Journal of Geodesy (2004) 78: 12–33
DOI 10.1007/s00190-003-0362-1

Journal of Geodesy

A technique for modeling the Earth's gravity field on the basis of satellite accelerations

P. Ditmar, A. A. van Eck van der Sluijs

Physical, Geometrical and Space Geodesy (FMR), Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands; e-mail: ditmar@geo.tudelft.nl; Tel.: +31-15-2782501; Fax: +31-15-2783711

Received: 7 April 2003 / Accepted: 20 October 2003 / Published online: 21 June 2004

Abstract. A technique is proposed for Earth's gravity field modeling on the basis of satellite accelerations that are derived from precise orbit data. The functional model rests on Newton's second law. The computational procedure is based on the pre-conditioned conjugate-gradient (PCCG) method. The data are treated as weighted average accelerations rather than as point-wise ones. As a result, a simple three-point numerical differentiation scheme can be used to derive them. Noise in the orbit-derived accelerations is strongly dependent on frequency. Therefore, the key element of the proposed technique is frequency-dependent data weighting. Fast convergence of the PCCG procedure is ensured by a block-diagonal pre-conditioner (approximation of the normal matrix), which is derived under the so-called Colombo assumptions. Both uninterrupted data sets and data with gaps can be handled. The developed technique is compared with other approaches: (1) the energy balance approach (based on the energy conservation law) and (2) the traditional approach (based on the integration of variational equations). Theoretical considerations, supported by a numerical study, show that the proposed technique is more accurate than the energy balance approach and leads to approximately the same results as the traditional one. The former finding is explained by the fact that the energy balance approach is only sensitive to the along-track force component. Information about the cross-track and the radial component of the gravitational potential gradient is lost because the corresponding force components do no work and do not contribute to the energy balance. Furthermore, it is shown that the proposed technique is much (possibly, orders of magnitude) faster than the traditional one because it does not require the computation of the normal matrix. Hints are given on how the proposed technique can be adapted to the explicit assembling of the normal matrix if the latter is needed for the computation of the model covariance matrix.

Correspondence to: P. Ditmar

Key words: Earth's gravity field – Satellite-to-satellite tracking – Satellite accelerations – CHAMP – GOCE

1 Introduction

The compilation of a high-precision model of the Earth's gravity field is a very important task, with numerous scientific and societal applications including oceanography, geodesy, and solid-Earth physics. A number of dedicated satellite missions have been launched, or will be launched soon, in order to accomplish this task: CHAMP in 2000 (Reigber et al. 1996), GRACE in 2002 (Tapley 1997), and GOCE in 2006 [European Space Agency (ESA) 1999]. These missions exploit different measurement techniques but have a common feature: they all use global positioning system (GPS) receivers in order to collect so-called high–low satellite-to-satellite tracking (hl-SST) data. These data form the necessary input for a precise orbit determination (POD) procedure. The computed orbit contains information that can be used to improve the Earth's gravity field model, particularly in the range of low spatial frequencies. Naturally, there are a number of non-gravitational forces that also influence a satellite's motion. In order to take them into account, CHAMP and GRACE satellites are equipped with accelerometers that allow the contribution of non-gravitational forces to be subtracted at the data pre-processing stage. The GOCE satellite will be equipped with a drag-free control system, which is intended to compensate for non-gravitational forces in real time.

Two types of POD techniques for modeling of the Earth's gravity field exist: reduced-dynamic and kinematic techniques (Visser and van den IJssel 2000). In our opinion, both of these should be used. We argue that a clear distinction should be made between the *observations*, which form a data vector in the functional model, and *observation points*, which are the locations

of the observations made. The best source of observations is the kinematic orbit. The reduced-dynamic orbit would lead to a poorer estimation of the gravity field because an orbit of this type is biased towards the a priori model (Gerlach et al. 2003a). To clarify this statement, we recall that the forces acting on the satellite can be extracted explicitly from a satellite orbit by means of double differentiation. The second derivative of the reduced-dynamic orbit corresponds not to the actual gravity field but to the gravity field used in the POD procedure. Exceptions are the points where the artificial impulse forces have been applied in order to minimize the misfits between the computed orbit and the acquired data. The entire gravity field signal that is not explained by the model used is, therefore, concentrated at these occasional points. Obviously, such a 'compressed' representation of the signal may distort the data processing results. On the other hand, the kinematic orbit is contaminated by relatively strong noise and may occasionally be interrupted. Hence it is inferior for the definition of observation points. The reduced-dynamic orbit seems to be more appropriate for this purpose. In this way, we may reduce errors in positions and eliminate gaps. In view of the data processing technique we propose, the latter aspect is especially important.

There are a number of ways to compute the Earth's gravity field model from a satellite orbit. The most traditional approach formulates the functional model in terms of the orbit disturbances, which are computed as differences between the output of a POD procedure and a reference orbit obtained on the basis of a reference gravity field (Rowlands et al. 1995; Visser et al. 2001; Reigber et al. 2002). In order to compute the partial derivatives of orbit disturbances with respect to the gravity field parameters, the so-called variational equations have to be integrated. One drawback of this approach is the non-linearity of the functional model. In addition, this approach usually includes an explicit assembly of the normal matrix, which is a rather time-consuming procedure. These are the reasons why alternative approaches have attracted attention in the course of recent years. One of these is the energy balance approach, which is based on the energy conservation principle. The satellite orbit is used to compute the kinetic energy of the satellite, which can then be related directly to the gravitational potential at the given point (O'Keefe 1957; Jekeli 1999; Sneeuw et al. 2002; Han et al. 2002; Gerlach et al. 2003b; Howe et al. 2003). The other alternative is to transform the precise orbit data into satellite accelerations and then relate the latter to the Earth's gravity field according to Newton's second law. At first glance, the latter approach is the most natural. A number of authors have already used it for the determination of the gravity field of other planets (see e.g. Barriot and Balmino 1992). However, the application of this approach to high-precision modeling of the Earth's gravity field is still in its infancy and, in spite of a number of recent publications (Schäfer 200l; Reubelt et al. 2003a,b; Fengler et al. submitted) further efforts in this direction are undoubtedly needed.

In this article, we present a new technique for modeling of the Earth's gravity field on the basis of satellite accelerations. It is somewhat similar to those designed earlier for processing of satellite gravity gradiometry (SGG) data (Schuh 1996; Ditmar and Klees 2002; Ditmar et al. 2003a; Klees et al. in press). The proposed technique is based on the pre-conditioned conjugate-gradient (PCCG) method. The issues deserving of our special attention are: (1) the relation of orbit-derived accelerations, which may differ from point-wise (instantaneous) ones, to the gravity field parameters; (2) a frequency-dependent data weighting, which is needed to suppress high-frequency noise produced by the double differentiation; and (3) efficient pre-conditioning aimed at reducing the number of PCCG iterations. Importantly, the developed technique can be applied to observations with gaps. It should be added that the concept of frequency-dependent weighting was originally developed just in the context of SGG data (Schuh 1996; Klees et al. 2003; Klees and Ditmar in press), where frequency-dependent noise may also be a problem.

An advantage of the PCCG method is that it does not require the normal matrix, thanks to which the data processing becomes very fast. It may be argued that the normal matrix should be computed anyway, for example because it is needed for generating the covariance matrix of the estimated parameters. We believe, however, that the gravity field modeling and the computation of the normal/covariance matrix should be considered as different tasks. With a fast gravity field modeling algorithm, we may try a whole variety of data processing strategies: different pre-processing schemes, different data weighting, different regularization, etc. The computation of the normal/covariance matrix is a much more time-consuming procedure. Hence it makes sense to start it only when the optimal data processing strategy is found and the final gravity field model is obtained. Application of the proposed methodology to the computation of the normal/covariance matrix is beyond the scope of this publication. However, we discuss this issue briefly in Sect. 5.

Is it important to add that the issues related to the regularization are also not covered by this publication. This topic has already been discussed extensively by Kusche and Klees (2002a, b) and Ditmar et al. (2003c) in the context of the SGG data processing. The conclusions drawn in these publications are also applicable to satellite accelerations.

The theoretical results presented in the paper are supported by a numerical study. The proposed technique is evaluated and compared with the energy balance approach as well as with the method based on the integration of variational equations.

With our publication, we try to debunk a myth that satellite accelerations cannot be used for building an accurate Earth's gravity field model because they are too noisy. We show, in particular, that three-component accelerations and three-component orbit data contain nearly the same information and lead to nearly the same optimal estimation. The only difference between these

quantities is that noise in orbit data can frequently be treated as white and, therefore, the data processing can be done without any data weighting. Noise in accelerations, on the contrary, strongly depends on frequency. Then, an accurate frequency-dependent data weighting is a must.

2 Functional model

2.1 Point-wise accelerations

A commonly used procedure to represent the Earth's gravitational potential $V(r, \theta, \lambda)$ is the expansion into a series of spherical harmonics (see e.g. Heiskanen and Moritz 1984)

$$V(r,\theta,\lambda) = \frac{GM_E}{R}\left\{\frac{R}{r} + \sum_{l=2}^{\infty}\left(\frac{R}{r}\right)^{l+1} \times \sum_{m=0}^{l}(C_{lm}\cos m\lambda + S_{lm}\sin m\lambda)\bar{P}_{lm}(\cos\theta)\right\} \tag{1}$$

where r, θ, λ are the spherical coordinates; G is the universal gravitational constant; M_E is the Earth's mass; R is the semi-major axis of a reference ellipsoid; C_{lm}, S_{lm} are the spherical harmonic coefficients; and $\bar{P}_{lm}(\cos\theta)$ are the fully normalized associated Legendre functions. Naturally, in practice the series of Eq. (l) is always truncated.

The goal of the orbit data processing is to improve the estimations of the spherical harmonic coefficients (C_{lm}, S_{lm}). Let us introduce the residual potential $T(r, \theta, \lambda)$ – a correction to be applied to the chosen reference field

$$T(r,\theta,\lambda) = \frac{GM_E}{R}\sum_{l=L_{\min}}^{L_{\max}}\left(\frac{R}{r}\right)^{l+1} \times \sum_{m=0}^{l}(\Delta C_{lm}\cos m\lambda + \Delta S_{lm}\sin m\lambda)\bar{P}_{lm}(\cos\theta) \tag{2}$$

with ΔC_{lm}, ΔS_{lm} the corrections of individual spherical harmonic coefficients. The contribution of the reference field should be subtracted from the data during the pre-processing. A reasonable choice of the maximum degree $L_{\max}$ is somewhere between 50 and 80 (or possibly a little higher if the satellite altitude is very low, as will be the case for the GOCE mission). The minimum degree $L_{\min}$ is usually set equal to 2: the zero-order coefficient can be better estimated on the basis of data from high-orbiting satellites (Ries et al. 1992), whereas the first-order terms vanish when the origin of the coordinate system coincides with the Earth's center of mass.

