Error propagation for high-degree gravity models developed from the GRAIL mission

Terence J. Sabaka[1], Joseph B. Nicholas[2], Sander Goossens[3], Frank G. Lemoine[1], and Erwan Mazarico[4]

[1]Planetary Geodynamics Laboratory, NASA Goddard Space Flight Center, Greenbelt, MD
[2]Emergent Space Technologies, Greenbelt, MD
[3]CRESST, University of Maryland, Baltimore County, MD
[4]Department of Earth, Atmospheric, and Planetary Sciences, Massachusetts Institute of Technology, Cambridge, MA

April 1, 2014

1 Introduction

The NASA Discovery dual satellite GRAIL mission mapped the gravity field of the Moon using intersatellite Ka-Band Range Rate (KBRR) data from March 1 to May 29, 2012, and from August 30 to December 14, 2012 (*Asmar et al.*, 2013; *Zuber et al.*, 2013). During the primary mission, the spacecraft orbited at a mean altitude of 55 km, whereas during the extended mission the spacecraft orbited at a mean altitude of 23 km. Models to spherical harmonic degree 660 were developed solely from the primary mission data (*Lemoine et al.*, 2013; *Konopliv et al.*, 2013). The extended mission data have sensitivity to much higher degrees, and already solutions to degree/order (d/o) 900 have been developed.

It is of interest to spatially map the error of the gravity model solution while avoiding the computational burden of manipulating a full covariance matrix or its factorization to d/o 900 to propagate the errors. This document outlines the procedures that have been developed by the Goddard Space Flight Center (GSFC) GRAIL gravity team using a Monte Carlo approach to compute the error in large gravity solutions (i.e., to d/o 900). The procedure is verified with the Kaguya gravity model SGM150J, a pre-GRAIL gravity model of the Moon complete to d/o 150 (*Goossens et al.*, 2011), by intercomparing the Monte Carlo approach with a direct propagation of the errors using the full covariance matrix. The procedure is then applied to a lunar gravity solution to d/o 900, GRGM900C (*Lemoine et al.*, 2014).

As shall be seen, samples drawn from the error distributions of the GSFC GRAIL gravity models, known as "clones", were generated as members of ensembles used to make Monte Carlo estimates of various error measurements of the base models. These clones are of interest in their own right since they provide realizations of the likely error perturbations in the actual model coefficients. Therefore, they will be made available on the Planetary Data System (PDS) Geosciences node at Washington University in St. Louis (http://pds-geosciences.wustl.edu/missions/grail/default.htm).

2 Error Propagation

The assessment of error in a gravity model often entails the mapping or propagation of errors in the Stokes coefficients into the errors of some linear functional of the coefficients. For instance, if the local gravity anomaly $\Delta g(\mathbf{r})$ at position $\mathbf{r}$ is given by

$$\Delta g(\mathbf{r}) = \mathbf{f}(\mathbf{r})^{\mathrm{T}}\mathbf{c}, \tag{1}$$

where $\mathbf{f}(\mathbf{r})^{\mathrm{T}}$ is a linear operator and $\mathbf{c}$ is the vector of Stokes coefficients of the base model, then the variance in $\Delta g(\mathbf{r})$ is given by

$$V\left[\Delta g(\mathbf{r})\right] = \mathbf{f}(\mathbf{r})^{\mathrm{T}}\mathbf{C}\mathbf{f}(\mathbf{r}), \tag{2}$$

where $\mathbf{C}$ is the error-covariance matrix of $\mathbf{c}$. Note that any evaluation of eq. 2 necessarily requires calculations involving the full error-covariance matrix or its factorization. For large numbers of evaluations this can become prohibitively expensive if $\mathbf{C}$ is large.

3 Monte Carlo Approximation

Consider a vector-valued random variable $\mathbf{z}$ whose mean is $\mathbf{c}$ and covariance is $\mathbf{C}$. These may be expressed in terms of integrals over a probability space such that

$$\begin{aligned} \mathbf{c} &= \int \mathbf{z}\, p\left(\mathbf{z}\right)\, d\mathbf{z}, \\ &= E\left[\mathbf{z}\right], \end{aligned} \tag{3}$$

$$\begin{aligned} \mathbf{C} &= \int \left[\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)^{\mathrm{T}}\right] p\left(\mathbf{z}\right)\, d\mathbf{z}, \\ &= E\left[\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)^{\mathrm{T}}\right], \end{aligned} \tag{4}$$

where p is the probability density function (pdf) of the distribution and $E\left[\cdot\right]$ is the expectation operator. The variance in $\Delta g(\mathbf{r})$ may now be expressed as

$$\begin{aligned} V\left[\Delta g(\mathbf{r})\right] &= E\left[\left(\Delta g(\mathbf{r}) - E\left[\Delta g(\mathbf{r})\right]\right)^2\right], \\ &= \mathbf{f}(\mathbf{r})^{\mathrm{T}} E\left[\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)\left(\mathbf{z} - E\left[\mathbf{z}\right]\right)^{\mathrm{T}}\right]\mathbf{f}(\mathbf{r}). \end{aligned} \tag{5}$$

The essence of Monte Carlo simulation is to approximate the integrals in eqs. 3 and 4 as summations over an ensemble of K members of the distribution such that an unbiased estimate is given by

$$\tilde{V}\left[\Delta g(\mathbf{r})\right] = \mathbf{f}(\mathbf{r})^{\mathrm{T}}\left(\frac{1}{K-1}\sum_{j=1}^{K}\left(\mathbf{z}_j - \overline{\mathbf{z}}\right)\left(\mathbf{z}_j - \overline{\mathbf{z}}\right)^{\mathrm{T}}\right)\mathbf{f}(\mathbf{r}), \tag{6}$$

where

$$\overline{\mathbf{z}} = \frac{1}{K}\sum_{j=1}^{K}\mathbf{z}_j. \tag{7}$$

4 Clone generation

We now focus on how to generate the ensemble members $\mathbf{z}_j$ such that their covariance matches that of the base model, $\mathbf{C}$. This can be achieved by an Affine mapping of a standardized random variable $\mathbf{t}$ (mean $\mathbf{0}$ and covariance $\mathbf{I}$, denoted as $\mathbf{t} \sim \mathcal{D}(\mathbf{0}, \mathbf{I})$ where $\mathcal{D}$ indicates the type of distribution) into $\mathbf{z}$ through a linear mapping matrix $\mathbf{S}$ and a translation $\boldsymbol{\mu}$. If $\mathbf{S}$ is chosen as a "square-root" matrix of $\mathbf{C}$, then

$$\begin{aligned} \mathbf{z} &= \mathbf{S}\mathbf{t} + \boldsymbol{\mu}, && (8) \\ E[\mathbf{z}] &= \boldsymbol{\mu}, && (9) \\ V[\mathbf{z}] &= \mathbf{S}V[\mathbf{t}]\mathbf{S}^{\mathrm{T}}, \\ &= \mathbf{S}\mathbf{I}\mathbf{S}^{\mathrm{T}}, \\ &= \mathbf{C}, && (10) \end{aligned}$$

where $\mathbf{C} = \mathbf{S}\mathbf{S}^{\mathrm{T}}$. We are free to select $\boldsymbol{\mu}$ and if we choose it to be $\mathbf{c}$, then the ensemble members are referred to as "clones" of the base model $\mathbf{c}$. What remains is to choose $\mathbf{S}$ and the distribution from which $\mathbf{t}$ is sampled.

