nm}(\cos\theta) \tag{8}$$

for $j = 1, 2$ and

$$\kappa(r,\theta) = \frac{a}{r\gamma(r,\theta)}. \tag{9}$$

To obtain the coefficients c^j_{nm}, s^j_{nm}, $j = 1, 2$, we use the derivative representations

$$\begin{aligned}\sqrt{1-x^2}\frac{d\bar{P}_{nm}(x)}{dx} &= \frac{\sqrt{(n-m)(n+m+1)}}{2}\bar{P}_{n,m+1}(x)\\ &\quad - \frac{\sqrt{(1+\delta_{0,m-1})(n+m)(n-m+1)}}{2}\bar{P}_{n,m-1}(x),\ \ m \geq 1;\end{aligned}$$

$$\sqrt{1-x^2}\frac{d\bar{P}_{n0}(x)}{dx} = \sqrt{\frac{n(n+1)}{2}}\bar{P}_{n,1}(x), \quad n \in \mathbb{N}, \tag{10}$$

where $\delta_{k,\ell}$ denotes the Kroneker symbol. These representations along with the standard identities

$$\cos m\lambda = \cos(m\pm 1)\lambda\cos\lambda \pm \sin(m\pm 1)\lambda\sin\lambda, \quad m \in \mathbb{N}\cup\{0\}; \tag{11}$$

$$\sin m\lambda = \sin(m\pm 1)\lambda\cos\lambda \mp \cos(m\pm 1)\lambda\sin\lambda, \quad m \in \mathbb{N}, \tag{12}$$

lead to

$$c^1_{n,m} = \begin{cases} \frac{1}{\sqrt{2}}\sqrt{n(n+1)}\tilde{c}_{n,1}, & \text{for } m = 0;\\ \frac{1}{2}\sqrt{(n-1)(n+2)}\tilde{c}_{n,2} - \frac{1}{\sqrt{2}}\sqrt{n(n+1)}\tilde{c}_{n,0}, & \text{for } m = 1;\\ \frac{1}{2}\sqrt{(n-m)(n+m+1)}\tilde{c}_{n,m+1} & \\ \quad -\frac{1}{2}\sqrt{(n-m+1)(n+m)}\tilde{c}_{n,m-1}, & \text{for } 2 \leq m \leq n; \end{cases} \tag{13}$$

$$s^1_{n,m} = \begin{cases} 0, & \text{for } m = 0;\\ \frac{1}{2}\sqrt{(n-1)(n+2)}\tilde{s}_{n,2}, & \text{for } m = 1;\\ \frac{1}{2}\sqrt{(n-m)(n+m+1)}\tilde{s}_{n,m+1} & \\ \quad -\frac{1}{2}\sqrt{(n-m+1)(n+m)}\tilde{s}_{n,m-1}, & \text{for } 2 \leq m \leq n; \end{cases} \tag{14}$$

$$c^2_{n,m} = \begin{cases} \frac{1}{\sqrt{2}}\sqrt{n(n+1)}\tilde{s}_{n,1}, & \text{for } m = 0;\\ \frac{1}{2}\sqrt{(n-1)(n+2)}\tilde{s}_{n,2}, & \text{for } m = 1;\\ \frac{1}{2}\sqrt{(n-m)(n+m+1)}\tilde{s}_{n,m+1} & \\ \quad +\frac{1}{2}\sqrt{(n-m+1)(n+m)}\tilde{s}_{n,m-1}, & \text{for } 2 \leq m \leq n; \end{cases} \tag{15}$$

$$s^2_{n,m} = \begin{cases} 0, & \text{for } m = 0;\\ -\frac{1}{2}\sqrt{(n-1)(n+2)}\tilde{c}_{n,2} - \frac{1}{\sqrt{2}}\sqrt{n(n+1)}\tilde{c}_{n,0}, & \text{for } m = 1;\\ -\frac{1}{2}\sqrt{(n-m)(n+m+1)}\tilde{c}_{n,m+1} & \\ \quad -\frac{1}{2}\sqrt{(n-m+1)(n+m)}\tilde{c}_{n,m-1}, & \text{for } 2 \leq m \leq n. \end{cases} \tag{16}$$

The NGA codes utilize the derivative representation

$$(17)\quad (1-x^2)\frac{d\bar{P}_{nm}(x)}{dx} = \sqrt{\frac{(n^2-m^2)(2n+1)}{2n-1}}\bar{P}_{n-1,m}(x) - nx\bar{P}_{nm}(x), \quad n\in\mathbb{N},$$

which implies a representation of the deflection of the vertical in the form

$$(18)\qquad \xi(r,\theta,\lambda) = \kappa(r,\theta)\frac{rF_1^\star(r,\theta,\lambda)+\cos\theta F_2^\star(r,\theta,\lambda)}{\sin\theta},$$

where $F_1^\star, F_2^\star$ are certain harmonic expansions.

Observe that the singularity of ξ in representation (7) is contained only in the two multipliers $\cos\lambda$ and $\sin\lambda$, that is, (7) has better numerical stability near the poles than (18), which has a $\sin\theta$ term in the denominator. More precisely, our algorithm produces an approximation $\tilde{\xi}$ to ξ, which obeys the same error all over the ellipsoid as shown in Subsection 4.6. When the evaluation point coincides with one of the poles (where ξ has a bounded discontinuity) then $\tilde{\xi}$ approximates the latitude limit of ξ with the same precision as at the other points, i.e.

$$\begin{aligned}\tilde{\xi}(r,0,\lambda) &\approx \lim_{\theta\to 0}\xi(r,\theta,\lambda)\\ &= \frac{GM}{\sqrt{2}ar\gamma(r,0)}\sum_{n=2}^{N}\left(\frac{a}{r}\right)^{n+1}\sqrt{n(n+1)(2n+1)}[\tilde{c}_{n,1}\cos\lambda+\tilde{s}_{n,1}\sin\lambda],\end{aligned}$$

$$\begin{aligned}\tilde{\xi}(r,\pi,\lambda) &\approx \lim_{\theta\to \pi}\xi(r,\theta,\lambda)\\ &= -\frac{GM}{\sqrt{2}ar\gamma(r,\pi)}\sum_{n=2}^{N}\left(-\frac{a}{r}\right)^{n+1}\sqrt{n(n+1)(2n+1)}[\tilde{c}_{n,1}\cos\lambda+\tilde{s}_{n,1}\sin\lambda],\end{aligned}$$

as the values of the limits follow from (7), (8), (9), (13), (15) with $m=0$.