In order to obtain an explicit relationship between the corrections (ΔC_{lm}, ΔS_{lm}) and the (point-wise) residual satellite accelerations, let us introduce the geographical reference frame, i.e. a local right-handed Cartesian frame where the X-axis points to the north, the Y-axis to the west, and the Z-axis radially outwards. In this frame, the residual acceleration vector $\Delta\mathbf{a}^{(\mathbf{G})}$ can be related to the residual potential as

$$\Delta a_x^{(G)} = -\frac{1}{r}\frac{\partial T}{\partial\theta}, \quad \Delta a_y^{(G)} = -\frac{1}{r\sin\theta}\frac{\partial T}{\partial\lambda}, \quad \Delta a_z^{(G)} = \frac{\partial T}{\partial r} \tag{3}$$

where

$$\frac{\partial T}{\partial\theta} = -\frac{GM_E}{R}\sum_{l=L_{\min}}^{L_{\max}}\left(\frac{R}{r}\right)^{l+1} \times \sum_{m=0}^{l}\{\Delta C_{lm}\cos m\lambda + \Delta S_{lm}\sin m\lambda\}\bar{P}'_{lm}(\cos\theta)\sin\theta$$

$$\frac{\partial T}{\partial\lambda} = \frac{GM_E}{R}\sum_{l=L_{\min}}^{L_{\max}}\left(\frac{R}{r}\right)^{l+1} \times \sum_{m=0}^{l} m\{-\Delta C_{lm}\sin m\lambda + \Delta S_{lm}\cos m\lambda\}\bar{P}_{lm}(\cos\theta) \tag{4}$$

$$\frac{\partial T}{\partial r} = -\frac{GM_E}{R^2}\sum_{l=L_{\min}}^{L_{\max}}(l+1)\left(\frac{R}{r}\right)^{l+2} \times \sum_{m=0}^{l}\{\Delta C_{lm}\cos m\lambda + \Delta S_{lm}\sin m\lambda\}\bar{P}_{lm}(\cos\theta)$$

with

$$\bar{P}'_{lm}(x) = \frac{\mathrm{d}\bar{P}_{lm}(x)}{\mathrm{d}x}$$

Furthermore, we find it more practical to use for the definition of the functional model not the geographical but the so-called local orbital reference frame (LORF): the frame where the X-axis is directed along the track, the Y-axis coincides with the direction of the orbital angular momentum, and the Z-axis completes the frame as a right-handed one. The advantages of this frame are: (1) consistency with the orientation of the on-board accelerometer (CHAMP and GRACE missions) or gradiometer (GOCE mission), and (2) a nearly block-diagonal normal matrix (see Sect. 3.3 for more details). Transformation of the residual acceleration vector from the geographical frame into the LORF can be performed by a standard rotation

$$\Delta\mathbf{a}^{(\mathbf{L})} = \mathbf{R}_{\mathbf{y}}(-\alpha)\mathbf{R}_{\mathbf{z}}(-\beta)\Delta\mathbf{a}^{(\mathbf{G})} \tag{5}$$

where $\mathbf{R}_{\mathbf{z}}(-\beta)$ is the matrix of rotation around the Z-axis with β being the satellite track azimuth (counted from the north clock-wise), and $\mathbf{R}_{\mathbf{y}}(-\alpha)$ is the matrix of rotation around the Y-axis with α being the angle between the horizon and the satellite track (counted upwards).

2.2 Orbit-derived accelerations

All the expressions presented so far are valid for point-wise accelerations. On the other hand, we propose to derive satellite accelerations from a precise orbit according to a simple three-point scheme

$$\bar{a}(t) = \frac{x(t-\Delta t) - 2x(t) + x(t+\Delta t)}{(\Delta t)^2} \tag{6}$$

where $x(t)$ is a component of the satellite position vector at time t and Δt is the data sampling interval. In Sect. 5 we discuss the advantages of this approach over a multi-point scheme, which strives for accurate extraction of point-wise accelerations from a satellite orbit (Reubelt et al. 2003a). Therefore, satellite accelerations derived by the three-point formula of Eq. (6) cannot be treated as point-wise. However, there is still a way to relate them to the gravity field model exactly.

It is obvious that the single numerical differentiation of a position results *exactly* in the average velocity within the differentiation interval (this follows from the definition of the average velocity). A similar statement holds also for accelerations. The acceleration $\bar{a}(t)$ obtained with Eq. (6) can be interpreted as the exact average acceleration

$$\bar{a}(t) = \int_{-\Delta t}^{\Delta t} w(s)a(t+s)\mathrm{d}s \tag{7}$$

with a weight function

$$w(s) = \frac{\Delta t - |s|}{(\Delta t)^2} \tag{8}$$

(cf. Fig. 1). To prove this statement, it is sufficient to integrate Eq. (7) by parts

$$\begin{aligned}
&\int_{-\Delta t}^{\Delta t} w(s)a(t+s)\mathrm{d}s \\
&= \frac{1}{(\Delta t)^2}\left\{ \int_{-\Delta t}^{0} (\Delta t+s)a(t+s)\mathrm{d}s + \int_{0}^{\Delta t} (\Delta t-s)a(t+s)\mathrm{d}s \right\} \\
&= \frac{1}{(\Delta t)^2}\left\{ (\Delta t+s)v(t+s)\big|_{s=-\Delta t}^{0} - \int_{-\Delta t}^{0} v(t+s)\mathrm{d}s \right. \\
&\quad \left. +(\Delta t-s)v(t+s)\big|_{s=0}^{\Delta t} + \int_{0}^{\Delta t} v(t+s)\mathrm{d}s \right\} \\
&= \frac{1}{(\Delta t)^2}\{x(t-\Delta t) - 2x(t) + x(t+\Delta t)\}
\end{aligned} \tag{9}$$

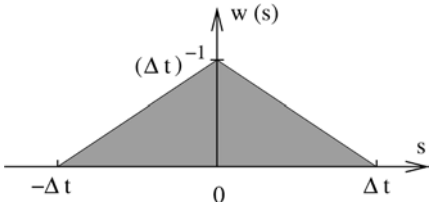

Fig. 1. Weight function that describes the averaging of the accelerations derived with the three-point differentiation scheme

where $v(t)$ is a component of the satellite velocity vector as a function of time: $a(t) = v'(t) = x''(t)$. Notice that the satellite positions must be differentiated in an inertial frame in order to yield the absolute accelerations (i.e. accelerations without centrifugal and Coriolis terms), which can be directly related to the gravity field.

A formal treatment of Eq. (7) would imply that, in order to proceed from point-wise accelerations to average ones, we should know the satellite position in an inertial frame at each moment within the averaging interval. A more practical procedure, which does not require such knowledge, is discussed in Sect. 3.1.2.

It is important to remember that the total satellite accelerations derived from a satellite orbit have to be converted into residual accelerations by subtracting the contribution of the reference gravity field. The discussion of a practical way to compute reference average accelerations is postponed until Sect. 4.2.

2.3 Optimal estimation of the gravity field

Obviously, the observed data (average residual accelerations) depend on the unknown parameters (spherical harmonic coefficients) in a linear manner. Assume that noise in the observations is random and Gaussian. Then, the functional model connecting the data and the unknowns is a standard Gauss–Markov model

$$\mathscr{E}\{\mathbf{d}\} = \mathbf{A}\mathbf{x}, \quad \mathscr{D}\{\mathbf{d}\} = \mathbf{C_d} \tag{10}$$

where $\mathscr{E}$ and $\mathscr{D}$ denote the expectation and dispersion operators, respectively; $\mathbf{d}$ is the vector of observations (average residual accelerations); $\mathbf{x}$ is the vector of unknowns (corrections of the spherical harmonic coefficients); $\mathbf{A}$ is the design matrix; and $\mathbf{C_d}$ is the noise covariance matrix.

In the absence of a regularization, the optimal solution $\hat{\mathbf{x}}$ of the problem of Eq. (10) can be found by means of solving the system of normal equations

$$\hat{\mathbf{x}} = \mathbf{N}^{-1}(\mathbf{A}^T\mathbf{C_d}^{-1}\mathbf{d}) \tag{11}$$

where $\mathbf{N}$ is the normal matrix

$$\mathbf{N} = \mathbf{A}^T\mathbf{C_d}^{-1}\mathbf{A} \tag{12}$$

An efficient algorithm with which to do this is the PCCG method (Hestenes and Stiefel 1952; Bertsekas 1982; Schuh 1996). In the simplest form, this method can be presented as follows.

1. $\mathbf{x_0} = \mathbf{0}, \mathbf{r_0} = \mathbf{A}^T\mathbf{C_d}^{-1}\mathbf{d}, \mathbf{p_0} = \mathbf{s_0} = \tilde{\mathbf{N}}^{-1}\mathbf{r_0}, k = 0$
2. $\mathbf{q_k} = \mathbf{N}\mathbf{p_k}$
3. $\alpha_k = \frac{\mathbf{r_k}^T\mathbf{p_k}}{\mathbf{q_k}^T\mathbf{p_k}}$
4. $\mathbf{x_{k+1}} = \mathbf{x_k} + \alpha_k\mathbf{p_k}$
5. $\mathbf{r_{k+1}} = \mathbf{r_k} - \alpha_k\mathbf{q_k}$
6. If convergence reached, set $\hat{\mathbf{x}} = \mathbf{x_{k+1}}$ and stop
7. $\mathbf{s_{k+1}} = \tilde{\mathbf{N}}^{-1}\mathbf{r_{k+1}}$
8. $\beta_{k+1} = \frac{\mathbf{r_{k+1}}^T\mathbf{s_{k+1}}}{\mathbf{r_k}^T\mathbf{s_k}}$
9. $\mathbf{p_{k+1}} = \mathbf{s_{k+1}} + \beta_{k+1}\mathbf{p_k}$
10. $k = k+1$, go to step (2)

In the description, we have introduced a so-called pre-conditioner $\tilde{\mathbf{N}}$ – an approximation of the normal matrix, which can be computed much faster than the exact one. The stopping criterion may be chosen in different ways. The one we prefer is to check whether the models at two successive iterations differ in average by less than 0.1 mm in terms of geoid heights.

An attractive feature of the PCCG method is that the only role of the true normal matrix is to be applied to a certain vector $\mathbf{p_k}$ (step 2). According to the definition of Eq. (12), this operation can be performed as a sequence of three matrix-to-vector multiplications

$$\mathbf{N}\mathbf{p_k} = (\mathbf{A}^T(\mathbf{C_d}^{-1}(\mathbf{A}\mathbf{p_k}))) \tag{13}$$

Thus, an explicit computation of the normal matrix, which may be a very time-consuming operation, is avoided.

Multiplication of the design matrix with the vector $\mathbf{p_k}$ is just a synthesis, i.e. prediction of the data on the basis of the model built from this vector. The application of matrix $\mathbf{C_d}^{-1}$ can be understood as data weighting: this operation is needed for any data set unless noise is uncorrelated and has unit variance. Finally, the multiplication of the matrix $\mathbf{A}^T$ with a vector is a transposed operation with respect to the synthesis; we will refer to this operation as the co-synthesis. In the following paragraphs, we consider these three steps – the synthesis, the data weighting, and the co-synthesis – as well as the design of an efficient pre-conditioner one by one.