4.1 Covariance square-root matrix

Smith et al. (2012) have used clones to compute statistical error properties of their Mercury gravity fields derived from MESSENGER data. They choose $\mathbf{S}$ based upon the eigen-decomposition of $\mathbf{C}$

$$\mathbf{C} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}}, \tag{11}$$

where $\mathbf{U}$ is an orthogonal matrix whose columns are the eigenvectors of $\mathbf{C}$ and where $\boldsymbol{\Lambda}$ is a diagonal matrix whose non-negative Λ_{jj} entry is the eigenvalue corresponding to the j^{th} column of $\mathbf{U}$. This leads to

$$\mathbf{S} = \mathbf{U}\boldsymbol{\Lambda}^{1/2}, \tag{12}$$

where $\boldsymbol{\Lambda}^{1/2}$ is a diagonal matrix whose positive $\Lambda_{jj}^{1/2}$ entry is $\sqrt{\Lambda_{jj}}$. However, a general eigen-decomposition is very computationally intensive and they considered only d/o 20 harmonic solutions. For our GRAIL solutions at d/o 900 and above, this is prohibitive, so we consider another factorization based upon our use of the Square-Root Information Filter (SRIF). This filter supplies the Cholesky square-root $\mathbf{R}$ of the Normal matrix $\mathbf{N}$

$$\mathbf{N} = \mathbf{R}^{\mathrm{T}}\mathbf{R}, \tag{13}$$

and since $\mathbf{C} = \mathbf{N}^{-1}$, then

$$\mathbf{C} = \mathbf{R}^{-1}\mathbf{R}^{-\mathrm{T}}, \tag{14}$$

which leads to

$$\mathbf{S} = \mathbf{R}^{-1}. \tag{15}$$

From eq. 8 we see that we only need to solve

$$\mathbf{R}(\mathbf{z} - \boldsymbol{\mu}) = \mathbf{t} \tag{16}$$

to obtain $\mathbf{z}$.

4.2 Standardized random vectors

First consider that eq. 2 may be rewritten as

$$\begin{aligned} V\left[\Delta g(\mathbf{r})\right] &= \mathbf{f}(\mathbf{r})^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{f}(\mathbf{r}), \\ &= Tr\left[\mathbf{S}^{\mathrm{T}}\mathbf{f}(\mathbf{r})\mathbf{f}(\mathbf{r})^{\mathrm{T}}\mathbf{S}\right], \\ &= Tr\left[\mathbf{A}\right], \end{aligned} \tag{17}$$

where $Tr\left[\cdot\right]$ is the trace operator. Now, an unbiased estimate, α, of the trace of a symmetric, positive semi-definite matrix $\mathbf{A}$, like that given in eq. 17, is given by

$$\alpha = \mathbf{t}^{\mathrm{T}}\mathbf{A}\mathbf{t}, \tag{18}$$

where $\mathbf{t}$ is a random vector of mean $\mathbf{0}$ and covariance $\mathbf{I}$. If $\mathbf{t}$ is drawn from a standardized Gaussian distribution, $\mathbf{t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$, then the variance of this estimate is given by

$$V\left[\alpha_g\right] = 2\|\mathbf{A}\|_F^2, \tag{19}$$

where the subscript "g" indicates a Gaussian distribution is being used, and $\|\cdot\|_F$ is the Frobenius norm of a matrix, that is, the square-root of the sum of the squares of its elements. However, we are not restricted to using Gaussian deviates. If instead we use a standardized, symmetric Rademacher distribution, where each element of $\mathbf{t}$ is either $+1$ or -1 with equal probability, $\mathbf{t} \sim \mathcal{R}\left(\mathbf{0}, \mathbf{I}\right)$, then we also have an unbiased estimate of the trace of $\mathbf{A}$, but with a variance of (*Hutchinson*, 1989)

$$V\left[\alpha_r\right] = 2\|\hat{\mathbf{A}}\|_F^2, \tag{20}$$

where the subscript "r" indicates the Rademacher distribution is being used and $\hat{\mathbf{A}}$ is a matrix identical to $\mathbf{A}$ except with zeros on the main diagonal (see Appendix A). Clearly $V\left[\alpha_r\right] < V\left[\alpha_g\right]$, and so we use the standardized, symmetric Rademacher distribution for selecting $\mathbf{t}_j$. This was also used by *Smith et al.* (2012). The set of clones provided to the community are then given by

$$\left.\begin{aligned} \mathbf{t}_j &\sim \mathcal{R}\left(\mathbf{0}, \mathbf{I}\right) \\ \mathbf{z}_j &= \mathbf{c} + \mathbf{R}^{-1}\mathbf{t}_j \end{aligned}\right\}, j = 1, \ldots, K. \tag{21}$$

4.3 Confidence regions

We have seen that distributions whose ensembles are consistent with both the mean and covariance of the GRAIL gravity models are not unique. In this section we discuss these differences in the context of confidence regions, that is, regions where a certain percentage of samples are expected to reside. These differences in sampling patterns can have an effect upon the results of those who use them and should therefore be understood.

Let us assume that the distribution of model parameter vectors is Gaussian such that its pdf is given by

$$p(\mathbf{z}) = \frac{1}{\sqrt{(2\pi)^N \det\left(\mathbf{C}\right)}} e^{-\frac{1}{2}(\mathbf{z}-\mathbf{c})^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{z}-\mathbf{c})}, \tag{22}$$

where N is the dimension of the model vectors and $\det(\cdot)$ is the determinant of the argument. The contours of p lie on hyper-ellipsoidal, or N-ellipsoid, surfaces centered on $\mathbf{c}$ with principal axes defined as the eigenvectors of $\mathbf{C}$ and lengths along these axes proportional to the square-root of the corresponding eigenvalues. A χ-sigma contour is then defined as the set of points $\mathbf{z}$ such that $(\mathbf{z}-\mathbf{c})^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{z}-\mathbf{c}) = \chi^2$.

For a given set of samples, we are interested in the percentage that are expected to reside in a certain region, $\widetilde{\Omega}$, of the entire distribution space, Ω. This percentage q can be expressed as

$$q = 100 \times \int_{\widetilde{\Omega}} p(\mathbf{z})\, d\Omega. \tag{23}$$

In fact, we usually choose q first and then $\widetilde{\Omega}$ follows and is known as a "confidence region" (CR) at the "q percentage level". For Gaussian distributions it is natural to choose $\widetilde{\Omega}$ as the N-ellipsoid defined by a χ-sigma level line. This is greatly expedited if we first transform $p(\mathbf{z})$ to a new system where one coordinate is defined as $y = (\mathbf{z}-\mathbf{c})^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{z}-\mathbf{c})$ and the remaining $N-1$ coordinates, $\omega_1, \ldots, \omega_{N-1}$, correspond to the angular variables describing the unit N-sphere. This allows us to rewrite eq. 23 as

$$q = 100 \times \frac{1}{A_S} \int_{\widetilde{\Omega}} p_{\chi^2}(x|N)\, dx\, dA(\omega_1, \ldots, \omega_{N-1}), \tag{24}$$

where p_{χ^2} is the pdf of the familiar "chi-squared" distribution with N degrees of freedom given by

$$p_{\chi^2}(y|N) = \frac{1}{2^{N/2}\Gamma(N/2)} y^{(N-2)/2} e^{-y/2}, \tag{25}$$

with the Gamma function defined as

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t}\, dt, \tag{26}$$

and A_S and dA are the area and differential area element of the N-sphere. If we allow $\widetilde{\Omega}$ to only vary along the y coordinate, then eq. 24 may be rewritten as

$$q = 100 \times \int_0^y p_{\chi^2}(x|N)\, dx. \tag{27}$$

For a given q value, the corresponding y value can be determined by using the inverse χ^2 cumulative density function (cdf) for the given degrees of freedom.

4.3.1 Rademacher clones

If we evaluate p at the locations of the clones $\mathbf{z}_j = \mathbf{c} + \mathbf{R}^{-1}\mathbf{t}_j$ generated using $\mathbf{t}_j \sim \mathcal{R}(\mathbf{0}, \mathbf{I})$, then

$$p(\mathbf{z}_j) = \frac{1}{\sqrt{(2\pi)^N \det(\mathbf{C})}} e^{-N/2}, \tag{28}$$

which means that they all lie on the $\sqrt{N}$-sigma contour. Note that this is true regardless of the form of the square-root matrix $\mathbf{S}$ that is used, e.g., eqs. 12 or 15. Because of the binary nature of

each element of $\mathbf{t}_j$, these clones can occupy any of 2^N discrete locations on this contour. However, different $\mathbf{S}$ will lead to different patterns.