4.1.7. *East-west deflection of the vertical.* The spherically approximated east-west deflection of the vertical

$$\begin{aligned}(19)\quad \eta(r,\theta,\lambda) &= -\frac{1}{r\gamma(r,\theta)\sin\theta}\frac{\partial T}{\partial\lambda}(r,\theta,\lambda)\\ &= \frac{GM}{ar\gamma(r,\theta)}\sum_{n=2}^{N}\sum_{m=1}^{n}\left(\frac{a}{r}\right)^{n+1} m\frac{\bar{P}_{nm}(\cos\theta)}{\sin\theta}[\tilde{c}_{n,m}\sin m\lambda-\tilde{s}_{n,m}\cos m\lambda]\end{aligned}$$

has the representation

$$(20)\qquad \eta(r,\theta,\lambda) = \kappa(r,\theta)\left(F_1(r,\theta,\lambda)\cos\lambda+F_2(r,\theta,\lambda)\sin\lambda\right)$$

with

$$(21)\qquad \kappa(r,\theta) = \frac{a^2}{r^2\gamma(r,\theta)}.$$

The functions F_j, $j=1,2$, are harmonic and have the representation

$$\begin{aligned}(22)\quad & F_j(r,\theta,\lambda)\\ &= \sum_{n=1}^{N-1}\sum_{m=0}^{n}\left(\frac{a}{r}\right)^{n+1}\frac{GM}{2a^2}\sqrt{\frac{2n+3}{2n+1}}(c_{nm}^j\cos m\lambda+s_{nm}^j\sin m\lambda)\bar{P}_{nm}(\cos\theta),\end{aligned}$$

where coefficients $c^j_{n,m}$, $s^j_{n,m}$, $j = 1, 2$, are defined by

$$(23)\qquad c^1_{n,m} = \begin{cases} -\sqrt{2(n+1)(n+2)}\tilde{s}_{n+1,1}, & \text{for } m = 0; \\ -\sqrt{(n+2)(n+3)}\tilde{s}_{n+1,2}, & \text{for } m = 1; \\ -\sqrt{(n+m+1)(n+m+2)}\tilde{s}_{n+1,m+1} & \\ -\sqrt{(n-m+1)(n-m+2)}\tilde{s}_{n+1,m-1}, & \text{for } 2 \le m \le n; \end{cases}$$

$$(24)\qquad s^1_{n,m} = \begin{cases} 0, & \text{for } m = 0; \\ \sqrt{(n+2)(n+3)}\tilde{c}_{n+1,2}, & \text{for } m = 1; \\ \sqrt{(n+m+1)(n+m+2)}\tilde{c}_{n+1,m+1} & \\ +\sqrt{(n-m+1)(n-m+2)}\tilde{c}_{n+1,m-1}, & \text{for } 2 \le m \le n; \end{cases}$$

$$(25)\qquad c^2_{n,m} = \begin{cases} \sqrt{2(n+1)(n+2)}\tilde{c}_{n+1,1}, & \text{for } m = 0; \\ \sqrt{(n+2)(n+3)}\tilde{c}_{n+1,2}, & \text{for } m = 1; \\ \sqrt{(n+m+1)(n+m+2)}\tilde{c}_{n+1,m+1} & \\ -\sqrt{(n-m+1)(n-m+2)}\tilde{c}_{n+1,m-1}, & \text{for } 2 \le m \le n; \end{cases}$$

$$(26)\qquad s^2_{n,m} = \begin{cases} 0, & \text{for } m = 0; \\ \sqrt{(n+2)(n+3)}\tilde{s}_{n+1,2}, & \text{for } m = 1; \\ \sqrt{(n+m+1)(n+m+2)}\tilde{s}_{n+1,m+1} & \\ -\sqrt{(n-m+1)(n-m+2)}\tilde{s}_{n+1,m-1}, & \text{for } 2 \le m \le n. \end{cases}$$

Representation (20)–(26) is obtained by replacing

$$(27)\quad m\frac{\bar{P}_{nm}(x)}{\sqrt{1-x^2}} = \frac{1}{2}\sqrt{\frac{2n+1}{2n-1}}\sqrt{(n-m-1)(n-m)}\bar{P}_{n-1,m+1}(x)$$
$$+\frac{1}{2}\sqrt{\frac{2n+1}{2n-1}}\sqrt{(1+\delta_{0,m-1})(n+m-1)(n+m)}\bar{P}_{n-1,m-1}(x), \quad 1 \le m \le n.$$

with $x = \cos\theta$ in (20) and applying identities (11)–(12).

Representation (20) shows that η has bounded discontinuities at the poles. The main disadvantage of representation (19) (compared to (20)) is the $\sin\theta$ term in the denominator, which generates computational instability around the poles. Our code gives an approximation $\tilde{\eta}$ to η, which obeys the same error bound all over the ellipsoid as shown in Subsection 4.6. When the evaluation point coincides with one of the poles (where η has a bounded discontinuity) then $\tilde{\eta}$ approximates the latitude limit of η with the same precision, i.e.

$$\tilde{\eta}(r,0,\lambda) \approx \lim_{\theta\to 0}\eta(r,\theta,\lambda)$$
$$= \frac{GM}{\sqrt{2}ar\gamma(r,0)}\sum_{n=2}^{N}\left(\frac{a}{r}\right)^{n+1}\sqrt{n(n+1)(2n+1)}[\tilde{c}_{n,1}\sin\lambda - \tilde{s}_{n,1}\cos\lambda],$$

$$\tilde{\eta}(r,\pi,\lambda) \approx \lim_{\theta\to \pi}\eta(r,\theta,\lambda)$$
$$= \frac{GM}{\sqrt{2}ar\gamma(r,\pi)}\sum_{n=2}^{N}\left(-\frac{a}{r}\right)^{n+1}\sqrt{n(n+1)(2n+1)}[\tilde{c}_{n,1}\sin\lambda - \tilde{s}_{n,1}\cos\lambda],$$

as the values of the limits follow from (20), (21), (22), (23), (25) with $m = 0$.

4.2. **From spherical harmonics to ellipsoidal harmonics.** Our next step is to represent the harmonic function $F(r,\theta,\lambda)$ (or F_1, F_2) in ellipsoidal coordinates (u,φ,λ), where φ is the complement of the reduced latitude and u is the semi-minor axis of the confocal ellipsoid. These coordinates are related to the spherical coordinates (r,θ,λ) by

$$\begin{cases} r\sin\theta\cos\lambda &= \sqrt{u^2+E^2}\sin\varphi\cos\lambda, \\ r\sin\theta\sin\lambda &= \sqrt{u^2+E^2}\sin\varphi\sin\lambda, \\ r\cos\theta &= u\cos\varphi. \end{cases}$$

Applying Jekeli's transformation [2] we transform the coefficients

$$c_{nm}, s_{nm}, 0\le m\le n, 2\le n\le N, \quad \text{to} \quad c_{n,m}^{\{ell\}}, s_{n,m}^{\{ell\}}, 0\le m\le n, 2\le n\le N_1,$$

so that

$$F(r,\theta,\lambda) = H(u,\varphi,\lambda). \tag{28}$$

The harmonic expansion of $H(u,\varphi,\lambda)$ takes the form

$$H(u,\varphi,\lambda) = \sum_{n=2}^{N_1}\sum_{m=0}^{n} \frac{\bar{S}_{n,m}\left(\frac{u}{E}\right)}{\bar{S}_{n,m}\left(\frac{b}{E}\right)} \left(c_{nm}^{\{ell\}}\cos m\lambda + s_{nm}^{\{ell\}}\sin m\lambda\right)\bar{P}_{nm}(\cos\varphi), \tag{29}$$

where $\bar{S}_{n,m}$ are Jekeli's functions and b is the Earth semi-minor axis. In theory $N_1=\infty$ but in practice $N_1 = N+70$ gives (28) with relative error less than 10^{-20}.