3 Steps of the PCCG method

3.1 Synthesis and co-synthesis

As explained above, the synthesis consists, in essence, of two procedures: (1) point-wise synthesis and (2) averaging. Both operations are linear, hence the following representation of the synthesis is valid:

$$\mathbf{y} = \mathbf{A}\mathbf{p} = \mathbf{E}\mathbf{A_{pw}}\mathbf{p} \tag{14}$$

where $\mathbf{p}$ is the synthesis input; $\mathbf{A_{pw}}$ is the point-wise design matrix (i.e. the matrix describing the point-wise synthesis); $\mathbf{E}$ is the averaging matrix; and the vector $\mathbf{y}$ is the synthesis output. A discussion on the length of the matrix $\mathbf{A_{pw}}$ (i.e. the number of points where the synthesis has to be performed) and the explicit representation of the matrix $\mathbf{E}$ is postponed until Sect. 3.1.2. As follows from Eq. (14), the co-synthesis – transposed synthesis – can be implemented in a similar way provided that all the operations are understood in the transposed sense and performed in the reverse order

$$\mathbf{q} = \mathbf{A}_{\mathbf{pw}}^T\mathbf{E}^T\mathbf{z} \tag{15}$$

where $\mathbf{z}$ is the co-synthesis input and $\mathbf{q}$ is the co-synthesis output. First of all, let us consider the point-wise synthesis and co-synthesis.

3.1.1 Point-wise synthesis and co-synthesis

The simplest way to perform these operations is to compute the design matrix explicitly [cf. Eqs. (4), (3), and (5)], and then to carry out the standard matrix-to-vector multiplication. However, this method is relatively time-consuming. Therefore, an alternative approach has been implemented. It is based on the fast synthesis/co-synthesis procedures developed earlier in the context of SGG data processing (Koop et al. 2000; Ditmar et al. 2003a). The fast synthesis consists of three steps. In the first step, the accelerations are computed at the nodes of a regular three-dimensional (3-D)spherical grid in the local geographical frame. This step can be implemented very efficiently thanks to fast Fourier transform (FFT), as well as to the fact that the Legendre functions have to be computed only once per latitude. Importantly, the time expenditure required by this step is independent of the number of observation points. In the second step, the computed accelerations are interpolated onto the observation points. The implemented interpolation approach makes use of the 3-D Overhauser splines (Overhauser 1968; Ditmar et al. 2003a). Interpolation of this type results in a relatively high accuracy; in addition, it is fast because only $4^3 = 64$ neighboring grid nodes participate in each interpolation operation. In the third step, the computed accelerations are rotated from the local geographical frame into the LORF (a fast operation, which requires only a few operations per point). Note that the number of operations required by the straightforward matrix-to-vector multiplication would be of the order of $L_{\max}^2$ per observation point. Thus, the fast synthesis algorithm may be much more efficient than the straightforward one, especially if the maximum degree $L_{\max}$ and the number of points are large. Furthermore, each of the three fast synthesis steps can be represented as a matrix-to-vector multiplication. Therefore, we can implement the fast co-synthesis simply by understanding these steps in the transposed sense and performing them in the reverse order (Ditmar et al. 2003a). The number of operations required by the fast co-synthesis is approximately the same as in the case of the fast synthesis.

3.1.2 Averaging

3.1.2.1 Averaging as a filter operation. Let us start the discussion from the case of synthesis. The most obvious way to apply the averaging is to perform the point-wise synthesis at a sufficient number of points (distributed much more densely than the original observation points) and then to compute the integral of Eq. (7) for each observation point numerically. Positions of the intermediate points can be found by means of an interpolation: a satellite orbit is a very smooth function of coordinates. On the other hand, the point-wise accelerations computed along the orbit can also be represented by a very smooth function. Therefore, it is smarter to perform an interpolation directly in terms of accelerations! Furthermore, we can represent the accelerations in a given time interval analytically (say, as a polynomial). Then, the integral in Eq. (7) can also be computed analytically. Let us consider the latter strategy in detail.

Let a current observation point be characterized by time t. Assume further that a sequence of point-wise accelerations is known in the time interval $[t-n\Delta t, t+n\Delta t]$ with step Δt (where n is an integer number). On the basis of this information, we can uniquely approximate the accelerations within the above-defined interval as a polynomial of the degree $2n$

$$a(t+s)=\sum_{j=0}^{2n} c_j s^j, \quad s\in[-n\Delta t; n\Delta t] \tag{16}$$

where $c_j = c_j(t)$ are the coefficients of the polynomial and the scalar function $a(t+s)$ represents one of the acceleration components in an inertial frame.

The coefficients c_j can be found explicitly, as the solution of a Vandermonde-type system of linear equations:

$$\mathbf{c}=\mathbf{V}^{-1}\mathbf{a} \tag{17}$$

where $\mathbf{a}$ is the vector composed of point-wise accelerations at times $(t-n\Delta t,\ldots,t+n\Delta t)$; $\mathbf{c}=(c_0,c_1,\ldots,c_{2n})^T$; and $\mathbf{V}$ is the following square matrix:

$$\mathbf{V}=\begin{pmatrix} 1 & -n\Delta t & \cdots & (-n\Delta t)^{2n} \\ 1 & -(n-1)\Delta t & \cdots & (-(n-1)\Delta t)^{2n} \\ \vdots & \vdots & & \vdots \\ 1 & n\Delta t & \cdots & (n\Delta t)^{2n} \end{pmatrix} \tag{18}$$

It is worthwhile to mention that such a system rapidly becomes very ill posed as the degree of the polynomial increases, therefore it must be solved with a tailored algorithm (see e.g. Press et al. 1992).

It easy to show that the polynomial representation of Eq. (16) yields an analytical expression for the integral of Eq. (7)

$$\bar{a}(t)=\mathbf{w}^T\mathbf{c} \tag{19}$$

where elements of the vector $\mathbf{w}$ are defined as

$$\{\mathbf{w}\}_j=\begin{cases} 2(\Delta t)^j\left(\frac{1}{j+1}-\frac{1}{j+2}\right) & \text{for even } j\ \ (j=0,1,\ldots,2n) \\ 0 & \text{for odd } j \end{cases} \tag{20}$$

From Eqs. (17) and (19) it follows that the average acceleration $\bar{a}(t)$ can be directly related to the vector of point-wise accelerations $\mathbf{a}$ as

$$\bar{a}(t)=\mathbf{e}^T\mathbf{a} \tag{21}$$

where

$$\mathbf{e}=(\mathbf{V}^T)^{-1}\mathbf{w} \tag{22}$$

Importantly, elements of the vector $\mathbf{e}$ are constant: they play the role of filter coefficients, which can be computed just once for the whole data set.

It is now obvious that we can represent the matrix $\mathbf{E}$ in Eq. (14) as

$$\mathbf{E}=\mathbf{R}_{\mathbf{I}\to\mathbf{L}}\mathbf{E}_{\mathbf{I}}\mathbf{R}_{\mathbf{L}\to\mathbf{I}} \tag{23}$$

where $\mathbf{R}_{\mathbf{L}\to\mathbf{I}}$ is the matrix for rotation of the whole data set from the LORF into an inertial frame; $\mathbf{R}_{\mathbf{L}\to\mathbf{I}}$ is the matrix for rotation in the opposite direction ($\mathbf{R}_{\mathbf{I}\to\mathbf{L}}=\mathbf{R}_{\mathbf{L}\to\mathbf{I}}^T$); and the matrix $\mathbf{E}_{\mathbf{I}}$ represents the averaging filter. It consists of three independent fragments related to the X, Y and Z data components, respectively

$$\mathbf{E}_{\mathbf{I}}=\begin{pmatrix} \mathbf{E}_{\mathbf{I}}^{(\mathbf{x})} & & \\ & \mathbf{E}_{\mathbf{I}}^{(\mathbf{y})} & \\ & & \mathbf{E}_{\mathbf{I}}^{(\mathbf{z})} \end{pmatrix} \tag{24}$$

In the absence of data gaps, each of these fragments is just a Toeplitz matrix filled with elements of the vector $\mathbf{e}$ (at least, if we are not too close to an edge of the data set). Notice that according to Eqs. (19) and (20), only even degrees of the polynomial participate in the computation of the average acceleration. This means that the averaging filter is symmetric.

In the case of the second-order polynomial, elements of the vector $\mathbf{e}$ can be easily computed with Eq. (22) analytically: (1/12, 5/6, 1/12).

3.1.2.2 Averaging at the edges and in the presence of gaps The way we handle the edges and gaps is based on the following concept. Assume for a moment that the data set contains no gaps and is infinite in time (or, at least, much longer than the actually collected one). Then, the corresponding Gauss–Markov model can be written as follows:

$$\mathscr{E}\{\mathbf{d}^\infty\}=\mathbf{A}^\infty\mathbf{x}, \quad \mathscr{D}\{\mathbf{d}\}=\mathbf{C}_{\mathbf{d}}^\infty \tag{25}$$

where $\mathbf{d}^\infty$, $\mathbf{A}^\infty$, and $\mathbf{C}_{\mathbf{d}}^\infty$ are 'infinite' extensions of the arrays $\mathbf{d}$, $\mathbf{A}$, and $\mathbf{C}_{\mathbf{d}}$, respectively. Assume further that the actual data set is obtained from the infinite one by deliberate 'picking' of selected measurements. In terms of linear algebra, such a picking can be performed by the application of a 'mask matrix' $\mathbf{M}$, which can be produced from the infinite unit matrix by retaining only those rows that correspond to actually made measurements. This means that the actual arrays $\mathbf{d}$, $\mathbf{A}$, and $\mathbf{C}_{\mathbf{d}}$ can be related to their infinite counterparts by the relationships

$$\mathbf{d}=\mathbf{M}\mathbf{d}^\infty \tag{26}$$

$$\mathbf{A}=\mathbf{M}\mathbf{A}^\infty \tag{27}$$

$$\mathbf{C}_{\mathbf{d}}=\mathbf{M}\mathbf{C}_{\mathbf{d}}^\infty\mathbf{M}^T \tag{28}$$

The infinite data set concept, combined with the expression for the averaging matrix of Eq. (23), allows the synthesis formula of Eq. (14) to be re-written as follows:

$$y=\mathbf{M}\mathbf{R}_{\mathbf{I}\to\mathbf{L}}^\infty\mathbf{E}_{\mathbf{I}}^\infty\mathbf{R}_{\mathbf{L}\to\mathbf{I}}^\infty\mathbf{A}_{\mathbf{pw}}^\infty\mathbf{p} \tag{29}$$

This expression means that we should make the point-wise synthesis not only at the true observation points but

also at the points where the observations are missing, as well as at the points preceding and following the actual data set (this should not be a problem if the observation points are defined by a reduced-dynamic POD procedure). Then, we must rotate the results into an inertial frame, apply the averaging filter assuming that the matrices $\mathbf{E}_{\mathbf{I}}^{(\mathbf{x})}$, $\mathbf{E}_{\mathbf{I}}^{(\mathbf{y})}$ and $\mathbf{E}_{\mathbf{I}}^{(\mathbf{z})}$ are purely Toeplitz, and perform the rotation back to the LORF. Finally, we should pick up the values corresponding to the true observation points. In practice, the 'infinite' data set can be produced by adding only n points at the beginning and at the end of the actual one (where n has the same meaning as in the previous section).