To illustrate this, we look at a $N = 2$ example where $\mathbf{c} = \mathbf{0}$ and

$$\mathbf{C} = \begin{pmatrix} 134.72907784289 & -34.1412432926311 \\ -34.1412432926311 & 18.2709221571098 \end{pmatrix}. \tag{29}$$

This matrix has eigenvalues of 144 and 9 and corresponding eigenvectors in the first and second columns, respectively, of

$$\mathbf{U} = \begin{pmatrix} -0.96505259055656 & 0.26205628681654 \\ 0.262056286816539 & 0.96505259055656 \end{pmatrix}. \tag{30}$$

We generated $K = 5000$ clones using Rademacher $\mathbf{t}_j$ and have plotted the locations of $\mathbf{z}_j$ on top of the contours of $p(\mathbf{z}_j)$ in fig. 1 from both eigen (black triangles) and Cholesky (black circles) square-root matrices $\mathbf{S}$ from eqs. 12 and 15, respectively. Clearly each type of $\mathbf{S}$ leads to $2^2 = 4$ discrete locations on the $\sqrt{2}$-sigma contour. These lie inside the 1.514-sigma contour within which is approximately 68.2% of the pdf region. To see this, we form the χ^2 cdf for $N = 2$

$$\begin{aligned} c_{\chi^2}(y|2) &= \int_0^y p_{\chi^2}(x|2)\, dx, \\ &= \frac{1}{2}\int_0^y e^{-x/2}\, dx, \\ &= 1 - e^{-y/2}, \end{aligned} \tag{31}$$

which is set to q and inverted for y. This leads to $y = -2\ln(1-q)$. For 68.2% we get a corresponding $y = 2.291$ such that $\sqrt{y} = 1.514$.

4.3.2 Gaussian clones

If we evaluate p at the locations of the clones $\mathbf{z}_j = \mathbf{c} + \mathbf{R}^{-1}\mathbf{t}_j$ generated using $\mathbf{t}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, then

$$p(\mathbf{z}_j) = \frac{1}{\sqrt{(2\pi)^N \det(\mathbf{C})}} e^{-y/2}, \tag{32}$$

where y follows a χ^2 distribution and is not restricted to a single sigma contour. Note that this is true regardless of the form of the square-root matrix $\mathbf{S}$ that is used, e.g., eqs. 12 or 15.

We used the same $\mathbf{c}$ and $\mathbf{C}$ as in the Rademacher case and generated $K = 5000$ clones $\mathbf{z}_j$ using Gaussian $\mathbf{t}_j$ and have plotted their locations on top of the contours of $p(\mathbf{z}_j)$ in fig. 2 for the Cholesky square-root $\mathbf{S}$ matrix case. The samples for these types of clones are spread across the space with densities in accordance with the χ^2 CRs. In fact, of the 5000 clones generated, 3406 lie within the 1.514-sigma contour, which is 68.12% of the samples and is very close to the expected 68.2% value. An eigen square-root matrix approach would have shown similar densities, but not exactly the same locations.

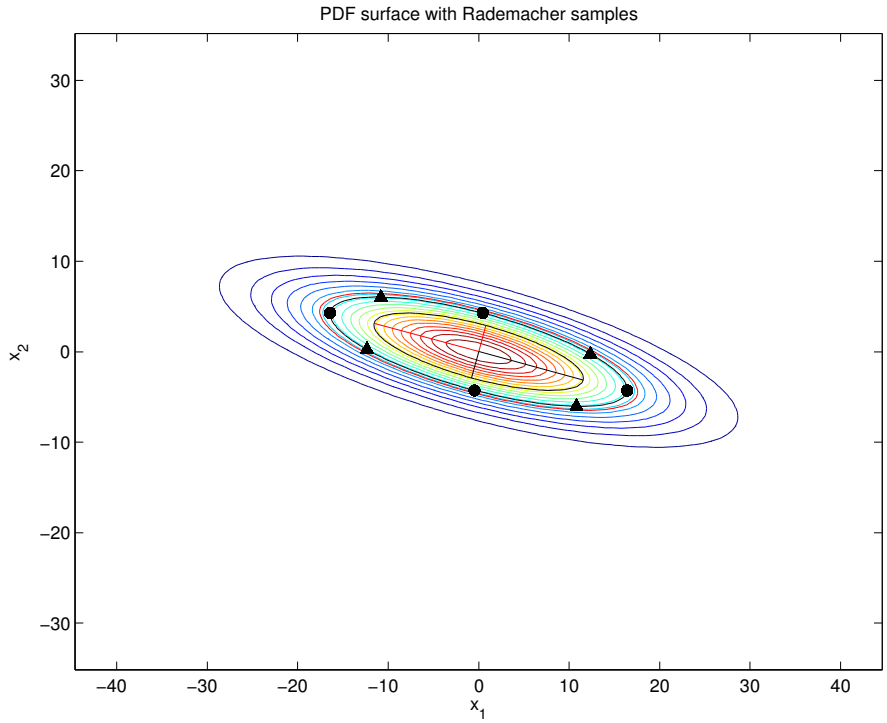

Figure 1: Gaussian pdf contours for $N = 2$ and the locations of $K = 5000$ clones generated from an eigen (black triangles) and a Cholesky (black circles) square-root matrix $\mathbf{S}$ both using $\mathbf{t}_j$ from a Rademacher distribution. Note that each set can occupy at most $2^2 = 4$ discrete locations on the $\sqrt{2}$-sigma contour (black). The larger red contour represents the 1.514-sigma contour within which is approximately 68.2% of the pdf region. The perpendicular lines represent the two principal axes of the 2×2 covariance matrix $\mathbf{C}$ scaled with the square-roots of their associated eigenvalues (12 and 3) and intersecting the 1-sigma contour.

5 Accuracy and ensemble size

A natural question arises as to how large of an ensemble size K is required to achieve a desired accuracy when using the Monte Carlo approach. As discussed in Section 4.2, the question is essentially how accurately the trace of a symmetric, positive semi-definite matrix $\mathbf{A}$ can be computed. The variance of the estimate, α, when computed using a single random sample has been given in eqs. 19 and 20, depending on the underlying distribution. If the estimate from the j-th sample is given by $\alpha_j = \mathbf{t}_j^{\mathrm{T}} \mathbf{A} \mathbf{t}_j$, then the Monte Carlo approximation is given by

$$\tilde{\alpha} = \frac{1}{K} \sum_{j=1}^{K} \alpha_j. \tag{33}$$

Because $\mathbf{t}_i$ and $\mathbf{t}_j$ are independent for $i \neq j$, then $E[\alpha_i \alpha_j] = E[\alpha_i] E[\alpha_j]$ and the covariance between α_i and α_j vanishes. This leads to the variance of the Monte Carlo approximation given by

$$V[\tilde{\alpha}] = \frac{1}{K} V[\alpha], \tag{34}$$

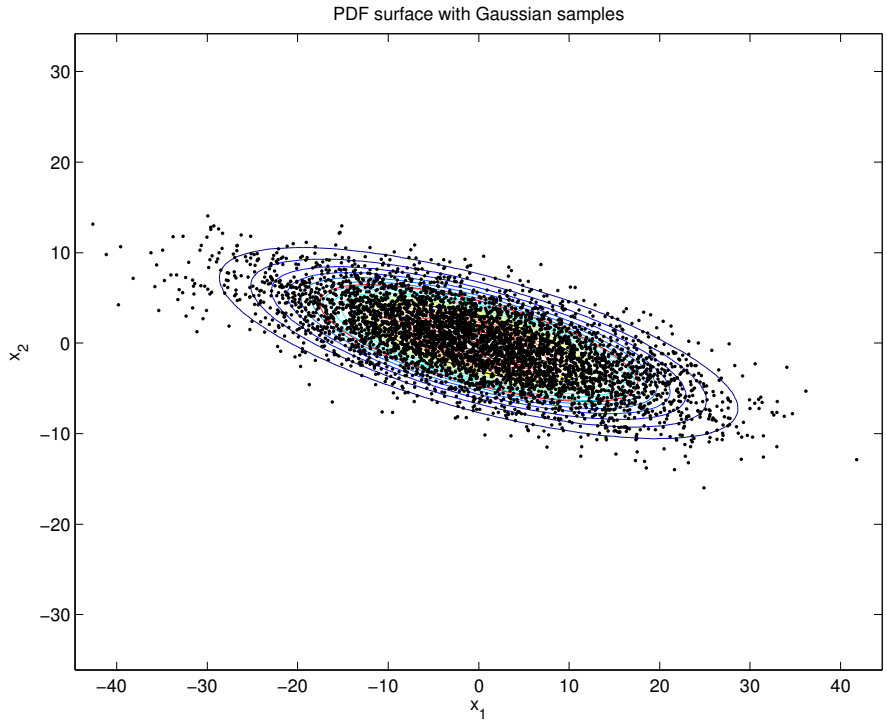

Figure 2: Gaussian pdf contours for $N = 2$ and the locations (black dots) of $K = 5000$ clones generated from a Cholesky square-root matrix $\mathbf{S}$ using $\mathbf{t}_j$ from a Gaussian distribution. The red contour represents the 1.514-sigma contour within which is approximately 68.2% of the pdf region. The perpendicular lines represent the two principal axes of the 2×2 covariance matrix $\mathbf{C}$ scaled with the square-roots of their associated eigenvalues (12 and 3) and intersecting the 1-sigma contour.