For every fixed u the function $H(u,\varphi,\lambda)$ (or $H_1(u,\varphi,\lambda)\cos\lambda + H_2(u,\varphi,\lambda)\sin\lambda$) is a bi-variate trigonometric polynomial of degree N_1 (or N_1+1) and tensor product needlets can be utilized for its fast evaluation (see Subsections 4.4-4.5). The reason for switching from spherical harmonic expansions to ellipsoidal harmonic expansions is to guarantee smaller approximation error as explained in Subsection 4.6.3.

4.3. **Evaluation of gravimetric quantities in an ellipsoidal shell.** The code `hsynth_fast.f90` computes an approximation $\tilde{G}(u,\varphi,\lambda)$ to the gravimetric quantity $G(u,\varphi,\lambda)$, $G = T,\zeta,\Delta g,\delta g,\xi,\eta,T_{rr}$, for u from the ellipsoidal shell

$$U_0\le u\le U_1 \quad \text{with } U_0 = b-125\ m \text{ and } U_1 = b+544,000\ m.$$

This approximation is obtained by interpolating the values of G on several confocal ellipsoids. For a fixed $(\varphi,\lambda)\in\mathbb{S}^2$ if $\tilde{G}(u,\varphi,\lambda)$ is the Lagrange interpolant of $G(u,\varphi,\lambda)$ at the points $u_j = u_1+(j-1)h$, $j=1,2,\dots,2J$, then the error takes the form

$$\tilde{G}(u,\varphi,\lambda) - G(u,\varphi,\lambda) = \frac{(u-u_1)\dots(u-u_{2J})}{(2J)!}\frac{\partial^{2J}G}{\partial u^{2J}}(z,\varphi,\lambda)$$

for some $z\in(u_1,u_{2J})$. The best choice for u is $u\in[u_J,u_{J+1}]$, where the product $(u-u_1)\dots(u-u_{2J})$ has a ch^{2J} bound with the smallest constant c. However, the $2J$-th derivative of G with respect to u grows very rapidly as u approaches b (i.e. the Earth surface), which leads to a big error!

In order to reduce the influence of the derivative term on the error we take an increasing function μ defined on $[0,s^*]$ such that $\mu(0)=U_0$, $\mu'(0)=0$, $\mu(\bar{s})=U_1$ for some $0<\bar{s}<s^*$ and set

$$g(s,\varphi,\lambda) = G(\mu(s),\varphi,\lambda), \quad (\varphi,\lambda)\in\mathbb{S}^2,\ 0\le s\le s^*.$$

Possible choices for μ are

$$\mu(s)=U_0+\frac{U_1-U_0}{1-\cos\bar{s}}(1-\cos s);\;\mu(s)=U_0+\frac{U_1-U_0}{\bar{s}^2}s^2;\;\mu(s)=U_0+\frac{U_1-U_0}{\bar{s}^4}s^4;$$

as the last one is used in the codes `hsynth_init` and `hsynth_fast`. They give us functions g with *essentially smaller oscillation* of the "radial" derivatives than those of G and at the same time these derivatives of g can be explicitly expressed in terms of the "radial" derivatives of G.

In the code we use $2J$ point Lagrange interpolation for G, where $J=3$ for $G=T,\zeta$; $J=4$ for $G=\Delta g,\delta g,\xi,\eta$ and $J=5$ for $G=T_{rr}$. Thus setting $s_j=jh$, $j=-J+1,-J+2,\dots,M$ with $\mu(s_{M-J-1})\le U_1\le\mu(s_{M-J})$ we pre-compute the values of $G(v_j,\varphi_k,\lambda_\ell)$ at regular points $(\varphi_k,\lambda_\ell)\in\mathbb{S}^2$ (see §4.4) on confocal ellipsoids of semi-minor axis $v_j=\mu(s_j)$. Then for the computation of $\tilde{G}(u,\varphi,\lambda)$ we utilize this algorithm:

(1) For $u\in[U_0,U_1]$ find $s=\mu^{-1}(u)$ and $j\ge 0$ such that $s\in[s_j,s_{j+1}]$, where μ^{-1} denotes the inverse function of μ;
(2) Use tensor product needlets (see §4.5) to compute $\tilde{g}(s_i,\varphi,\lambda):=\tilde{G}(v_i,\varphi,\lambda)$, $i=j-J+1,j-J+2,\dots,j+J$ from the values of G at the regular points from each of these $2J$ ellipsoids;
(3) Use Lagrange interpolation to compute $\tilde{G}(u,\varphi,\lambda)\;:=\;\tilde{g}(s,\varphi,\lambda)$ from $\tilde{g}(s_i,\varphi,\lambda)$, $i=j-J+1,j-J+2,\dots,j+J$.

Note that the choice of μ as an even function gives $v_{-1}=v_1$, $v_{-2}=v_2$, etc., which allows to only use values of $\tilde{G}(v_j,\varphi,\lambda)$ at points with $v_j\ge U_0$ in the computation of $\tilde{G}(u,\varphi,\lambda)$! Thus we avoid derivatives of G at points with coordinate u smaller than U_0. The choice $U_0=b-125$ m allows to include in the ellipsoidal shell all points *above the geoid*.

Note that the above computational scheme represents for fixed φ,λ an interpolation of $G(u,\varphi,\lambda)$ at the non-equally spaced points v_j by a $2J-1$ degree polynomial of $\mu^{-1}(u)$.

The binary file `values_*.bin` contains the values of $G(v_j,\varphi_k,\lambda_\ell)\dfrac{2}{KL}$, $j=0,1,\dots,M$, $k=0,1,\dots,K$, $\ell=0,1,\dots,L-1$. For K and L see Subsection 4.4 and (30).

4.4. **Regular grid.** The regular grids on the sphere are given in spherical coordinates by

$$\mathcal{X}=\{x_{k,\ell}=(\theta_k,\lambda_\ell)\;:\;k=0,1,\dots,K,\;\ell=0,1,\dots,L-1\},$$

where

$$\theta_k=\frac{\pi k}{K},\;k=0,1,\dots,K;\qquad\lambda_\ell=\frac{2\pi\ell}{L},\;\ell=0,1,\dots,L-1.$$

Here L *must be even* in order to allow the values of the same grid to be used for continuation through the poles. In the applications we usually takes $L=2K$.