3.1.2.3 Averaging in the co-synthesis. A somewhat similar approach can also be followed in the context of the co-synthesis. By analogy with Eq. (29), the co-synthesis formula of Eq. (15) can be re- written as follows:

$$\mathbf{q} = (\mathbf{A}_{\mathbf{pw}}^{\infty})^T \mathbf{R}_{\mathbf{I}\to\mathbf{L}}^{\infty} \mathbf{E}_{\mathbf{I}}^{\infty} \mathbf{R}_{\mathbf{L}\to\mathbf{I}}^{\infty} \mathbf{M}^T \mathbf{z} \tag{30}$$

This means that the input vector for the co-synthesis should be extended to become sufficiently long and uninterrupted, even if the actual data have gaps. In doing so, the new elements should be filled with zeros. The result of this operation is subject to the averaging filtering in an inertial frame. As long as the matrix $\mathbf{E}_{\mathbf{I}}^{\infty}$ is symmetric $[(\mathbf{E}_{\mathbf{I}}^{\infty})^T = \mathbf{E}_{\mathbf{I}}^{\infty}]$, application of the averaging filter in the co-synthesis does not differ from that in the synthesis. Finally, the point-wise co-synthesis should be performed. In doing so, values at the true observation points and at the points introduced during the extension should be treated in the same way.

3.2 Data weighting

As long as noise in orbit-derived accelerations is correlated, a proper data weighting is crucial. As was shown in Sect. 2.3, the data weighting can be implemented by applying the inverse covariance matrix $\mathbf{C}_{\mathbf{d}}^{-1}$ to a certain vector at each iteration of the PCCG method. In other words, we must solve at each iteration a system of linear equations with the matrix $\mathbf{C}_{\mathbf{d}}$. Thereafter, we discuss how this can be done under the assumption that noise in satellite positions is white: in this case, the data weighting can be implemented in a particularly simple and efficient way. We will start the discussion from an approximate data weighting scheme, which is based on the theory of circulant matrices (Davis 1979; Voevodin and Tyrtyshnikov 1987) and applicable to uninterrupted data sets. After that, we will show how an exact data weighting algorithm can be built on the basis of the approximate one. Finally, we will consider the case of data with gaps.

3.2.1 Circulant approximation of the covariance matrix
According to the definition of the functional model, the input for the gravity field modeling is a set of residual accelerations in the LORF. The relationship between the acceleration variations $\boldsymbol{\delta}\mathbf{d}$ and the orbit data variations in an inertial frame $\boldsymbol{\delta}\mathbf{r}$ is as follows:

$$\boldsymbol{\delta}\mathbf{d} = \mathbf{R}_{\mathbf{I}\to\mathbf{L}} \mathbf{D} \delta\mathbf{r} \tag{31}$$

where $\mathbf{D}$ is a matrix that represents the double numerical differentiation: a fragment of this matrix related to a given data component is a tridiagonal matrix with elements $(1/(\Delta t)^2, -2/(\Delta t)^2, 1/(\Delta t)^2)$. Let the noise covariance matrix of the orbit data be $\mathbf{C}_{\mathbf{r}} = \sigma^2\mathbf{I}$, where σ^2 is the noise variance (equal for all three observational components). Then, the data covariance matrix $\mathbf{C}_{\mathbf{d}}$, which is used in the definition of the functional model of Eq. (10), can be represented as

$$\mathbf{C}_{\mathbf{d}} = \mathbf{R}_{\mathbf{I}\to\mathbf{L}} \mathbf{C}_{\mathbf{d_I}} \mathbf{R}_{\mathbf{L}\to\mathbf{I}} \tag{32}$$

where $\mathbf{C}_{\mathbf{d_I}}$ is the covariance matrix of the orbit-derived accelerations in an inertial frame: $\mathbf{C}_{\mathbf{d_I}} = \sigma^2\mathbf{D}\mathbf{D}^T$. The inverse of Eq. (32) reads

$$\mathbf{C}_{\mathbf{d}}^{-1} = \mathbf{R}_{\mathbf{I}\to\mathbf{L}} \mathbf{C}_{\mathbf{d_I}}^{-1} \mathbf{R}_{\mathbf{L}\to\mathbf{I}} \tag{33}$$

This means that the multiplication of the matrix $\mathbf{C}_{\mathbf{d}}^{-1}$ and a vector can be split into three steps: (1) rotation from the LORF to an inertial frame; (2) solving the system of linear equations with matrix $\mathbf{C}_{\mathbf{d_I}}$; and (3) rotation back to the LORF.

The matrix $\mathbf{C}_{\mathbf{d_I}}$ consists of three Toeplitz sub-matrices, each of which corresponds to one of the components of the acceleration vector

$$\mathbf{C}_{\mathbf{d_I}}^{(\mathbf{x}|\mathbf{y}|\mathbf{z})} = \frac{\sigma^2}{(\Delta t)^4} \begin{pmatrix} 6 & -4 & 1 & & \\ -4 & 6 & -4 & 1 & \\ 1 & -4 & 6 & -4 & \ddots \\ & 1 & -4 & 6 & \ddots \\ & & \ddots & \ddots & \ddots \end{pmatrix} \tag{34}$$

where '$(\mathbf{x}|\mathbf{y}|\mathbf{z})$' denotes one of the components: 'x', 'y', or 'z'. For the sake of brevity, this superscript will hereafter be omitted.

Let us approximate the component-related matrix $\mathbf{C}_{\mathbf{d_I}}$ with a circulant one, i.e. with a matrix each row of which can be produced from the previous one by putting the last element into the first place, so that the whole row gets shifted to the right by one position. This approximation looks as follows:

$$\tilde{\mathbf{C}}_{\mathbf{d_I}} = \frac{\sigma^2}{(\Delta t)^4} \begin{pmatrix} 6 & -4 & 1 & & & \ddots & 1 & -4 \\ -4 & 6 & -4 & 1 & & & \ddots & 1 \\ 1 & -4 & 6 & -4 & 1 & & & \ddots \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ \ddots & & 1 & -4 & 6 & -4 & 1 & \\ 1 & \ddots & & 1 & -4 & 6 & -4 & \\ -4 & 1 & \ddots & & 1 & -4 & 6 & \end{pmatrix} \tag{35}$$

Obviously, the circulant approximation $\tilde{\mathbf{C}}_{\mathbf{d_I}}$ differs from the true matrix $\mathbf{C}_{\mathbf{d_I}}$ by only three elements in the upper right and in the lower left corner.

The circulant approximation yields the explicit representation for the square root of the covariance matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$, which plays a significant role in the proposed data weighting algorithm. It can be checked directly that

$$(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}} = \frac{\sigma}{(\Delta t)^2}\begin{pmatrix} 2 & -1 & & \ddots & -1 \\ -1 & 2 & -1 & & \ddots \\ & \ddots & \ddots & \ddots & \\ \ddots & & -1 & 2 & -1 \\ -1 & \ddots & & -1 & 2 \end{pmatrix} \tag{36}$$

We may also treat the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$, up to the sign, as a circulant approximation of the matrix $\sigma\mathbf{D}$ (or, more precisely, of its fragment related to a given component). As far as the sign is concerned, we can define it arbitrarily, just as in case of the square root of a scalar value. The definition we prefer corresponds to the positive semidefinite matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ (i.e. such that its eigenvalues are non-negative).

Furthermore, the inverse of a circulant matrix can be found with ease by means of the discrete Fourier transform (Davis 1979). Let us introduce the matrix of discrete Fourier transform $\mathbf{F}$

$$\{\mathbf{F}\}_{j,k} = \mathrm{e}^{i\frac{2\pi(j-1)(k-1)}{N}}; \quad j,k = 1,\ldots,N \tag{37}$$

where N is the number of rows/columns in the circulant matrices defined above. It can be easily checked that $\mathbf{F}\mathbf{F}^* = \mathbf{F}^*\mathbf{F} = N\mathbf{I}$, where $\mathbf{F}^*$ is the Hermitian conjugate (transposed complex-conjugate) of $\mathbf{F}$.

The application of the matrix $\mathbf{F}$ to a certain vector $\mathbf{v}$ with a subsequent scaling results in the spectrum of this vector

$$\mathscr{F}[\mathbf{v}] = \frac{1}{N}\mathbf{F}\mathbf{v} \tag{38}$$

whereas the application of the matrix $\mathbf{F}^*$ to the spectrum restores the original vector

$$\mathbf{v} = \mathbf{F}^*(\mathscr{F}[\mathbf{v}]) \tag{39}$$

Let us introduce a diagonal matrix $\mathbf{S_D}$ obtained by distributing the spectrum of the first row of the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ along the main diagonal

$$\mathbf{S_D} = \mathrm{diag}\left(\frac{1}{N}\mathbf{F}\left\{(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}\right\}_1\right) \tag{40}$$

Then, the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ can be restored from the matrix $\mathbf{S_D}$ as follows (Davis 1979):

$$(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}} = \mathbf{F}^*\mathbf{S_D}\mathbf{F} \tag{41}$$

From this representation, it immediately follows that the inverse of the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ can be found by taking the inverse spectrum with a proper re-scaling

$$(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{-\frac{1}{2}} = \frac{1}{N^2}\mathbf{F}^*(\mathbf{S_D})^{-1}\mathbf{F} \tag{42}$$

Importantly, the inverse of a circulant matrix is also circulant (Davis 1979). Thus, the application of either the original matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ or its inverse to a certain vector is nothing but a cyclic convolution, i.e. filtering. Then, the above-written method of inverting a circulant matrix is in agreement with the well-known statement that a de-convolution can be performed in the Fourier domain by multiplication with the inverse spectrum. The filter represented by the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{-\frac{1}{2}}$ is sometimes called 'whitening' because the application of this matrix to a data set makes the data noise white (except for the edges).

The elements of the diagonal matrix $\mathbf{S_D}$ can be computed analytically from Eqs. (36), (37), and (40)

$$\mathbf{S_D}\}_{kk} = \frac{2\sigma}{N(\Delta t)^2}\left(1-\cos\frac{2\pi k}{N}\right) = \frac{2\sigma}{N(\Delta t)^2}(1-\cos\omega_k\Delta t);$$
$$k = 0,i,\ldots N-1 \tag{43}$$

where ω_k is the cyclic frequency corresponding to the k-th spectral line: $\omega_k = 2\pi k/(N\Delta t)$.

Let us consider the spectrum of Eq. (43) at low frequencies: $\omega_k\Delta t \ll 1$. Under this condition, the function $\cos\omega_k\Delta t$ can be approximated as $1-(\omega_k\Delta t)^2/2$, hence Eq. (43) transforms into

$$\{\mathbf{S_D}\}_{kk} \approx \frac{\sigma}{N}\omega_k^2 \tag{44}$$

This expression is in agreement with the well-known fact that the double differentiation of a function is equivalent to the multiplication of the function spectrum with the factor $-\omega^2$.

Unfortunately, the expressions of Eqs. (43) and (44) reach zero at the zero frequency, so that the spectrum is not invertible. This is not surprising because the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ is ill-posed: it has a zero eigenvalue that corresponds to the eigenvector $(1,1,\ldots,1)^T$. In other words, the double-differentiation operation, if approximated by the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$, always returns a function with zero mean.