where $V[\alpha]$ is from either eq. 19 or 20. Thus, as expected, the error in the Monte Carlo approximation is seen to decrease as $1/\sqrt{K}$ with ensemble size K. Note that $K-1$ is used instead of K in eq. 6 in order to obtain an unbiased estimate since the ensemble members $\mathbf{z}_j$ are used to compute $\bar{\mathbf{z}}$ in eq. 7.

A practical way of determining the size of the ensemble is to simply check the differences in global grids of, say, $\sqrt{\alpha}$ made with different numbers of clones to see if the increase in K is warranted. Unfortunately, this still does not provide an absolute level for the error in $\sqrt{\alpha}$. However, given the fact that $\mathbf{A}$ has a rank of unity, the following property of the Frobenius norm is apparent

$$\begin{aligned} \|\mathbf{A}\|_F^2 &= Tr\left[\mathbf{A}^{\mathrm{T}}\mathbf{A}\right], \\ &= Tr\left[\mathbf{S}^{\mathrm{T}}\mathbf{f}\ \mathbf{f}^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{f}\ \mathbf{f}^{\mathrm{T}}\mathbf{S}\right], \\ &= Tr\left[\mathbf{f}^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{f}\ \mathbf{f}^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{f}\right], \\ &= \left(\mathbf{f}^{\mathrm{T}}\mathbf{S}\mathbf{S}^{\mathrm{T}}\mathbf{f}\right)^2, \\ &\approx \tilde{\alpha}^2. \end{aligned} \tag{35}$$

This means that the variance on the Monte Carlo estimate $\tilde{\alpha}_g$ made from a Gaussian ensemble can be approximately expressed as

$$V[\tilde{\alpha}_g] \approx \frac{2\tilde{\alpha}_g^2}{K}. \tag{36}$$

Typically, one works with grids of $\sqrt{\alpha}$, and therefore, wants to know the numerical error in these grids. To this end, let $\sigma[\cdot] \equiv \sqrt{V[\cdot]}$, which represents the error in the argument. Thus, to first order

$$\begin{aligned} \sigma\left[\sqrt{\tilde{\alpha}_g}\right] &= \frac{1}{2\sqrt{\tilde{\alpha}}}\sigma\left[\tilde{\alpha}_g\right], \\ &= \sqrt{\frac{\tilde{\alpha}_g}{2K}}, \end{aligned} \tag{37}$$

which gives the error in $\sqrt{\tilde{\alpha}_g}$ at a particular location. However, since there is only a single sample of $\tilde{\alpha}_g$, this quantity can significantly fluctuate from its true value. Therefore, a more stable measure of numerical error is seen in the root-mean-square (RMS) error over the grid. If the RMS of a grid of scalar x values having N points is defined as

$$RMS[x] = \sqrt{\frac{1}{N}\sum_{j=1}^{N} x_j^2}, \tag{38}$$

then its RMS error (RMSE) is given by $RMSE[x] = RMS[\sigma[x]]$. This leads to

$$RMSE\left[\sqrt{\tilde{\alpha}_g}\right] = \frac{1}{\sqrt{2K}} RMS\left[\sqrt{\tilde{\alpha}_g}\right]. \tag{39}$$

Thus, an absolute global error level can be obtained for a Gaussian Monte Carlo approximation $\sqrt{\tilde{\alpha}_g}$ in terms of known, available quantities. A relative RMSE (RRMSE) can also be defined as

$$\begin{aligned} RRMSE\left[\sqrt{\tilde{\alpha}_g}\right] &= \frac{RMSE\left[\sqrt{\tilde{\alpha}_g}\right]}{RMS\left[\sqrt{\tilde{\alpha}_g}\right]}, \\ &= \frac{1}{\sqrt{2K}}, \end{aligned} \tag{40}$$

which only depends on the ensemble size K.

An alternative to using a visual convergence criteria for ensemble size is to compare the global RMS of the difference between two grids generated with different ensemble sizes K and L, where $K < L$ and both share the same K clones, with the predicted value. To this end we seek an expression for $RMSE\left[\sqrt{\tilde{\alpha}_K} - \sqrt{\tilde{\alpha}_L}\right]$, where $\sqrt{\tilde{\alpha}_K}$ and $\sqrt{\tilde{\alpha}_L}$ are the Gaussian Monte Carlo estimates using ensemble sizes K and L, respectively. From eqs. 33 and 34 we see that the covariance between $\sqrt{\tilde{\alpha}_K}$ and $\sqrt{\tilde{\alpha}_L}$ is equal to $V[\tilde{\alpha}_L]$, which leads to

$$\begin{aligned} V\left[\sqrt{\tilde{\alpha}_K} - \sqrt{\tilde{\alpha}_L}\right] &= \begin{pmatrix} \frac{1}{2\sqrt{\tilde{\alpha}_K}} & \frac{-1}{2\sqrt{\tilde{\alpha}_L}} \end{pmatrix} \begin{pmatrix} V[\tilde{\alpha}_K] & V[\tilde{\alpha}_L] \\ V[\tilde{\alpha}_L] & V[\tilde{\alpha}_L] \end{pmatrix} \begin{pmatrix} \frac{1}{2\sqrt{\tilde{\alpha}_K}} \\ \frac{-1}{2\sqrt{\tilde{\alpha}_L}} \end{pmatrix}, \\ &= \frac{1}{2K}\tilde{\alpha}_K + \frac{1}{2L}\tilde{\alpha}_L - \frac{1}{L}\frac{\tilde{\alpha}_L^2}{\sqrt{\tilde{\alpha}_K}\sqrt{\tilde{\alpha}_L}}, \\ &\approx \left(\frac{1}{2K} - \frac{1}{2L}\right)\tilde{\alpha}_L, \end{aligned} \tag{41}$$

where the approximation comes from setting $\tilde{\alpha}_K = \tilde{\alpha}_L$. This leads to

$$RMSE\left[\sqrt{\tilde{\alpha}_K} - \sqrt{\tilde{\alpha}_L}\right] = \sqrt{\frac{1}{2K} - \frac{1}{2L}}\, RMS\left[\sqrt{\tilde{\alpha}_L}\right]. \tag{42}$$

Thus, $RMSE\left[\sqrt{\tilde{\alpha}_K} - \sqrt{\tilde{\alpha}_L}\right]$ can be computed directly from two maps and compared with eq. 42 to check the global convergence of the ensemble.

As for a Rademacher Monte Carlo approximation $\sqrt{\tilde{\alpha}_r}$, the situation is not quite as clear since $\|\hat{\mathbf{A}}\|_F^2$ in eq. 20 is not as easily expressible in known quantities. One practical approach is to use the Gaussian error bounds in eq. 39 since these will be larger than the actual Rademacher bounds, and this is what we do in the remainder of this study.