The regular grids on an ellipsoid are defined by the same formulas with θ replaced by φ, where (φ,λ) denotes the ellipsoidal coordinates. Note, that for fixed θ and λ the set $\{(r,\theta,\lambda):r>0\}$ is a half-line in spherical coordinates, while for fixed φ and λ the set $\{(u,\varphi,\lambda):u>0\}$ is part of a hyperbola in ellipsoidal coordinates. This hyperbola reduces to a half-line only in the partial cases $\varphi=0,\pi/2,\pi$.

4.5. **Evaluation of bi-variate trigonometric polynomials.** As shown in §4.1–4.2 the restrictions of any of the quantities $f = G/\kappa$, $G = T, \zeta, \Delta g, \delta g, \xi, \eta, T_{rr}$, on any ellipsoid confocal with the Earth reference ellipsoid is a bi-variate trigonometric polynomial in φ and λ of degree at most $N_1 + 1$. Similarly, the restrictions of f on any sphere centred at the centre of the Earth is a bi-variate trigonometric polynomial in θ and λ of degree at most N.

The bi-variate trigonometric polynomial $f(\varphi, \lambda)$ is evaluated at a point $(\varphi, \lambda) \in \mathbb{S}^2$ by tensor product trigonometric needlets of the form

$$\tilde{f}(\varphi, \lambda) = \sum_{|\varphi-\varphi_k|\le\delta_1} \sum_{|\lambda-\lambda_\ell|\le\delta_2} \mathcal{K}_1(\varphi - \varphi_k)\mathcal{K}_2(\lambda - \lambda_\ell)\frac{2}{KL} f(\varphi_k, \lambda_\ell), \tag{30}$$

where $\mathcal{K}_1$ and $\mathcal{K}_2$ are trigonometric needlet kernels in φ and λ, respectively.

The number of knots $\{\varphi_k\}$ on the interval $[\varphi - \delta_1, \varphi + \delta_1]$ and the number of knots $\{\lambda_\ell\}$ on the interval $[\lambda - \delta_2, \lambda + \delta_2]$ ranges from 26 to 32 (depending on f) for targeted accuracy `ParKer(3)`$= 2 \times 10^{-7}$.

In (30) for λ close to 0 or to 2π we assume that the definition of λ_ℓ from §4.4 is extended by the same formula for $\ell < 0$ or $\ell \ge L$, which implies the *periodic* extension $f(\varphi, \lambda + 2\pi) = f(\varphi, \lambda)$ of f. Similarly, for θ close to 0 or to π we extend the definition of θ_k from §4.4 by the same formula for $k < 0$ or $k > K$, which implies the *even semi-periodic* extension $f(-\varphi, \lambda + \pi) = f(\varphi, \lambda)$ in the case of $T, \zeta, \Delta g, \delta g$ or T_{rr}, and the *odd semi-periodic* extension $f(-\varphi, \lambda + \pi) = -f(\varphi, \lambda)$ in the case of ξ or η. These extensions do not require evaluation of the polynomial f at new points $(\varphi_k, \lambda_\ell)$ whenever L is even!

We refer the reader to [1] for more details on tensor product trigonometric needlets.

4.6. **Accuracy.**

4.6.1. *Relative error.* For a gravimetric quantity G, G being T, ζ, Δg, δg, ξ, η or T_{rr}, denote by

$$\mathcal{N}(G, \mathbb{E}, u) = \max_{(\varphi,\lambda)\in\mathbb{S}^2} |G(u, \varphi, \lambda)|$$

its norm on an ellipsoid with semi-minor axis u confocal to the Earth reference ellipsoid. Here u, φ, λ stand for the ellipsoidal coordinates.

For a point P with spherical coordinates $(r_P, \theta_P, \lambda_P)$ and ellipsoidal coordinates $(u_P, \varphi_P, \lambda_P)$, $U_0 \le u_P \le U_1$, $(\varphi_P, \lambda_P) \in \mathbb{S}^2$, the code `hsynth_fast.f90` gives an approximation $\tilde{G}(u_P, \varphi_P, \lambda_P)$ to the gravimetric quantity $G(u_P, \varphi_P, \lambda_P)$ with relative error 10^{-6}, i.e.

$$\max_{\substack{(\varphi_P,\lambda_P)\in\mathbb{S}^2,\\ U_0\le u_P\le U_1}} \frac{|\tilde{G}(u_P, \varphi_P, \lambda_P) - G(u_P, \varphi_P, \lambda_P)|}{\mathcal{N}(G, \mathbb{E}, u_P)} \le 10^{-6}. \tag{31}$$

Note that (31) holds even when G denotes ξ or η, which are bounded discontinuous functions at the poles.

The relative precision 10^{-6} in (31) is selected to be better than the relative accuracy of the model EGM2008 itself. When the relative accuracy of the gravimetric model is improved then our codes can be easily modified to work with higher accuracy.

Table 1 illustrates the dependance of the norms $\mathcal{N}(G, \mathbb{E}, u)$ on the ellipsoid semiminor axis. Notice the modest decrease of the norms of T and ζ, the faster decrease of the norms of Δg, δg, ξ and η and the very fast decrease of the norms of T_{rr}. The decreasing of $\mathcal{N}(G, \mathbb{E}, u_P)$ results in smaller absolute error in (31) when u_P increases.

The entries in Table 1 are obtained by subroutine `norms_e` in `hsynth_standard.f90`. This subroutine can be used for determining the norms for other values of u_P as well.

Height h	*km*	0	100	200	300	400	500
$\mathcal{N}(T, \mathbb{E}, b+h)$	m^2/s^2	1041.54	945.64	873.39	811.31	755.93	706.09
$\mathcal{N}(\zeta, \mathbb{E}, b+h)$	m	106.49	99.75	95.01	90.97	87.33	84.01
$\mathcal{N}(\Delta g, \mathbb{E}, b+h)$	*mGal*	989.20	87.39	50.94	35.95	31.40	27.79
$\mathcal{N}(\delta g, \mathbb{E}, b+h)$	*mGal*	993.45	102.67	68.51	59.96	53.48	48.14
$\mathcal{N}(\xi, \mathbb{E}, b+h)$	*arcsec.*	126.78	21.72	11.63	7.52	5.87	5.41
$\mathcal{N}(\eta, \mathbb{E}, b+h)$	*arcsec.*	94.77	15.70	9.95	8.45	7.63	7.09
$\mathcal{N}(T_{rr}, \mathbb{E}, b+h)$	*Eötvös*	1296.45	10.49	2.39	1.29	0.81	0.55

Table 1. The norms $\mathcal{N}(G, \mathbb{E}, b+h)$ on the ellipsoids with semiminor axis $b+h$ for several heights h.