3.2.2 Regularization

One possible way to make the matrix $(\tilde{\mathbf{C}}_{\mathbf{d_I}})^{\frac{1}{2}}$ invertible is to introduce a regularized matrix $(\tilde{\mathbf{C}}^{\mathbf{reg}}_{\mathbf{d_I}})^{\frac{1}{2}}$ by adding a small positive value α_c^2 to the diagonal elements of the original matrix

$$(\tilde{\mathbf{C}}^{\mathbf{reg}}_{\mathbf{d_I}})^{\frac{1}{2}} = \frac{\sigma}{(\Delta t)^2}\begin{pmatrix} 2+\alpha_c^2 & -1 & & \ddots & -1 \\ -1 & 2+\alpha_c^2 & -1 & & \ddots \\ & \ddots & \ddots & \ddots & \\ \ddots & & -1 & 2+\alpha_c^2 & -1 \\ -1 & \ddots & & -1 & 2+\alpha_c^2 \end{pmatrix} \tag{45}$$

Then, the corresponding representation in the Fourier domain $\mathbf{S}^{\mathbf{reg}}_{\mathbf{D}}$ becomes

$$\{\mathbf{S}_{\mathbf{D}}^{\mathbf{reg}}\}_{kk} = \frac{\sigma}{N(\Delta t)^2}\left[2(1-\cos\omega_k\Delta t) + \left(\frac{\Delta t}{\tau}\right)^2\right] \tag{46}$$

with $\tau \equiv \Delta t/\alpha_c$. The low-frequency approximation of this expression is as follows:

$$\{\mathbf{S}_{\mathbf{D}}^{\mathbf{reg}}\}_{kk} \approx \frac{\sigma}{N}\left(\omega_k^2 + \frac{1}{\tau^2}\right) \tag{47}$$

Thus, the goal is reached: the spectrum is not equal to zero at zero (or any other) frequency, hence it is invertible. Dependence of the obtained model on the parameter τ is considered in Sect. 4.4.

Adding a regularization to the covariance matrix can be understood as postulating that the data uncertainties are not infinitesimal at any frequency. Then, the weights assigned to the data, which are inversely proportional to the uncertainties, are never infinite.

3.2.3 Whitening filter design

Equations (39), (42), and (46) allow the coefficients of the whitening filter to be obtained numerically: it is enough to take elements of the diagonal matrix $\frac{1}{N^2}(\mathbf{S}_{\mathbf{D}}^{\mathbf{reg}})^{-1}$, arrange them as a vector and perform the inverse Fourier transform (i.e. apply the matrix $\mathbf{F}^*$). We will show, however, that it is more practical to build the whitening filter analytically. This can be done on the basis of the low-frequency approximation of Eq. (47), which has to be inverted, re-scaled, and subject to the inverse Fourier transform as

$$\frac{1}{N\sigma}\sum_{k=0}^{N-1} \mathrm{e}^{-i\omega_k\Delta t}\left(\omega_k^2 + \frac{1}{\tau^2}\right)^{-1}, \quad j = 0,\ldots,N-1 \tag{48}$$

Assume further that the sampling interval tends to zero whereas the duration of the data series tends to infinity. Then, Eq. (48) turns into the Fourier integral, which provides the whitening filter $f(t)$ as a continuous function

$$f(t) = \frac{1}{2\pi\sigma}\int_{-\infty}^{\infty} \mathrm{e}^{-i\omega t}\left(\omega^2 + \frac{1}{\tau^2}\right)^{-1} \mathrm{d}\omega, \quad t \in [-\infty,\infty] \tag{49}$$

The computation of the integral can easily be done analytically, yielding

$$f(t) = \frac{\tau}{2\sigma}\mathrm{e}^{-\frac{|t|}{\tau}} \tag{50}$$

or after discretization

$$f_j = \frac{\Delta t\tau}{2\sigma}\mathrm{e}^{-\frac{|j|\Delta t}{\tau}} = \frac{\Delta t\tau}{2\sigma}\mathrm{e}^{-|j|\alpha_c} \tag{51}$$

Thus, the parameter τ may be interpreted as the characteristic half-width of the whitening filter: at the time $t = \tau$ the filter coefficients approach the level of 37% of the maximum, which is reached at the time $t = 0$.

We compared the analytically derived filter with the exact one – obtained numerically from Eq. (46) – for several ratios $\tau/\Delta t$. We found that for $\tau/\Delta t \geq 10$ the analytical expression offers a sufficiently good approximation of the whitening filter; the difference between the two filters does not exceed 1%. Furthermore, we will show in Sect. 3.2.5 that the accuracy of the whitening filter specification does not influence the obtained model at all if the exact data weighting scheme is applied. However, the analytical expression of Eq. (51) allows the whitening filter to be implemented in a very efficient way. Hence we routinely use the analytically derived whitening filter in the computations presented below.

3.2.4 Whitening filter implementation.

As long as the filter coefficients have been derived, it is no longer necessary to assume that the filtering is carried out by means of the cyclic convolution; the conventional convolution can be used instead. Then, the application of the filter of Eq. (51) to an input vector $\mathbf{y}$ generates the output vector $\mathbf{z}$ according to the formula

$$z_n = \frac{\Delta t\tau}{2\sigma}\sum_{j=-\infty}^{\infty} y_{n+j}\mathrm{e}^{-|j|\alpha_c} \tag{52}$$

Equation (52) suggests that indices in the input vector run from 'minus infinity' to 'plus infinity'. For the time being, let us assume that the input vector elements outside the range from 1 to N are simply set equal to zero. A somewhat more elaborate approach is considered in Sect. 4.4.2.

The summation in Eq. (52) can be split into two: over non-positive indices j and over positive ones, so that we have $z_n = z_n^- + z_n^+$ with

$$z_n^- = \frac{\Delta t\tau}{2\sigma}\sum_{j=-\infty}^{0} y_{n+j}\mathrm{e}^{j\alpha_c} \tag{53}$$

and

$$z_n^+ = \frac{\Delta t\tau}{2\sigma}\sum_{j=1}^{\infty} y_{n+j}\mathrm{e}^{-j\alpha_c} \tag{54}$$

Consider the first term. Let us re-write Eq. (53) for the output element number $n+1$

$$\begin{aligned} z_{n+1}^- &= \frac{\Delta t\tau}{2\sigma}\sum_{j=-\infty}^{0} y_{n+j+1}\mathrm{e}^{j\alpha_c} \\ &= \frac{\Delta t\tau}{2\sigma}\left[y_{n+1} + \sum_{j=-\infty}^{-1} y_{n+j+1}\mathrm{e}^{j\alpha_c}\right] \\ &= \frac{\Delta t\tau}{2\sigma}\left[y_{n+1} + \mathrm{e}^{-\alpha_c}\sum_{j=-\infty}^{0} y_{n+j}\mathrm{e}^{j\alpha_c}\right] \end{aligned} \tag{55}$$

The summation in the latter expression is exactly equal to that in Eq. (53) for element n. Thus, we have obtained a recursive relationship

$$z_{n+1}^- = \frac{\Delta t\tau}{2\sigma}y_{n+1} + \mathrm{e}^{-\alpha_c}z_n^- \tag{56}$$

A similar recursive scheme can also be derived for the second term, Eq. (54). The only difference is that in

order to make the recursive computations stable we should relate the current element not to the previous element but to the next one

$$z_n^+ = \mathrm{e}^{-\alpha_c}\left[\frac{\Delta t\tau}{2\sigma}y_{n+1} + z_{n+1}^+\right] \tag{57}$$

Equations (56) and (57) allow the filtering to be implemented very efficiently.

It is important to remember that the filter presented is the whitening filter, which represents the square root of the matrix $\tilde{\mathbf{C}}_{\mathbf{d_l}}^{\mathbf{reg}}$. Therefore, this filter must be applied twice at each iteration of the PCCG method.

3.2.5 Exact data weighting

The weighting procedure we have derived so far is based on the circulant approximation of the covariance matrix, i.e. is by definition inexact. In Sect. 4, we will show that the errors introduced for this reason may be very significant. Therefore, it is preferable to multiply the inverse covariance matrix with a vector more accurately. This can be done by means of a low-level PCCG procedure (Klees and Ditmar in press). Such a procedure is similar to the high-level PCCG procedure presented in Sect. 2.3, provided that: (1) the normal matrix $\mathbf{N}$ is replaced by the covariance matrix $\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}}$; (2) the data vector $\mathbf{d}$ is replaced by the vector to which the matrix $(\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}})^{-1}$ has to be applied; (3) the pre-conditioner is re-defined (see below); and (4) the length of all the vectors involved is now equal to N. The low-level PCCG procedure contains two core operations: (1) exact multiplication of the covariance matrix $\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}}$ with a vector and (2) approximate multiplication of the inverse covariance matrix with a vector (pre-conditioning). The first operation is straightforward because the covariance matrix is just a five-diagonal Toeplitz matrix with explicitly known elements

$$\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}} = \frac{\sigma^2}{(\Delta t)^4} \times \begin{pmatrix} 2+(\alpha_c^2+2)^2 & -2(\alpha_c^2+2) & 1 & \\ -2(\alpha_c^2+2) & 2+(\alpha_c^2+2)^2 & -2(\alpha_c^2+2) & \ddots \\ 1 & -2(\alpha_c^2+2) & 2+(\alpha_c^2+2)^2 & \ddots \\ \ddots & \ddots & & \ddots \end{pmatrix} \tag{58}$$

As far as the pre-conditioning is concerned, the filtering procedure presented in Sects. 3.2.3 and 3.2.4 offers a fast and efficient way to carry out this operation. Simulations show that the system of linear equations with matrix $\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}}$ can be solved by the low-level PCCG procedure in only a few iterations (typically, not more than 10). This is important because this procedure is executed at each iteration of the high-level PCCG procedure.

Hereafter, we will refer to this algorithm as the 'exact data weighting'. It is important to emphasize that it is exact only in the sense that it does not approximate the covariance matrix with a circulant one. A regularization of the covariance matrix still remains essential.

3.2.6 Data with gaps

First of all, let us consider how an adaptation to data with gaps can be done in the case of exact data weighting. The approach we follow was originally developed by Klees and Ditmar (in press) in the context of the ARMA (AutoRegressive – Moving Average) filters. We reproduce this idea below, using the notion of the mask matrix, which allows us to make the description very concise and clear.

The accuracy of the exact data weighting algorithm depends on how precisely we perform step 2 in the PCCG method, i.e. the multiplication of the covariance matrix with a vector. According to the infinite data set concept [cf. Eq. (28)], step 2 can be re-written in the presence of gaps as follows:

$$\bar{\mathbf{q}}_{\mathbf{k}} = \mathbf{M}(\mathbf{C}_{\mathbf{d_l}}^{\mathbf{reg}})^{\infty}\mathbf{M}^T\bar{\mathbf{p}}_{\mathbf{k}} \tag{59}$$

where vectors $\bar{\mathbf{p}}_{\mathbf{k}}$ and $\bar{\mathbf{q}}_{\mathbf{k}}$ are supplied with bars in order to distinguish them from vectors $\mathbf{p}_{\mathbf{k}}$ and $\mathbf{q}_{\mathbf{k}}$ in the high-level PCCG procedure. Notice that the 'infinite extension' is applied to the component-dependent covariance matrix in the inertial frame, not to the actual covariance matrix $\mathbf{C}_{\mathbf{d}}$. This is justified if the gaps in all three observational components are identical. Equation (59) can be executed from the right to the left, which suggests the following algorithm.