6 Applications

As an application of the clone ensemble approach in determining various error measurements, we have computed $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ on the 1738 km lunar reference sphere for the SGM150J model of *Goossens et al.* (2011) to d/o 140 in three different ways shown in fig. 3. The top map shows the quantity computed directly from the error-covariance matrix of the SGM150J base model, the middle map is the same quantity determined by a Monte Carlo simulation of $K = 100$ using clones defined in eq. 21, and the bottom map with $K = 8192$. The agreement is respectable for the ensemble size and shows the expected increase in uncertainty on the far side of the moon where there were no direct measurements and a decrease on the nearside where historical data were available. It can be seen that with an increase in ensemble size the Monte Carlo maps are converging to the top map.

To quantify this, plots were made of the true $RMSE\left[\sqrt{V\left[\Delta g(\mathbf{r})\right]}\right]$ using both Gaussian and Rademacher clones versus the approximation from eq. 39, which uses Gaussian clones, for this model and are shown in fig. 4. The true errors for both types of clones are about the same over the range of K, while the approximation converges to the true values before $K = 100$. From the dotted lines on this plot we can see that the global $RMSE$ of the middle and bottom maps in fig. 3 are approximately 2.140 and 0.227 mGal, respectively, with a ratio of $(2.140/0.227) = 9.43$ versus the ensemble sizes of $K = 100$ and 8192, respectively, with a ratio of $\sqrt{8192/100} = 9.05$, which is roughly in keeping with the $1/\sqrt{K}$ decay in error indicated by eq. 39. The RRMSE values for $K = 100$ and 8192 are 7.1% and 0.78%, respectively. The true RMS is 29.01 mGal.

In addition, we computed $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ on the 1738 km lunar reference sphere from an ensemble of $K = 1000$ for the full d/o 900 GRGM900C model of GSFC (*Lemoine et al.*, 2014), which is shown in the top map of fig. 5. We also show a minimum spacecraft altitude map during the primary and extended missions derived from lunar topography obtained by the Lunar Orbiter Laser Altimeter (LOLA) on the Lunar Reconnaissance Orbiter (LRO) (*Smith et al.*, 2010) in the bottom of the same figure. One can clearly see a significant correlation of high error levels with high minimum altitude regions in the proximity of the 0°E/180°E meridians. One can also see lower error levels in the polar regions, where presumably the higher track density is having an effect.

A plot was made of the approximate $RMSE\left[\sqrt{V\left[\Delta g(\mathbf{r})\right]}\right]$ from eq. 39, which uses Gaussian clones, for this model and is shown in fig. 6. From the dotted lines on this plot we can see that the global

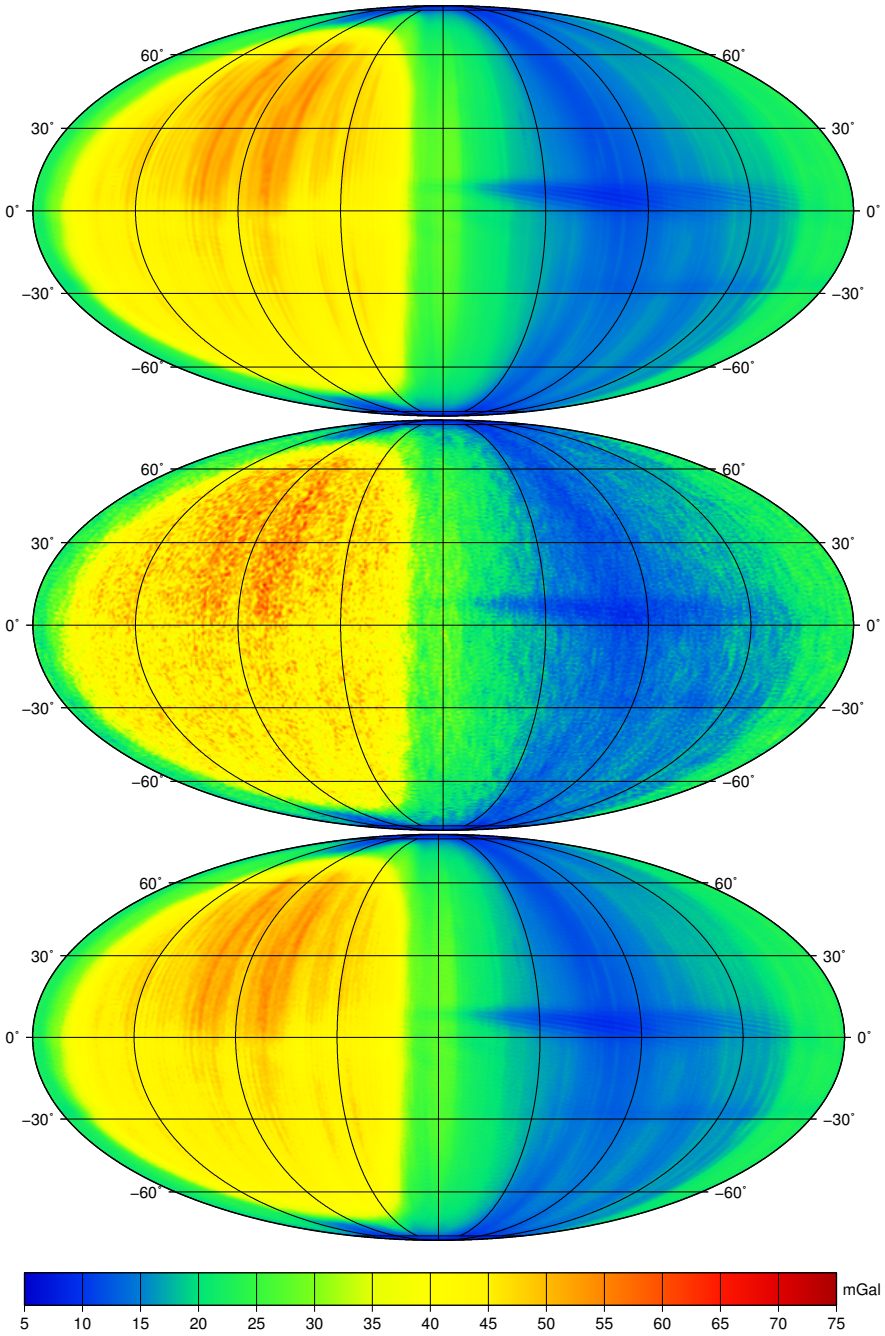

Figure 3: Maps of $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ on the 1738 km lunar reference sphere for the SGM150J model of *Goossens et al.* (2011) to d/o 140 computed directly from the error-covariance matrix of the base model (eq. 2, top), from a Monte Carlo simulation (eq. 6) of $K = 100$ using clones defined in eq. 21 (middle), and from a simulation with $K = 8192$ (bottom). Longitude 270°E is the center meridian of the maps.

$RMSE$ of the top map in fig. 5 is approximately 0.717 mGal at $K = 1000$. The RRMSE value for $K = 1000$ is 2.24%. The approximate RMS is 32.07 mGal.

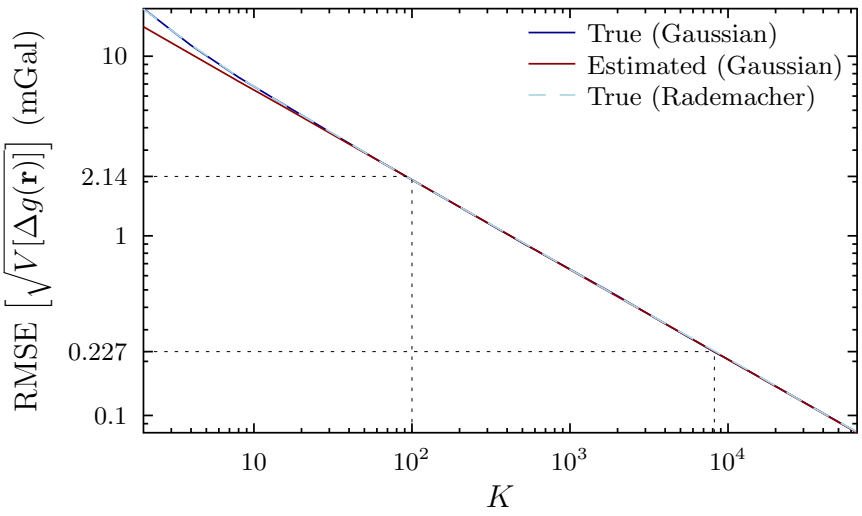

Figure 4: Plots of true $RMSE\left[\sqrt{V\left[\Delta g(\mathbf{r})\right]}\right]$ using both Gaussian (dark blue solid) and Rademacher (light blue dashed) clones and the approximation from eq. 39, which uses Gaussian clones (dark red solid), versus ensemble size K for the SGM150J model of *Goossens et al.* (2011) to d/o 140. The dotted lines show the values at $K = 100$ and 8192 that correspond to the middle and bottom maps in fig. 3, respectively.