4.6.2. *Absolute error.* In some instances it is useful to know how the bound 10^{-6} on the relative error translates into absolute error for various gravimetric quantities at various heights h above the surface of the earth (geoid). The accuracy in absolute units for each of the gravimetric quantities T, ζ, Δg, δg, ξ, η, and T_{rr} is given in the next table.

h	km	0	20	100	200	500
T	m^2/s^2	1.0×10^{-3}	1.0×10^{-3}	9.5×10^{-4}	8.7×10^{-4}	7.1×10^{-4}
ζ	m	1.1×10^{-4}	1.0×10^{-4}	1.0×10^{-4}	9.5×10^{-5}	8.4×10^{-5}
Δg	*mGal*	9.9×10^{-4}	3.4×10^{-4}	8.7×10^{-5}	5.1×10^{-5}	2.8×10^{-5}
δg	*mGal*	9.9×10^{-4}	3.5×10^{-4}	1.0×10^{-4}	6.9×10^{-5}	4.8×10^{-5}
ξ	*arcsec.*	1.3×10^{-4}	5.2×10^{-5}	2.2×10^{-5}	1.2×10^{-5}	5.4×10^{-6}
η	*arcsec.*	8.6×10^{-5}	4.1×10^{-5}	1.4×10^{-5}	9.6×10^{-6}	7.1×10^{-6}
T_{rr}	*Eötvös*	1.3×10^{-3}	1.3×10^{-4}	1.0×10^{-5}	2.4×10^{-6}	5.5×10^{-7}

Table 2. Evaluation accuracy as an absolute error bound

4.6.3. *Norms on ellipsoids versus norms on spheres.* One may avoid the use of ellipsoidal coordinates as described in §4.2 and work only in spherical coordinates. Then the target error estimate would be

$$\max_{\substack{(\theta_P,\lambda_P)\in\mathbb{S}^2,\\ U_0\le u_P\le U_1}} \frac{|\tilde{G}(r_P,\theta_P,\lambda_P) - G(r_P,\theta_P,\lambda_P)|}{\mathcal{N}(G,\mathbb{S}^2,r_P)} \le 10^{-6}, \tag{32}$$

where $(r_P, \theta_P, \lambda_P)$ and $(u_P, \varphi_P, \lambda_P)$ denote the spherical and the ellipsoidal coordinates of a point P, respectively, and

$$\mathcal{N}(G, \mathbb{S}^2, r) = \max_{(\varphi,\lambda)\in\mathbb{S}^2} |G(r, \theta, \lambda)|$$

denotes the norm of G on the sphere of radius r.

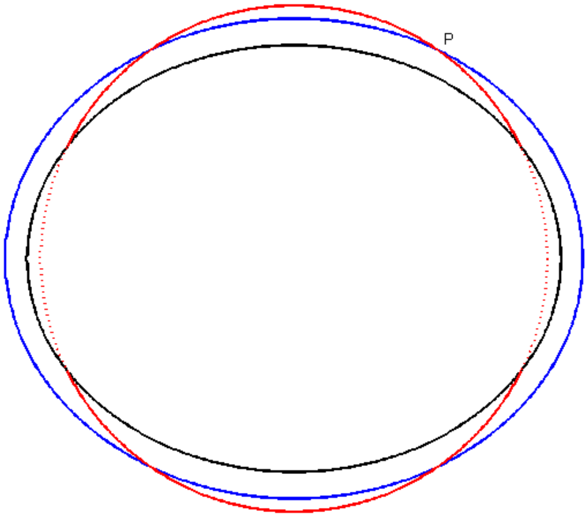

FIGURE 1. Point P above the Earth reference ellipsoid (black) congruent to a confocal ellipsoid (blue) and to a sphere (red). Parts of the sphere (dotted red) may pass inside the reference ellipsoid.

The main drawback of an error estimate as in (32) is the very high oscillation of G in the domain under the surface of the Earth (see dotted red curves in Figure 1), i.e. whenever $b \le r_P \le a$, $b \le u_P$. This would lead to the use of larger size of RAM because more concentric spheres will be needed to get (32) compared with the number of confocal ellipsoids necessary for (31).

$G \setminus \varphi_P$	$\pi/2$	$5\pi/12$	$\pi/3$	$\pi/4$	$\pi/6$	$\pi/12$	0
ζ	1.00	1.00	1.00	1.01	1.43	3.03	4.45
Δg	0.93	1.07	1.75	5.70	26.32	83.74	129.19
ξ	0.77	0.90	1.61	6.24	29.25	94.73	146.31
η	0.86	1.00	2.22	9.19	40.64	130.28	202.19
T_{rr}	0.68	0.94	2.59	12.07	58.86	191.90	296.94

TABLE 3. The ratio of the norms $\mathcal{N}(G, \mathbb{S}^2, r_P)/\mathcal{N}(G, \mathbb{E}, b)$ for points P on the reference ellipsoid with ellipsoidal coordinates $(b, \varphi_P, \lambda_P)$ and spherical coordinates $(r_P, \theta_P, \lambda_P)$. The entries for T and δg are not reported because they are very close to the entries for ζ and Δg, respectively.

Another advantage of the ellipsoidal coordinates here is the fact that $\mathcal{N}(G, \mathbb{S}^2, r_P)$ could be essentially larger than $\mathcal{N}(G, \mathbb{E}, u_P)$ when the point P is

$G \setminus \varphi_P$	$\pi/2$	$5\pi/12$	$\pi/3$	$\pi/4$	$\pi/6$	$\pi/12$	0
ζ	1.00	1.00	1.00	1.01	1.01	1.02	1.02
Δg	0.93	0.97	1.14	1.47	1.99	2.70	3.10
ξ	0.92	0.94	1.01	1.13	1.33	1.85	2.18
η	0.97	1.00	1.12	1.33	1.61	1.96	2.29
T_{rr}	0.96	1.04	1.34	2.03	3.64	6.81	9.35

TABLE 4. The ratio of the norms $\mathcal{N}(G, \mathbb{S}^2, r_P)/\mathcal{N}(G, \mathbb{E}, b+20,000)$ for points P 20 km above the reference ellipsoid with ellipsoidal coordinates $(b + 20,000, \varphi_P, \lambda_P)$ and spherical coordinates $(r_P, \theta_P, \lambda_P)$. The entries for T and δg are not reported because they are very close to the entries for ζ and Δg, respectively.

away from the equator as Table 3 shows. This means that estimate (32) could be a lot less precise than estimate (31). Of course, if the point is at the equator, then the inequality is reversed but the two norms $\mathcal{N}(G, \mathbb{E}, u_P)$ and $\mathcal{N}(G, \mathbb{S}^2, r_P)$ remain close to each other in this case. Table 3 also shows that the disadvantage of (32) over (31) on the reference ellipsoid is a lot stronger for Δg, δg, ξ, η and T_{rr} than for ζ and T.