1. Take the original vector $\bar{\mathbf{p}}_{\mathbf{k}}$ and extend it by appending zero elements at the beginning and at the end as well as inside the gaps.
2. Apply the filter with coefficients $(1, -2(\alpha_c^2+2), 2+(\alpha_c^2+2)^2, -2(\alpha_c^2+2), 1)$ (one row of the matrix $\mathbf{C}_{\mathbf{d}}^{\infty}$).
3. Pick up the values that correspond to the actual measurements made.

The other operation to be considered is the pre-conditioning, i.e. application of the whitening filter. As long as this operation does not influence the final results, we are reasonably flexible in choosing a way to implement it. For example, we can fill the gaps with zeros prior to filtering and ignore them thereafter.

Finally, the approximate data weighting scheme needs an adaptation to data with gaps too. In the first instance, the approximate data weighting can be implemented in the same way as the pre-conditioning in the low-level PCCG procedure. A somewhat better (though empirical) approach is considered in the course of the numerical study; see Sect. 4.4.2.

3.3 Pre-conditioning

In order to minimize the number of iterations in the high-level PCCG procedure, we must specify a proper pre-conditioner, i.e. an approximation of the normal matrix. The requirements for the selection of a

pre-conditioner are somewhat contradictory. On the one hand, it must be sufficiently close to the normal matrix. On the other hand, it must be quickly computable, otherwise a speed-up due to a reduced number of iterations will be over-balanced by the CPU time required for the generation of the pre-conditioner itself. Ideally, a pre-conditioner should also allow for a fast solution of the corresponding system of linear equations. In general, building a good pre-conditioner is a non-trivial problem, which does not have a universal solution. Fortunately, some satellite data (in particular, satellite accelerations) offer a natural way to build an efficient pre-conditioner with a minor computational load.

A column of the point-wise design matrix $\mathbf{A_{pw}}$ can be split into three fragments, each of which corresponds to a certain observational component. Consider the column elements within one of these fragments, having in mind that the observational components are defined in the LORF. Then, as was shown by Colombo (1986, 1989), the column elements can be represented by the sum of a very limited number of sinusoidal/co-sinusoidal functions of different frequencies, provided that the following conditions are met: (1) the radius and inclination of the satellite orbit are constant; (2) the angular velocity of the satellite and the orbit precession rate are constant; (3) the data have no gaps and the sampling rate is constant; (4) the orbit is repeat, i.e. the number of nodal days N_d and the number of satellite revolutions N_r are integer values. If, furthermore, the satellite makes a sufficient number of revolutions (at least $2L_{\max}+1$) without returning to the original track in the terrestrial frame, the frequencies related to the design matrix cohimns of different orders m do not coincide. In the absence of data weighting, each element of the normal matrix is produced at a scalar product of two design matrix columns. This operation is equivalent (up to a scaling factor) to computing the scalar product of column spectra. It means that two columns related to different orders produce a zero element in the normal matrix. Moreover, an element of the normal matrix is also equal to zero if the two design matrix columns belong to the same order but one of them is related to a 'C-' coefficient in the gravity field model whereas the other one is related to an 'S-' coefficient. For these reasons, the normal matrix becomes block-diagonal under Colombo's assumptions: it gets split into $2L_{\max}+1$ independent blocks, the size of which does not exceed $(L_{\max}+1)\times(L_{\max}+1)$ (Koop 1993). Naturally, the corresponding systems of linear equations can be solved with ease.

It is well known that a block-diagonal approximation of the normal matrix can also be derived for the case of colored noise in the data (Schrama 1990; Koop 1993; Ditmar and Klees 2002). Let us show how this approximation will look like in the presence of both frequency-dependent data weighting and averaging filtering. Equations (12), (23), and (33), in combination with the definition of the design matrix $\mathbf{A}=\mathbf{E}\mathbf{A_{pw}}$, yield the following representation of the normal matrix:

$$\begin{aligned}\tilde{\mathbf{N}} &= \tilde{\mathbf{A}}_{\mathbf{pw}}^{T}(\mathbf{R_{I\to L}}\mathbf{E_I}\mathbf{R_{L\to I}})\left(\mathbf{R_{I\to L}}(\mathbf{C}_{\mathbf{d_I}}^{\mathbf{reg}})^{-1}\mathbf{R_{L\to I}}\right)\\ &\quad\times(\mathbf{R_{I\to L}}\mathbf{E_I}\mathbf{R_{L\to I}})\tilde{\mathbf{A}}_{\mathbf{pw}}\\ &= \tilde{\mathbf{A}}_{\mathbf{pw}}^{T}\mathbf{R_{I\to L}}\mathbf{E_I}(\mathbf{C}_{\mathbf{d_I}}^{\mathbf{reg}})^{-1}\mathbf{E_I}\mathbf{R_{L\to I}}\tilde{\mathbf{A}}_{\mathbf{pw}}\end{aligned} \tag{60}$$

where the 'tilde' indicates the Colombo approximation.

The presence of rotation matrices $\mathbf{R_{I\to L}}$ and $\mathbf{R_{L\to I}}$ ensures that filtering is performed in an inertial frame. In a sufficiently short time period, however, the orientation of the satellite velocity changes insignificantly, so that the LORF itself can be considered as an inertial frame. Therefore, the rotation matrices can be omitted in Eq. (60) under the assumption that the averaging and the whitening filters are short, i.e. much shorter than the satellite revolution period. Then, the normal matrix can be represented as

$$\tilde{\mathbf{N}} = \tilde{\mathbf{B}}^{T}\tilde{\mathbf{B}} \tag{61}$$

where

$$\tilde{\mathbf{B}} \equiv (\mathbf{C}_{\mathbf{d_I}}^{\mathbf{reg}})^{-\frac{1}{2}}\mathbf{E_I}\tilde{\mathbf{A}}_{\mathbf{pw}} \tag{62}$$

is a filtered design matrix; each column of it is produced by applying the whitening and the averaging filter to the corresponding column of the matrix $\tilde{\mathbf{A}}_{\mathbf{pw}}$. Importantly, the filtering operations handle all three observational components independently, so that Eq. (62) can also be understood in the component-wise sense. Then, Eq. (61) can be re-written as

$$\tilde{\mathbf{N}} = \tilde{\mathbf{B}}^{(\mathbf{x})^T}\tilde{\mathbf{B}}^{(\mathbf{x})} + \tilde{\mathbf{B}}^{(\mathbf{y})^T}\tilde{\mathbf{B}}^{(\mathbf{y})} + \tilde{\mathbf{B}}^{(\mathbf{z})^T}\tilde{\mathbf{B}}^{(\mathbf{z})} \tag{63}$$

In the rest of the discussion, we will refer to the design matrix and to the filters in the sense of a particular observational component.

Following the approach presented in Sect. 3.2, let us assume that the averaging and the whitening filters are implemented as the cyclic convolution, so that the matrices $\mathbf{E_I}$ and $(\mathbf{C}_{\mathbf{d_I}}^{\mathbf{reg}})^{-\frac{1}{2}}$ can be approximated as

$$\mathbf{E_I} \approx \mathbf{F}^{*}\mathbf{S_E}\mathbf{F}, \quad (\mathbf{C}_{\mathbf{d_I}}^{\mathbf{reg}})^{-\frac{1}{2}} \approx \mathbf{F}^{*}\mathbf{S_W}\mathbf{F} \tag{64}$$

where $\mathbf{S_E}$ and $\mathbf{S_W}$ are diagonal matrices filled with the spectra of averaging and whitening filters, respectively (the superscript referring to a particular observational component is omitted). Furthermore, let us introduce a matrix of spectral amplitudes $\mathscr{A}$ as the result of the Fourier transform applied to the columns of the point-wise design matrix

$$\mathscr{A} = \frac{1}{N}\mathbf{F}\tilde{\mathbf{A}}_{\mathbf{pw}} \tag{65}$$

Under Colombo's assumptions, the matrix $\mathscr{A}$ is very sparse: the number of non-zero elements in each column of this matrix equals the number of spectral lines in the corresponding column of the matrix $\tilde{\mathbf{A}}_{\mathbf{pw}}$. The substitution of Eq. (64) into Eq. (62) yields

$$\tilde{\mathbf{B}} = \mathbf{F}^{*}\mathbf{S_E}\mathbf{F}\mathbf{F}^{*}\mathbf{S_W}\mathbf{F}\tilde{\mathbf{A}}_{\mathbf{pw}} = N^2\mathbf{F}^{*}\mathbf{S_W}\mathbf{S_E}\mathscr{A} \tag{66}$$

It is convenient to assume that the spectra $\mathbf{S_E}$ and $\mathbf{S_W}$ are represented as continuous functions, $S_E(\omega)$ and $S_W(\omega)$ respectively. Then, we can say that the application of filters changes the amplitude of a spectral line at frequency ω by the factor $N^2 S_E(\omega) S_W(\omega)$.

Hereafter, we present the explicit expressions for elements of each block in the approximated normal matrix $\tilde{\mathbf{N}}$. These expressions are slightly different for 'CC' and 'SS' elements of the normal matrix (a 'CC element' means that it is produced from two design matrix columns related to 'C' coefficients in the gravity model; a similar explanation holds for an 'SS element')

$$\begin{Bmatrix} \{\tilde{\mathbf{N}}^{(\mathbf{CC}),(\mathbf{m})}\}_{l_1,l_2} \\ \{\tilde{\mathbf{N}}^{(\mathbf{SS}),(\mathbf{m})}\}_{l_1,l_2} \end{Bmatrix} = \frac{N}{2}\mathscr{P}_{l_1,l_2}\begin{Bmatrix} (1+\delta_{m,o}) \\ (1-\delta_{m,o}) \end{Bmatrix} \times \sum_{k=-\min(l_1,l_2)}^{\min(l_1,l_2)} (H_{l_1,m,k}H_{l_2,m,k} \times [N^4 S_E^2(\omega_{k,m}) S_W^2(\omega_{k,m})]) \tag{67}$$

The notation used is as follows: $\mathscr{P}_{l_1,l_2}$ is the parity function

$$\mathscr{P}_{l_1,l_2} = \begin{cases} 1 & \text{if } l_1 \text{ and } l_2 \text{ have the same parity} \\ 0 & \text{if } l_1 \text{ and } l_2 \text{ have a different parity} \end{cases} \tag{68}$$

$\delta_{m,0}$ is the Kronecker symbol

$$\delta_{m,0} = \begin{cases} 1 & \text{if } m = 0 \\ 0 & \text{if } m \neq 0 \end{cases} \tag{69}$$

$\omega_{k,m}$ is the cyclic frequency of a given spectral line: $\omega_{k,m} = 2\pi(kN_r + mN_d)/(N\Delta t)_i$ $H_{l,m,k} = H_{l,m,k}^{(x|y|z)}$ is a component-dependent factor

$$\begin{aligned} H_{l,m,k}^{(x)} &= k\frac{GM_E}{R^2}\left(\frac{R}{r}\right)^{l+2} F_{l,m,k}(I) \\ H_{l,m,k}^{(y)} &= \frac{GM_E}{R^2}\left(\frac{R}{r}\right)^{l+2} F_{l,m,k}^{*}(I) \\ H_{l,m,k}^{(z)} &= -(l+1)\frac{GM_E}{R^2}\left(\frac{R}{r}\right)^{l+2} F_{l,m,k}(I) \end{aligned} \tag{70}$$

where r is the orbit radius; I is the orbit inclination; and $F_{l,m,k}(I)$ and $F_{l,m,k}^{*}(I)$ are the so-called inclination functions and cross-track inclination functions, respectively (Kaula 1966; Betti and Sansò 1989; Schrama 1990; Sneeuw 1992; Koop 1993).