The Monte Carlo generated maps of $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ from the SGM150J and GRGM900C models highlight a possible caveat that exists when determining a sufficient ensemble size K by comparing maps. In fig. 7 we show the difference between the $K = 8192$ and the true maps for SGM150J at the top, and the difference between the $K = 1000$ and 5000 maps of GRGM900C at the bottom. As expected, the largest variation in both maps occur where $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ is largest. However, when comparing sequences of maps of increasing K it appears to us that a much smaller ensemble size is needed for convergence with GRGM900C versus SGM150J, but this is due to the distribution of numerical error in the maps. In fact, the largest differences for SGM150J and GRGM900C are 1.581 and 4.956 mGal, respectively, even though GRGM900C maps appear to us to have converged. In this case, the large errors appear as isolated pinnacles amongst a lower background error and are difficult to detect by eye. To see this, we have expanded the outlined magenta box in the bottom map of fig. 7, which is shown in fig. 8. The box contains the largest error point, which is located at 179.875°E/24.375°N. This suggests that rather than using a visual comparison of maps to determine ensemble size K, one should consider using the RMSE of the grids.

Finally, as to the computational savings gained by using the Monte Carlo approach, consider that if we evaluate $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ at N points from a covariance square-root of a model with M parameters, then to first order the operation count goes as $\mathcal{O}(\frac{1}{2}NM^2)$, but using a Monte Carlo approach with K clones goes as $\mathcal{O}(\frac{1}{2}KM^2 + NMK)$. The speedup factor is then

$$\xi = \frac{N}{K\left(2\frac{N}{M} + 1\right)} \tag{43}$$

when using eqs. 6 and 21 over eq. 2. If the N points form a quadrature grid of approximately $2L_{max}^2$ points, where L_{max} is the spherical harmonic degree truncation level of the gravity field

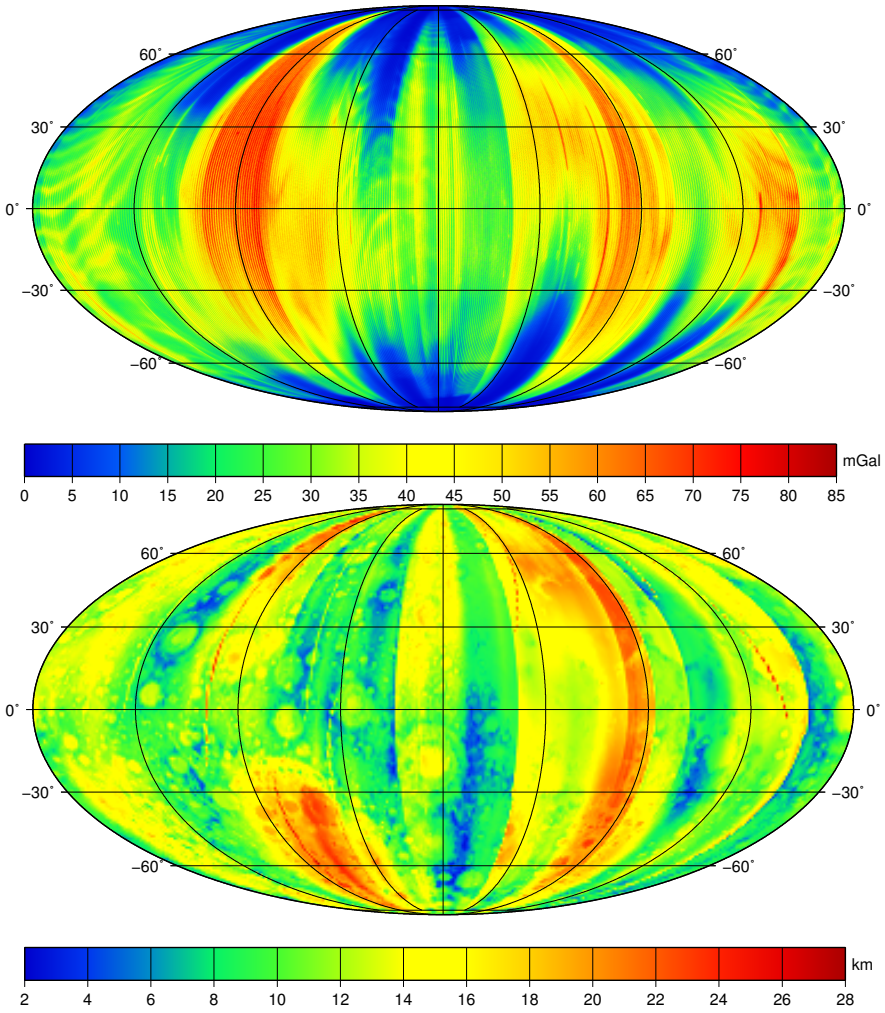

Figure 5: Map of $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ on the 1738 km lunar reference sphere for the full d/o 900 GRGM900C model of GSFC (*Lemoine et al.*, 2014) computed from a Monte Carlo simulation of $K = 1000$ using clones defined in eq. 21 (top) and map of minimum spacecraft altitude during the primary and extended missions derived from the lunar topography obtained from the data of the Lunar Orbiter Laser Altimeter (LOLA) on the Lunar Reconnaissance Orbiter (LRO) (*Smith et al.*, 2010). Longitude 270°E is the center meridian of both maps.

model, then eq. 43 becomes

$$\xi = \frac{2}{5}\frac{L_{max}^2}{K}. \tag{44}$$

For a d/o 900 model with $K = 1000$ we have $\xi \approx 324$.

7 Conclusions

The GSFC GRAIL gravity team has developed a procedure to greatly expedite error propagation of its gravity models based upon Monte Carlo approximation. It has already been used success-

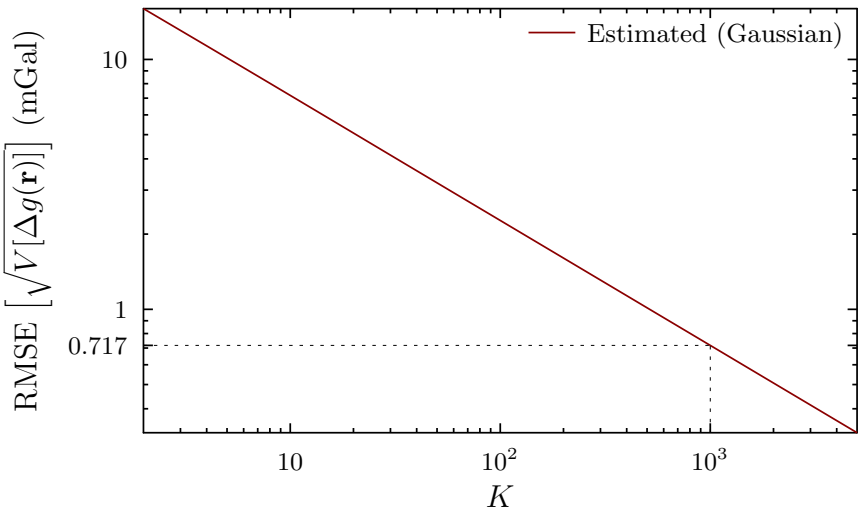

Figure 6: Plot of the approximate $RMSE\left[\sqrt{V\left[\Delta g(\mathbf{r})\right]}\right]$ from eq. 39, which uses Gaussian clones, versus ensemble size K for the GRGM900C model of GSFC (*Lemoine et al.*, 2014) to d/o 900. The dotted lines show the value at $K = 1000$ that corresponds to the top map in fig. 5.

fully for generating degree-strength maps of the Jet Propulsion Laboratory (JPL) d/o 900 gravity models GL0900C and GL0900D (*Konopliv et al.*, 2014) and the GSFC d/o 900 gravity model GRGM900C (*Lemoine et al.*, 2014). The basis of the approximation is the generation of an ensemble of "clone" models that are samples drawn from the error distribution of the particular base model of interest, which are of interest in their own right. These will be made available to the general community.