The two estimates (31) and (32) become closer with the increase of the point height above the reference ellipsoid. As Table 4 shows the discrepancies at height 20 km for Δg, δg, ξ, η and T_{rr} still exist but they have smaller magnitude, while the discrepancies for ζ and T are practically negligible.

5. The code hsynth_init.f90

The code `hsynth_init.f90` is an auxiliary program design to produce the binary data files `values_*.bin` listed in §2.4, which are not included in the package because of their 9.3 GB sizes. These files are necessary to initialize `hsynth_fast.f90`.

The code computes the gravimetric quantity values at regular points located on confocal ellipsoids from the standard EGM2008 coefficients file `EGM2008_to2190_TideFree`. The "standard" NGA method for evaluating *surface* spherical harmonic *gridded* values is applied. The coefficients of the ellipsoidal harmonic expansions are obtained via Jekeli's transformation from the coefficients of the spherical harmonic expansions (see Subsection 4.2), which in turn are derived as explained in Section 4.1 from the EGM2008 coefficients given in file `EGM2008_to2190_TideFree`. The number of confocal ellipsoids is chosen to cover the range $[U_0, U_1]$ with step h, which in turn depends on the evaluated gravimetric quantity as explained in Subsection 4.3.

The number of regular grid points located on an ellipsoid is: 4015×8028 for T and ζ ($2.69' \times 2.69'$ grid); 3346×6690 for Δg, δg, ξ, η and T_{rr} ($3.23' \times 3.23'$ grid). The grid is further extended as explain in §4.4 and §4.5. The number of confocal ellipsoids used by the code is 38 for T and ζ, 53 for Δg and δg, 54 for ξ, 55 for η and T_{rr}.

The precomputed values are not included in the package because of their 9.3 GB sizes. On the computer where the tests were performed it takes approximately 54 minutes to run the code for T or ζ, 1 hour for Δg, δg or T_{rr} and 2 hours for ξ or η.

6. The code hsynth_standard.f90

The purpose of the code `hsynth_standard.f90` is to test the *accuracy* of the main code `hsynth_fast.f90` and to be used as a benchmark for its *speed*.

The code `hsynth_standard.f90` evaluates the gravimetric quantities using the "standard" method of the NGA software on the base of the EGM2008 coefficients. Only the north-south and easth-westh deflections of the vertical ξ and η are computed by a different methods, which are more stable. The evaluations are done in spherical coordinates without passing to ellipsoidal coordinates as in the other two codes.

The model coefficients are read from `EGM2008_to2190_TideFree`. The file `scattered_points_values.dat`, which is the output of `hsynth_fast.f90`, is also used as an input file for the testing program.

Among the displayed program statistics one can find the maximal relative error (31) of the gravimetric quantity computed by `hsynth_fast.f90`. The speed of our realization of the two methods can be compared using the reported numbers "Values per second". In both programs these numbers are formed on the base of pure computational time, ignoring the time necessary to read the input or to write the output.

Two options are provided in the code for evaluating the east-west deflection of the vertical η. The option `eta` implements representation (20)–(26) and is used to measure the accuracy of `hsynth_fast.exe`, while the option `eta1` implements representation (19) and could be used to compare the computational speed. The option `eta1` follows the method of the "standard" NGA software, which is two times faster than the method used in option `eta` but tends to lose precision near the poles due to division by $\sin\theta$.

As the speed of the method of the "standard" NGA software is approximately 46 values per second for T, ζ, Δg, δg and T_{rr} or 23 values per second for ξ and η it is not advisable to run this code with more than $10,000$ points!

7. Experiments

The software has been extensively tested on a 2.4 GHz PC, CPU Intel Core i7 with 16 GB of RAM and 250 GB SSD. The Solid-State Disk is used to store the binary data files `values_*.bin`, the input file `scattered_points.dat` and the output file `scattered_points_values.dat`. Program `hsynth_fast.f90` uses 11 GB of RAM.

7.1. **The test files.** The test input file `scattered_points.dat` consists of the geographical geocentric spherical coordinates of 1,000 (or 1,000,000) points, obtained from their ellipsoidal coordinates, such that the latitude, the longitude and the semi-minor axis are randomly distributed in $[-90, 90]$, $[0, 360)$ and $[U_0, U_1]$, respectively. Here $U_0 = b - 125\ m$, $U_1 = b + 544,000\ m$ and b is the Earth semi-minor axis, $b \approx 6,356,752\ m$. The first three points from the file are modified to be at the North pole, at the South pole and at the Equator of the Earth reference ellipsoid.

7.2. **Tests.** The codes are run under GNU FORTRAN compiled with GCC v.4.9.2 with `-mcmodel=medium` and `-O3` options in Windows 10 terminal.

The testing statistics of program `hsynth_fast.exe` for the test input files `scattered_points.dat` with 1,000,000 and 1,000 points and of program `hsynth_standard.exe` with 1,000 points follow for different gravimetric quantities. For the east-west deflection of the vertical η the statistics of `hsynth_standard.exe` for option `eta1` are also included.

7.2.1. *The disturbing potential T.*

```
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: disturbing potential T in m^2/s^2
   Total time               =        78.29099 CPU seconds
   Grid values load time    =        30.34288 CPU seconds
   Coordinates load time    =         6.25691 CPU seconds
   Synthesis time           =        24.45910 CPU seconds
   Values write time        =        13.29679 CPU seconds
   Semi-length of supports  =   13;  13
   Total number of points   =   1000000
   Values per second        =     40884.57727
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: disturbing potential T in m^2/s^2
   Total time               =        34.43972 CPU seconds
   Grid values load time    =        30.66280 CPU seconds
   Coordinates load time    =         0.02949 CPU seconds
   Synthesis time           =         0.02914 CPU seconds
   Values write time        =         0.01431 CPU seconds
   Semi-length of supports  =   13;  13
   Total number of points   =      1000
   Values per second        =     34312.76138
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: disturbing potential T in m^2/s^2
   Total time                  =        33.04112 CPU seconds
   Model load time             =        11.35298 CPU seconds
   Coefficient formation time  =         0.16091 CPU seconds
   Coordinates load time       =         0.00900 CPU seconds
   Synthesis time              =        21.50353 CPU seconds
   Values write time           =         0.01462 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.02150 CPU seconds
   Values per second           =        46.50398
   Maximal relative error      =   0.4144485E-07
   Geocentric coordinates of the maximal error:
   (    5.46826406,    79.58292965,   6497932.9395)
---------------------------------------------------------------
```