The factor in square brackets in Eq. (67) indicates the contribution of the averaging and the whitening filters. The spectrum $S_E(\omega)$ can be obtained numerically by applying the discrete Fourier transform to the averaging filter coefficients, which are given by Eq. (22). The spectrum $S_W(\omega)$ can readily be obtained by inverting Eq. (46) and applying, according to Eq. (42), the scaling factor N^{-2}

$$S_W(\omega) = \frac{(\Delta t)^2}{N\sigma}[2(1-\cos\omega\Delta t) + \alpha_c^2]^{-1} \tag{71}$$

It is worth adding that the inclination functions $F_{l,m,k}(I)$ are equal to zero when indices l and k have a different parity, so that the summation in Eq. (67) can be done with stride 2 in the case of the x and z components. Furthermore, the cross-track inclination functions $F_{l,m,k}^{*}(I)$ are equal to zero when indices l and k have the same parity, so that in case of the y components the summation in Eq. (67) can be done with stride 2 from $-\min(l_1,l_2)+1$ to $\min(l_1,l_2)-1$.

The final expression for a block in the block-diagonal approximation of the normal matrix is the sum of Eqs. (67) over all three components [cf. Eq. (63)].

4 Numerical experiments

4.1 Simulated orbits

In order to illustrate the capacities of the developed technique, we have performed a numerical study. Two satellite orbits have been considered (see Table 1). Both orbits have been computed with a numerical integration technique (see e.g. Seeber 1993) by Dr. P. Visser (Aerospace Department, Delft University of Technology), who used for that purpose the GEODYN software (Pavlis et al. 1997). These orbits are purely gravitational (non-conservative forces are not modeled), which is consistent with the GOCE mission setup. The direct and indirect tidal effects have been taken into account. One of the orbits is considered as is, whereas the other one is artificially contaminated by 1-cm white noise. Each of the orbits is used both to derive the satellite accelerations and to define the observation points related to these accelerations.

The noise-free orbit is mostly used to compare the performance of averaging filters of different orders. With the noisy orbit, we demonstrate the influence of colored noise in the data and the ability of different data weighting schemes to cope with it.

In order to quantity the quality of computed gravity field models, geoid height errors have been computed in the latitudinal band $\pm 80°$ with step $1°$ in both the latitudinal and the longitudinal direction (the polar areas

Table 1. Parameters of the simulated orbits

	Noise-free orbit	Noisy orbit
'True' gravity field model	EGM96 (Lemoine et al. 1998)	EGM96
Truncated at degree/order	50	80
Duration	10 days	10 days
Number of revolutions	161	161
Mean inclination	96.6°	96.6°
Radius	6624 ± 6 km	6624 ± 6 km
Average elevation above the equator	246 km	246 km
Noise	–	1-cm white noise
Sampling rate	30 s	1 s

are eliminated from the evaluation because they are not covered by the orbits). These errors are used to determine the root mean square (RMS) and the maximum geoid height error for different data processing scenarios.

4.2 Deriving residual accelerations

The double differentiation of the orbit data [Eq. (6)] results in the total average accelerations at the points of the observed orbit: $\bar{\mathbf{a}} = \bar{\mathbf{a}}(\mathbf{r}^{\mathrm{obs}})$ (as before, the bar denotes average values). They have to be converted into residual average accelerations $\Delta\bar{\mathbf{a}}(\mathbf{r}^{\mathrm{obs}})$

$$\Delta\bar{\mathbf{a}}(\mathbf{r}^{\mathrm{obs}}) = \bar{\mathbf{a}}(\mathbf{r}^{\mathrm{obs}}) - \bar{\mathbf{a}}_0(\mathbf{r}^{\mathrm{obs}}) \tag{72}$$

where $\bar{\mathbf{a}}_0(\mathbf{r}^{\mathrm{obs}})$ are the reference average accelerations at the observed satellite positions. In this numerical study, the reference accelerations are obtained by differentiation of an auxiliary orbit computed on the basis of the reference gravity field (Table 2). The auxiliary orbits have been split into ten 1-day arcs in order to ensure smaller differences with respect to the 'observed' orbits. The derived sets of reference accelerations contain nine two-sample gaps, which correspond to the beginnings/ends of the arcs. In computing the residual accelerations, we took into account the fact that the reference accelerations $\bar{\mathbf{a}}_0(\mathbf{r}^{\mathrm{ref}})$ are related to the points at the reference orbits. Therefore, the residual accelerations have been corrected by the amount $\bar{\mathbf{a}}_0(\mathbf{r}^{\mathrm{obs}}) - \bar{\mathbf{a}}_0(\mathbf{r}^{\mathrm{ref}})$.

It is worth adding that the average reference accelerations can also be computed without an auxiliary orbit. This option is discussed in Sect. 5.

4.3 Comparison of the averaging filters

In order to study the influence of the averaging filter on the produced model, we have considered the noise-free data set ($\Delta t = 30$ s). First of all, we have tried to process the data without the averaging filter at all. The resulting RMS geoid height error in this case equals 4.4 cm. After switching the averaging filter on, the model quality improves. Figure 2 shows how the RMS geoid height error decreases with increasing filter order. A 'saturation' is reached at about the order 12, when the error equals 0.044 cm. Further increasing the order yields only negligible improvements (e.g. the 20th-order filter results in an error of 0.043 cm). Therefore, the 12th-order averaging filter is routinely used in the rest of the numerical study. However, we realize that processing of data with a shorter sampling interval could be done with the averaging filter of a smaller order or without the averaging filter at all. The latter case would mean that we do not distinguish between average and point-wise accelerations. Naturally, such an approximation may be acceptable only in the context of residual accelerations.

Table 2. Description of the reference gravity field model, auxiliary orbits, and derived accelerations

	Noise-free case	Noisy case
Reference gravity field model	JGM-3 (Tapley et al. 1996)	JGM-3
Truncated at degree and order	50	70 (the full model)
RMS difference between the 'observed' and auxiliary orbit	6.797 m	8.697 m
Number of three-component residual accelerations	28 781	863 981
Maximum degree solved for	50	80

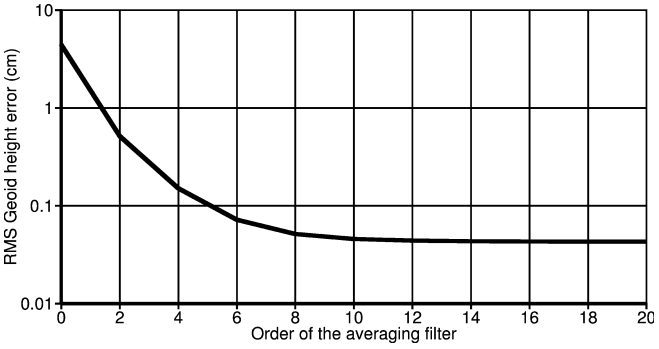

Fig. 2. Processing of the noise-free data set, Dependence of the RMS geoid height error on the order of the averaging filter. The zero order corresponds to the computation without averaging

Figure 3 shows the map of geoid height errors when the noise-free data set is processed with the 12th-order averaging filter. The remaining errors are probably caused by the limited accuracy of the orbit integration.

4.4 Optimal data weighting

4.4.1. Naive attempts

In this example, the noisy data set has been considered ($\Delta t = 1$ s). To be begin with, we have tried to process this set without any data weighting. The model obtained, however, shows strong edge effects at the beginning and at the end of each arc (Fig. 4a). These are caused by the fact that noise in accelerations is, roughly speaking, proportional to the frequency squared, i.e. severely colored. At high frequencies it can be very strong, especially if the sampling interval is short [remember that noise level is inversely proportional to the sampling interval squared, cf. Eq. (36)]. At lower frequencies, however, the noise rapidly decreases. This means that the averaging of a noise realization over even a short interval results in a near-zero value. The latter explains why the model obtained is surprisingly insensitive to the data noise in the areas that do not fall into the vicinity of an arc edge.

Furthermore, a naive attempt to apply a whitening filter as discussed in Sect. 3.2.4 fails (Fig. 4b). Increasing the filter half-width leads to even worse results.

4.4.2 Suppression of edge effects

We interpret the severe edge effects shown above as evidence that the averaging of the noise realization does not return a near-zero value at the edges. This is

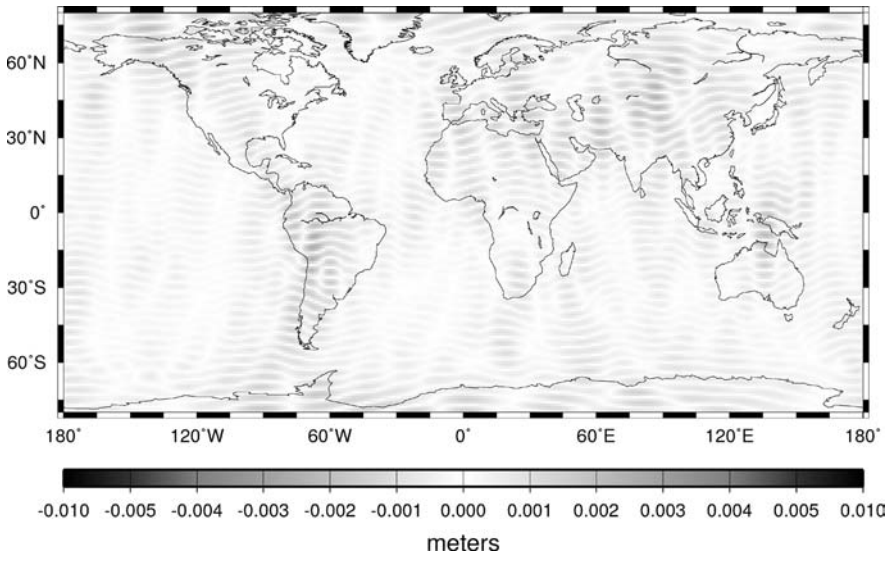

Fig. 3. Processing of the noise-free data set. Map of geoid height errors, 12th order averaging filter is applied (RMS error 0.044 cm, maximum error 0.26 cm)