A Appendix

The purpose of this Appendix is to derive eqs. 19 and 20 from eq. 18.

A.1 Gaussian case

Let us first consider the Gaussian case where $\mathbf{t} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Clearly

$$\begin{aligned} E[\alpha_g] &= E\left[\mathbf{t}^{\mathrm{T}}\mathbf{A}\mathbf{t}\right], \\ &= Tr\left[\mathbf{A}E\left[\mathbf{t}\mathbf{t}^{\mathrm{T}}\right]\right], \\ &= Tr\left[\mathbf{A}\right]. \end{aligned} \tag{45}$$

Now in order to derive $V[\alpha_g]$ we need to evaluate $E\left[\alpha_g^2\right]$ which requires the first four moments of the distribution. It is well known that for a scalar standardized Gaussian random variable t the

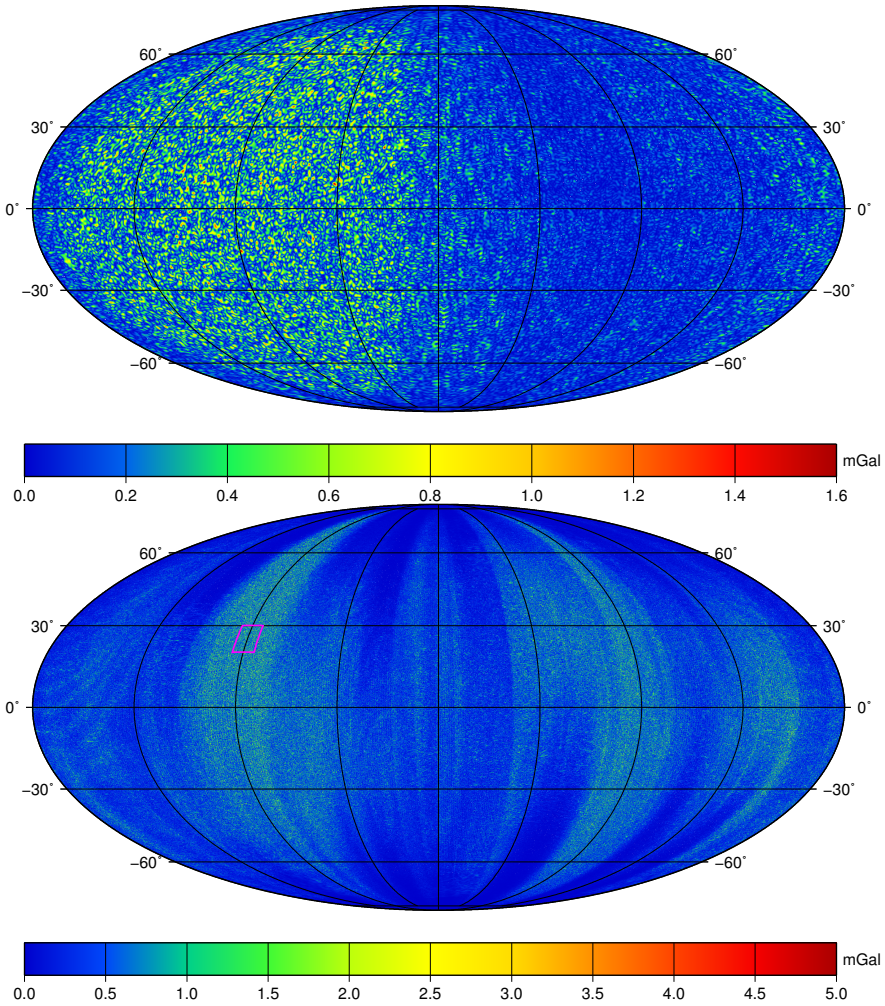

Figure 7: Maps of differences in $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ between the $K = 8192$ Monte Carlo approximation and the true grid for the SGM150J model of *Goossens et al.* (2011) (top), and between the $K = 1000$ and 5000 Monte Carlo approximations for the GRGM900C model of GSFC (*Lemoine et al.*, 2014). Longitude 270°E is the center meridian of both maps.

first four moments are given by

$$E\left[t^n\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^n e^{-t^2/2}\, dt = \begin{cases} 0, & n = 1 \\ 1, & n = 2 \\ 0, & n = 3 \\ 3, & n = 4 \end{cases} . \tag{46}$$

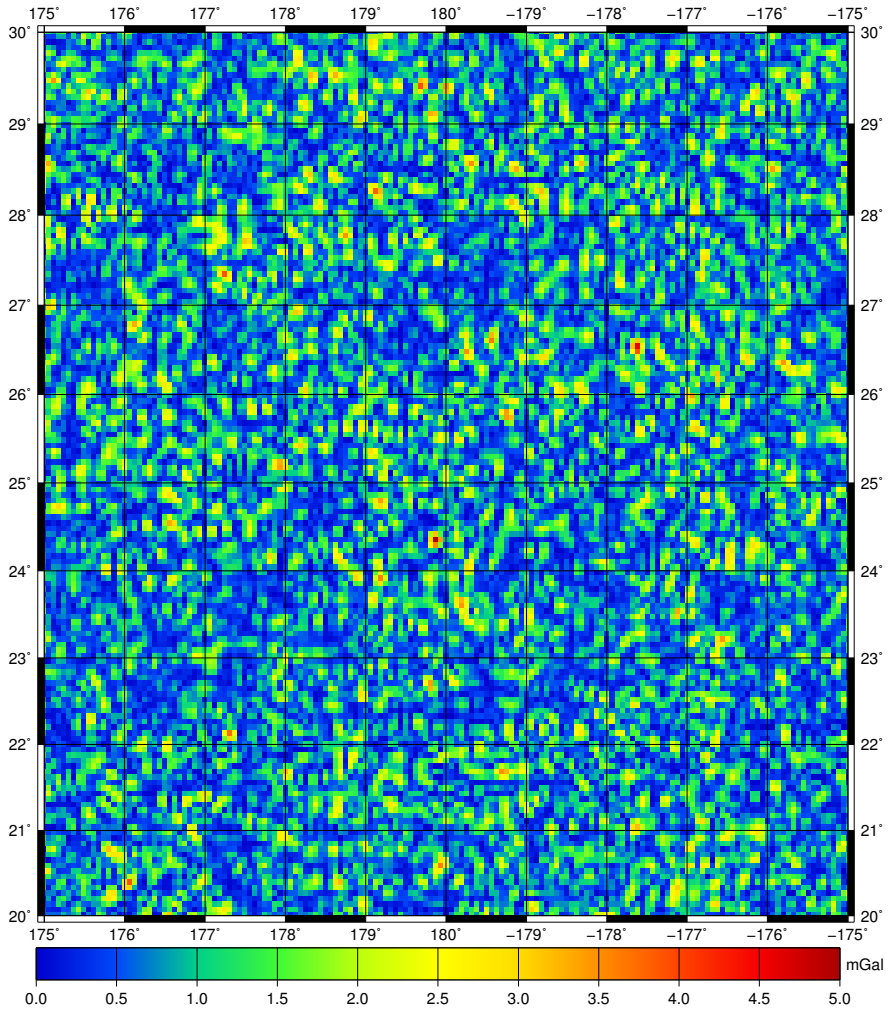

Figure 8: Map of difference in $\sqrt{V\left[\Delta g(\mathbf{r})\right]}$ between the $K = 1000$ and 5000 Monte Carlo approximations for the GRGM900C model of GSFC (*Lemoine et al.*, 2014) over the region outlined by the magenta box in the bottom map in fig. 7. The range of the box is longitude 175°E − 175°W and latitude 20° − 30°N. The largest difference is located at 179.875°E/24.375°N.