7.2.2. *The height anomaly ζ.*

```
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: height anomaly in meters
   Total time               =        78.46042 CPU seconds
   Grid values load time    =        30.52104 CPU seconds
   Coordinates load time    =         6.18772 CPU seconds
   Synthesis time           =        24.58356 CPU seconds
   Values write time        =        13.18387 CPU seconds
   Semi-length of supports  =   13;  13
   Total number of points   =   1000000
   Values per second        =     40677.58347
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: height anomaly in meters
   Total time               =        34.52072 CPU seconds
   Grid values load time    =        30.33005 CPU seconds
   Coordinates load time    =         0.03029 CPU seconds
   Synthesis time           =         0.03529 CPU seconds
   Values write time        =         0.01439 CPU seconds
   Semi-length of supports  =   13;  13
   Total number of points   =      1000
   Values per second        =     28335.56291
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: height anomaly in meters
   Total time                  =        33.04321 CPU seconds
   Model load time             =        11.31501 CPU seconds
   Coefficient formation time  =         0.14015 CPU seconds
   Coordinates load time       =         0.00878 CPU seconds
   Synthesis time              =        21.56465 CPU seconds
   Values write time           =         0.01455 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.02156 CPU seconds
   Values per second           =        46.37219
   Maximal relative error      =   0.4098860E-07
   Geocentric coordinates of the maximal error:
   (    5.46826406,    79.58292965,   6497932.9395)
---------------------------------------------------------------
```

7.2.3. *The gravity anomaly Δg.*

```
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: gravity anomaly Dg in mGal
   Total time               =        93.76596 CPU seconds
   Grid values load time    =        29.97501 CPU seconds
   Coordinates load time    =         6.38973 CPU seconds
   Synthesis time           =        40.45838 CPU seconds
   Values write time        =        13.22747 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =   1000000
   Values per second        =     24716.75538
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: gravity anomaly Dg in mGal
   Total time               =        33.32451 CPU seconds
   Grid values load time    =        29.48255 CPU seconds
   Coordinates load time    =         0.01938 CPU seconds
   Synthesis time           =         0.04338 CPU seconds
   Values write time        =         0.01393 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =      1000
   Values per second        =     23050.41943
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: gravity anomaly Dg in mGal
   Total time                  =        33.64376 CPU seconds
   Model load time             =        11.57636 CPU seconds
   Coefficient formation time  =         0.35760 CPU seconds
   Coordinates load time       =         0.00868 CPU seconds
   Synthesis time              =        21.68656 CPU seconds
   Values write time           =         0.01449 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.02169 CPU seconds
   Values per second           =        46.11151
   Maximal relative error      =   0.5255340E-07
   Geocentric coordinates of the maximal error:
   (  -16.85575013,   292.39107577,   6632672.5475)
---------------------------------------------------------------
```

7.2.4. The gravity disturbance δg.

```
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: gravity disturbance dg in mGal
   Total time               =        93.91593 CPU seconds
   Grid values load time    =        29.80503 CPU seconds
   Coordinates load time    =         6.43476 CPU seconds
   Synthesis time           =        40.79958 CPU seconds
   Values write time        =        13.23632 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =   1000000
   Values per second        =     24510.05707
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: gravity disturbance dg in mGal
   Total time               =        33.97065 CPU seconds
   Grid values load time    =        30.30102 CPU seconds
   Coordinates load time    =         0.02210 CPU seconds
   Synthesis time           =         0.04532 CPU seconds
   Values write time        =         0.01492 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =      1000
   Values per second        =     22066.81079
---------------------------------------------------------------
---------------------------------------------------------------
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: gravity disturbance dg in mGal
   Total time                  =        33.57853 CPU seconds
   Model load time             =        11.55973 CPU seconds
   Coefficient formation time  =         0.35318 CPU seconds
   Coordinates load time       =         0.00872 CPU seconds
   Synthesis time              =        21.64259 CPU seconds
   Values write time           =         0.01425 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.02164 CPU seconds
   Values per second           =        46.20520
   Maximal relative error      =   0.4633827E-07
   Geocentric coordinates of the maximal error:
   (    6.06207401,    75.35419746,   6498618.3602)
---------------------------------------------------------------
```

7.2.5. *The north-south deflection of the vertical ξ.*

```
---------------------------------------------------------------
        program  hsynth_fast
---------------------------------------------------------------
 Gravimetric quantity: north-south vertical deflection in arcseconds
 Total time              =       93.59642 CPU seconds
 Grid values load time   =       30.07957 CPU seconds
 Coordinates load time   =        6.23561 CPU seconds
 Synthesis time          =       40.27181 CPU seconds
 Values write time       =       13.20634 CPU seconds
 Semi-length of supports =   16;  16
 Total number of points  =   1000000
 Values per second       =     24831.26635
---------------------------------------------------------------
---------------------------------------------------------------
        program  hsynth_fast
---------------------------------------------------------------
 Gravimetric quantity: north-south vertical deflection in arcseconds
 Total time              =       34.46133 CPU seconds
 Grid values load time   =       30.58784 CPU seconds
 Coordinates load time   =        0.02337 CPU seconds
 Synthesis time          =        0.04879 CPU seconds
 Values write time       =        0.01440 CPU seconds
 Semi-length of supports =   16;  16
 Total number of points  =      1000
 Values per second       =     20495.92869
---------------------------------------------------------------
---------------------------------------------------------------
        program  hsynt_standard
---------------------------------------------------------------
 Gravimetric quantity: north-south vertical deflection in arcseconds
 Total time                 =       54.52510 CPU seconds
 Model load time            =       11.26231 CPU seconds
 Coefficient formation time =        0.36457 CPU seconds
 Coordinates load time      =        0.00860 CPU seconds
 Synthesis time             =       42.87533 CPU seconds
 Values write time          =        0.01423 CPU seconds
 Total number of points     =      1000
 Execution time per point   =        0.04288 CPU seconds
 Values per second          =       23.32343
 Maximal relative error     =   0.6371636E-07
 Geocentric coordinates of the maximal error:
 (  -15.51338799,    75.26156371,   6902930.3459)
---------------------------------------------------------------
```

7.2.6. *The east-west vertical deflection η.*

Option `eta`

```
---------------------------------------------------------------
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: east-west vertical deflection in arcseconds
   Total time               =        95.46257 CPU seconds
   Grid values load time    =        31.29893 CPU seconds
   Coordinates load time    =         6.41526 CPU seconds
   Synthesis time           =        40.50132 CPU seconds
   Values write time        =        13.24499 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =   1000000
   Values per second        =     24690.55506
===============================================================
          program  hsynth_fast
---------------------------------------------------------------
   Gravimetric quantity: east-west vertical deflection in arcseconds
   Total time               =        34.90200 CPU seconds
   Grid values load time    =        30.90171 CPU seconds
   Coordinates load time    =         0.02490 CPU seconds
   Synthesis time           =         0.05039 CPU seconds
   Values write time        =         0.01427 CPU seconds
   Semi-length of supports  =   16;  16
   Total number of points   =      1000
   Values per second        =     19845.94759
===============================================================
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: east-west vertical deflection in arcseconds
   Total time                  =        54.91672 CPU seconds
   Model load time             =        11.55685 CPU seconds
   Coefficient formation time  =         0.38586 CPU seconds
   Coordinates load time       =         0.00877 CPU seconds
   Synthesis time              =        42.95107 CPU seconds
   Values write time           =         0.01411 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.04295 CPU seconds
   Values per second           =        23.28231
   Maximal relative error      =   0.5907279E-07
   Geocentric coordinates of the maximal error:
   (   55.51261069,   303.36387932,   6681813.8032)
---------------------------------------------------------------
```