This leads to

$$\begin{aligned}
E\left[\alpha_g^2\right] &= E\left[\mathbf{t}^{\mathrm{T}}\mathbf{A}\mathbf{t}\mathbf{t}^{\mathrm{T}}\mathbf{A}\mathbf{t}\right], \\
&= E\left[\left(\sum_i\sum_j A_{ij}t_it_j\right)\left(\sum_k\sum_\ell A_{k\ell}t_kt_\ell\right)\right], \\
&= \sum_i\sum_j\sum_k\sum_\ell A_{ij}A_{k\ell}E\left[t_it_jt_kt_\ell\right], \\
&= \sum_i\sum_k\sum_\ell A_{ii}A_{k\ell}E\left[t_i^2t_kt_\ell\right] + \sum_i\sum_{j\neq i}\sum_k\sum_\ell A_{ij}A_{k\ell}E\left[t_it_jt_kt_\ell\right], \\
&= \sum_i\sum_k A_{ii}A_{kk}E\left[t_i^2t_k^2\right] + \sum_i\sum_{j\neq i}\left(A_{ij}^2 + A_{ij}A_{ji}\right)E\left[t_i^2t_j^2\right], \\
&= \sum_i A_{ii}^2E\left[t_i^4\right] + \sum_i\sum_{k\neq i}A_{ii}A_{kk}E\left[t_i^2t_k^2\right] + \sum_i\sum_{j\neq i}\left(A_{ij}^2 + A_{ij}A_{ji}\right)E\left[t_i^2t_j^2\right], \qquad (47) \\
&= 2\sum_i A_{ii}^2 + \sum_i\sum_k A_{ii}A_{kk} + \sum_i\sum_{j\neq i}\left(A_{ij}^2 + A_{ij}A_{ji}\right), \\
&= \sum_i\sum_k A_{ii}A_{kk} + \sum_i\sum_j\left(A_{ij}^2 + A_{ij}A_{ji}\right), \\
&= Tr\left[\mathbf{A}\right]^2 + Tr\left[\mathbf{A}^{\mathrm{T}}\mathbf{A}\right] + Tr\left[\mathbf{A}\mathbf{A}\right]. \qquad (48)
\end{aligned}$$

However, we are dealing with symmetric $\mathbf{A}$ matrices, and so eq. 48 reduces to

$$\begin{aligned}
E\left[\alpha_g^2\right] &= Tr\left[\mathbf{A}\right]^2 + 2Tr\left[\mathbf{A}^{\mathrm{T}}\mathbf{A}\right], \\
&= 2\|\mathbf{A}\|_F^2 + Tr\left[\mathbf{A}\right]^2. \qquad (49)
\end{aligned}$$

Finally

$$\begin{aligned}
V\left[\alpha_g\right] &= E\left[\alpha_g^2\right] - E\left[\alpha_g\right]^2, \\
&= 2\|\mathbf{A}\|_F^2 + Tr\left[\mathbf{A}\right]^2 - Tr\left[\mathbf{A}\right]^2, \\
&= 2\|\mathbf{A}\|_F^2, \qquad (50)
\end{aligned}$$

which is eq. 19.

A.2 Rademacher case

The derivation for the Rademacher case follows closely that of the Gaussian case. The only difference lies in the fourth moment of the distribution. For a scalar standardized Rademacher random variable t that has a probability p of being $+1$ and probability $(1-p)$ of being -1, the first four moments are given by

$$E\left[t^n\right] = (-1)^n\cdot(1-p) + (1)^n\cdot p = \begin{cases} 2p-1, & n=1 \\ 1, & n=2 \\ 2p-1, & n=3 \\ 1, & n=4 \end{cases}. \qquad (51)$$

For the symmetric distribution there is equal probability of both states, i.e., $p = 1/2$, which makes the first and third moments 0. Thus, the fourth moment is 1 for Rademacher, but 3 for Gaussian. Clearly

$$E\left[\alpha_r\right] = Tr\left[\mathbf{A}\right], \tag{52}$$

and the derivation of $E\left[\alpha_r^2\right]$ exactly follows that of $E\left[\alpha_g^2\right]$ down to eq. 47 where we now begin

$$\begin{aligned} E\left[\alpha_r^2\right] &= \sum_i A_{ii}^2 E\left[t_i^4\right] + \sum_i \sum_{k\neq i} A_{ii}A_{kk} E\left[t_i^2 t_k^2\right] + \sum_i \sum_{j\neq i} \left(A_{ij}^2 + A_{ij}A_{ji}\right) E\left[t_i^2 t_j^2\right], \\ &= \sum_i \sum_k A_{ii}A_{kk} + \sum_i \sum_{j\neq i} \left(A_{ij}^2 + A_{ij}A_{ji}\right), \\ &= \sum_i \sum_k A_{ii}A_{kk} + \sum_i \sum_j \left(\hat{A}_{ij}^2 + \hat{A}_{ij}\hat{A}_{ji}\right), \\ &= Tr\left[\mathbf{A}\right]^2 + Tr\left[\hat{\mathbf{A}}^{\mathrm{T}}\hat{\mathbf{A}}\right] + Tr\left[\hat{\mathbf{A}}\hat{\mathbf{A}}\right]. \end{aligned} \tag{53}$$

However, we are dealing with symmetric $\hat{\mathbf{A}}$ matrices, and so eq. 53 reduces to

$$\begin{aligned} E\left[\alpha_r^2\right] &= Tr\left[\mathbf{A}\right]^2 + 2Tr\left[\hat{\mathbf{A}}^{\mathrm{T}}\hat{\mathbf{A}}\right], \\ &= 2\|\hat{\mathbf{A}}\|_F^2 + Tr\left[\mathbf{A}\right]^2. \end{aligned} \tag{54}$$

Recall that $\hat{\mathbf{A}}$ is the same as $\mathbf{A}$, but with a main diagonal of zeros. Finally

$$\begin{aligned} V\left[\alpha_r\right] &= E\left[\alpha_r^2\right] - E\left[\alpha_r\right]^2, \\ &= 2\|\hat{\mathbf{A}}\|_F^2 + Tr\left[\mathbf{A}\right]^2 - Tr\left[\mathbf{A}\right]^2, \\ &= 2\|\hat{\mathbf{A}}\|_F^2, \end{aligned} \tag{55}$$

which is eq. 20.

References

Asmar, S. W., et al. (2013), The scientific measurement system of the Gravity Recovery and Interior Laboratory (GRAIL) Mission, *Space Sci. Rev.*, doi:10.1007/s11214-013-9962-0.

Goossens, S. J., et al. (2011), Improved high-resolution lunar gravity field model from SELENE and historical tracking data, American Geophysical Union Fall Meeting, San Francisco, CA, Abstract P44B-05.

Hutchinson, M. F. (1989), A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines, *Communications in Statistics, Simulation and Computation*, *18*, 1059–1076.

Konopliv, A. S., et al. (2013), The JPL lunar gravity field to spherical harmonic degree 660 from the GRAIL primary mission, *J. Geophys. Res.*, *118*, 1415–1434, doi:10.1002/jgre.20097.

Konopliv, A. S., et al. (2014), High resolution lunar gravity fields from the GRAIL primary and extended mission, *Geophys. Res. Lett.*, doi:10.1002/2013GL059066, accepted.

Lemoine, F. G., et al. (2013), High-degree gravity models from GRAIL data, *J. Geophys.Res.*, *118*, 16761698, doi:10.1002/jgre.20118.

Lemoine, F. G., et al. (2014), A degree 900 lunar gravity model from GRAIL mission data, in preparation.

Smith, D. E., et al. (2010), Initial observations from the Lunar Orbiter Laser Altimeter (LOLA), *Geophys. Res. Lett.*, *37*(L18204), doi:10.1029/2010GL043751.

Smith, D. E., et al. (2012), Gravity field and internal structure of Mercury from MESSENGER, *Science*, *336*, 214–, doi:10.1126/science.1218809.

Zuber, M. T., D. E. Smith, D. H. Lehman, T. L. Hoffman, S. W. Asmar, and M. M. Watkins (2013), Gravity Recovery and Interior Laboratory (GRAIL): Mapping the lunar interior from crust to core, *Space Sci. Rev.*, doi:10.1007/s11214-012-9952-7.