Option eta1

```
---------------------------------------------------------------
          program  hsynt_standard
---------------------------------------------------------------
   Gravimetric quantity: east-west vertical deflection in arcseconds
   Total time                  =        33.16066 CPU seconds
   Model load time             =        11.40941 CPU seconds
   Coefficient formation time  =         0.14106 CPU seconds
   Coordinates load time       =         0.00873 CPU seconds
   Synthesis time              =        21.58586 CPU seconds
   Values write time           =         0.01548 CPU seconds
   Total number of points      =      1000
   Execution time per point    =         0.02159 CPU seconds
   Values per second           =        46.32661
   Maximal relative error      =   0.5973175E-07
   Geocentric coordinates of the maximal error:
   (    0.69906096,    88.11711514,   6675337.1589)
---------------------------------------------------------------
```

7.2.7. *The second radial derivative of the disturbing potential T_{rr}.*

```
---------------------------------------------------------------
        program  hsynth_fast
---------------------------------------------------------------
 Gravimetric quantity: second radial derivative of T in Eotvos Units
 Total time               =       104.01463 CPU seconds
 Grid values load time    =        30.60787 CPU seconds
 Coordinates load time    =         6.19974 CPU seconds
 Synthesis time           =        50.20390 CPU seconds
 Values write time        =        13.17706 CPU seconds
 Semi-length of supports  =   16;  16
 Total number of points   =   1000000
 Values per second        =     19918.77257
---------------------------------------------------------------
---------------------------------------------------------------
        program  hsynth_fast
---------------------------------------------------------------
 Gravimetric quantity: second radial derivative of T in Eotvos Units
 Total time               =        34.85333 CPU seconds
 Grid values load time    =        30.83817 CPU seconds
 Coordinates load time    =         0.02228 CPU seconds
 Synthesis time           =         0.06690 CPU seconds
 Values write time        =         0.01451 CPU seconds
 Semi-length of supports  =   16;  16
 Total number of points   =      1000
 Values per second        =     14947.80141
---------------------------------------------------------------
---------------------------------------------------------------
        program  hsynt_standard
---------------------------------------------------------------
 Gravimetric quantity: second radial derivative of T in Eotvos Units
 Total time                  =        33.39470 CPU seconds
 Model load time             =        11.42128 CPU seconds
 Coefficient formation time  =         0.34734 CPU seconds
 Coordinates load time       =         0.00856 CPU seconds
 Synthesis time              =        21.60326 CPU seconds
 Values write time           =         0.01419 CPU seconds
 Total number of points      =      1000
 Execution time per point    =         0.02160 CPU seconds
 Values per second           =        46.28931
 Maximal relative error      =   0.4282001E-07
 Geocentric coordinates of the maximal error:
 (  -16.85575013,   292.39107577,   6632672.5475)
---------------------------------------------------------------
```

The relative error measured for the 1,000 point test file vary from 4.10×10^{-8} to 6.37×10^{-8}. Recall that the relative error is computed by (31) and some values of the norms $\mathcal{N}(G, \mathbb{E}, u)$ are given in Table 1.

The improvement in computational speed of our code `hsynth_fast` (measured by `Values per second`) compared to the software `hsynth_standard` using the "standard" NGA method is given in the last column of Table 5.

Gravimetric quantity	`hsynth_standard` (1,000 points)	`hsynth_fast` (1,000,000 points)	improvement (times)
T	46.50	40884.58	879.24
ζ	46.37	40677.58	877.24
Δg	46.11	24716.76	536.04
δg	46.21	24510.06	530.41
ξ	23.32	24831.27	1064.81
η	46.33	24690.56	532.93
T_{rr}	46.29	19918.77	430.30

TABLE 5. `Values per second` for codes `hsynth_standard` and `hsynth_fast` and the improvement in computational speed

The 1,000,000 point input file is used to measure the speed of `hsynth_fast` because the data for the 1,000 point input file are unstable due to the unstable output of Fortran intrinsic timing routine `system_clock` for small time intervals (see the statistics above). The computational speed of `hsynth_standard`, however, is stable for the 1,000 point input file. The different entries in the third column of Table 5 reflects the different approximation methods used in `hsynth_fast` to compute the respective gravimetric quantities. The speed for ξ in the second column of Table 5 is twice smaller than for the other quantities because two harmonic expansions (instead of one) are used for its evaluation in `hsynth_standard`. In order to measure the speed for η in the second column of Table 5 we use option `eta1`. For option `eta` the speed is similar to the speed for ξ as the statistics above show.

7.3. **Other experiments.** The code `hsynth_fast.f90` has been tested on up to $30,000,000$ points, which were processed in the speed given in the third column of Table 5. The relative error did not exceed 4.64×10^{-7} for all tests measured with `hsynth_standard.f90`.

8. Conclusions

The experiments with the software described above demonstrate the capability of our needlet method for fast evaluation of gravimetric quantities represented in terms of *solid* spherical harmonics at many scattered points *in the exterior space*. The current version `hsynth_fast.f90` of our software runs at speed between 20,00 and 40,000 point evaluations per second, which is between 430 and 1064 times faster than the software using the "standard" NGA methods.

The needlet method described in this manual is not limited to spherical/ellipsoidal harmonics of degree 2190. There is strong evidence that this technology, applied with the current computers with double precision arithmetic, can handle quantities represented in spherical/ellipsoidal harmonics of degrees above 10,000. Therefore, the needlets can be a handy tool in future more accurate models.

References

[1] K.G. Ivanov, P. Petrushev, Highly effective stable evaluation of bandlimited functions on the sphere, Numerical Algorithms **71** (2016), 585–611.

[2] C. Jekeli, The exact transformation between ellipsoidal and spherical harmonic expansions, Manuscripta Geodaetica, **13** (1988), 106–113.

[3] N.K. Pavlis, S.A. Holmes, S.C. Kenyon, J.K. Factor, The development and evaluation of the Earth Gravitational Model 2008 (EGM2008). J. Geophys. Res., **117** (2012), B04406, doi:10.1029/2011JB008916