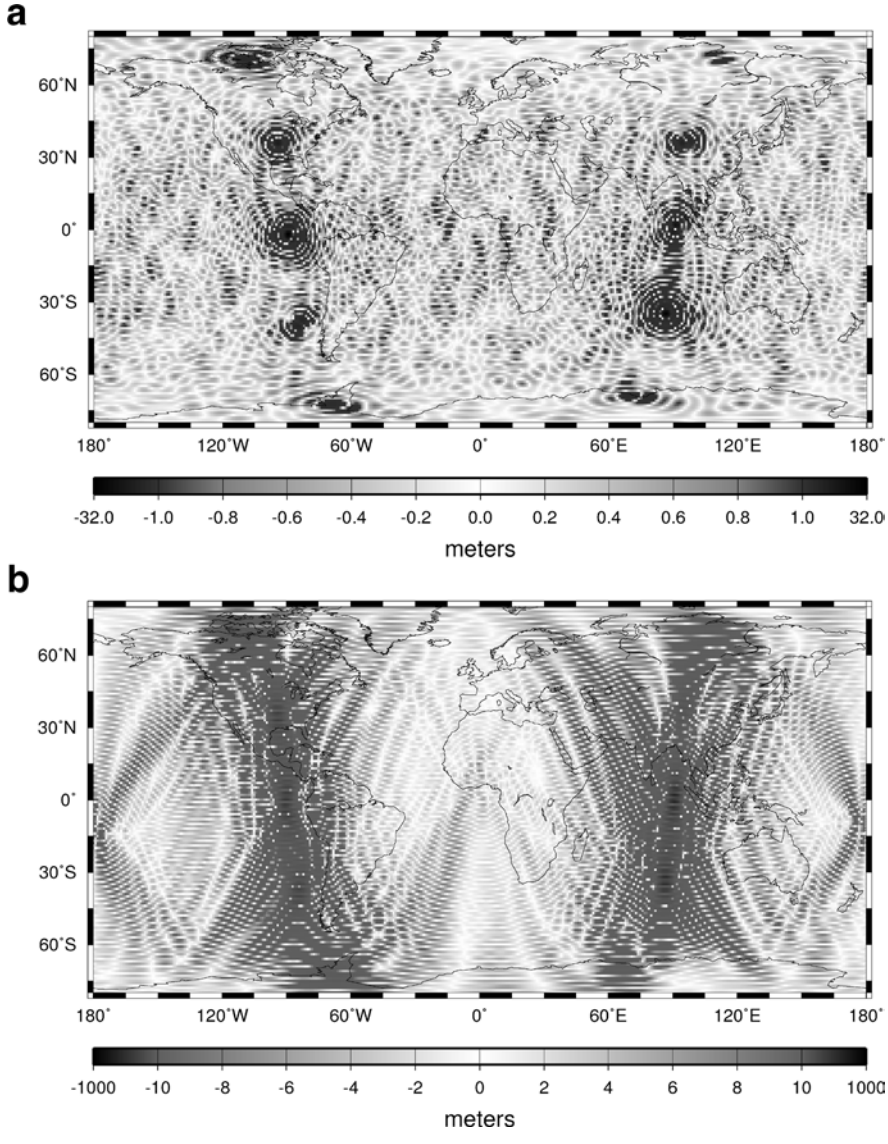

Fig. 4. Processing of the noisy data set. Maps of geoid height errors when **a** no data weighting is applied (RMS error 8.2 m, maximum error 31.3 m) and **b** the data are weighted by applying twice the whitening filter ($\tau = 180$ s) without suppression of edge effects (RMS error 30.7 m, maximum error 602.2 m)

not surprising because a part of noise samples that contributes to a near-zero sum is lost (remember the concept of an infinite data set and the 'mask matrix').

In view of this, we can try to compensate the noise 'imbalance' at an edge of an arc artificially. Consider as an example the very beginning of the data set. Let us append to the original data set one additional sample ('zero sample'). The value assigned to this sample should be such that an averaging would produce a near-zero result at the edge. In practice, this can be reached, e.g. by assuming that the whitening filter returns exactly zero for the zero sample. As long as the result of the filtering is a smooth function, we can expect that the filter will return near-zero values for the samples 1, 2, 3, ...

Obtaining the zero filter output for the zero sample is especially simple if the data set starts from a sufficiently long uninterrupted data fragment. Then, according to Eq. (52), the zero sample obtains after the filtering the following value:

$$z_0 = \frac{\Delta t \tau}{2\sigma} \sum_{j=0}^{\infty} y_j \mathrm{e}^{-j\alpha_c} = \frac{\Delta t \tau}{2\sigma} \left(y_0 + \sum_{j=1}^{\infty} y_j \mathrm{e}^{-j\alpha_c} \right) \tag{73}$$

Then, setting $z_0 = 0$ yields

$$y_0 = -\sum_{j=1}^{\infty} y_j \mathrm{e}^{-j\alpha_c} \tag{74}$$

A similar idea can also be used to handle the beginning and the end of each arc, including short arcs, although formulae in that case are slightly more complicated.

Restoration of the 'noise balance' improves the quality of the results dramatically. The filter half-width τ, however, should be kept rather short. Table 3 shows that the optimal model is obtained with $\tau = 60$ s. In an attempt to use a longer filter, we face the problem of edge effects again (Fig. 5a). Such a behavior is annoying because τ is inversely proportional to α_c – the regularization parameter applied to the covariance matrix. We could expect that by increasing τ we would introduce a smaller bias into the stochastic model, so that the quality of the obtained models should increase.

We can probably think of a better empirical procedure for the suppression of edge effects. There is, however, a more drastic remedy: to apply the exact data weighting scheme based on the low-level PCCG procedure as discussed in Sect. 3.2.5.

4.4.3 Exact data weighting

Indeed, the exact data weighting algorithm does a better job (Table 3, Fig. 5b). First, the errors in the models obtained are further reduced. Second, the dependence of the model error on the filter half-width τ is more reasonable: the model quality improves as τ increases, at least until $\tau = 600$ s. In practice, however, a somewhat smaller value of τ (namely, 180 s) seems to be preferable. This gives a model of almost the same quality as $\tau = 600$ s but requires three times fewer PCCG iterations (Table 3). It is worthwhile to add that setting $\tau > 600$ s does not make much sense because the further improvement of model quality is negligible, whereas time expenditures increase dramatically.

Interestingly, switching off the averaging filter does not introduce any noticeable error into the solution when the 1-s data set is processed. To show this, we have included in Table 3 the results of the corresponding computation with $\tau = 180$ s.

We have also assessed how important it is to perform the data weighting in an inertial frame rather than in the LORF. We have repeated the computations having switched off the rotation in the data weighting procedure [cf. Eq. (32)]. Results of these computations are listed in Table 3 only for $\tau \geq 180$ s, because for a smaller τ the differences with respect to the first series are negligible. We can see that data weighting in the LORF does not deteriorate the model, but may speed up the computations when τ is large. The latter feature is not surprising because in developing the pre-conditioner we neglected the rotation to/from an inertial frame. The comparison

Table 3. Dependence of the model quality and the numerical performance on the data weighting scheme and the parameter τ

Data weighting scheme	τ (s)	RMS geoid error (cm)	Max geoid error (cm)	Number of PCCG iterations	CPU time on a Pentium IV 2.4-GHz laptop PC (min)
No data weighting	–	81.63	3128	6	21.1
Double whitening filtering (edge effects are suppressed)	20	31.29	149	6	21.5
	40	30.41	139	6	21.5
	60	30.26	144	7	24.8
	120	30.53	206	9	31.2
	180	31.05	246	11	38.0
	600	34.11	424	34	120.6
Exact data weighting in an inertial frame	20	31.25	149	6	22.3
	40	30.27	140	7	25.6
	60	29.98	138	7	25.5
	180	29.67	137	11	39.3
	600	29.65	137	39	136.1
	180[a]	29.66[a]	137[a]	11	35.9
Exact data weighting in the LORF	180	29.68	136	10	35.4
	600	29.70	136	35	121.1

[a] The averaging filter was switched off.

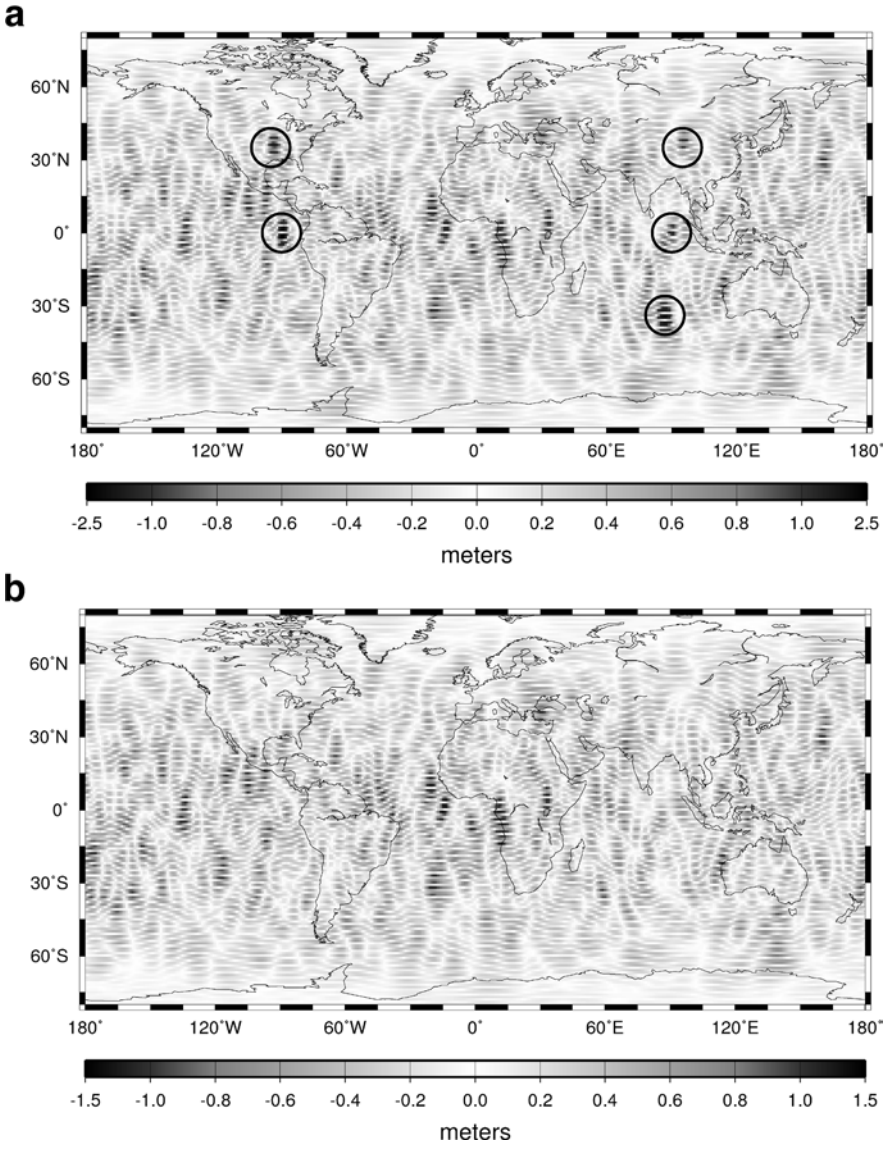

Fig. 5. Processing of the noisy data set; $\tau = 180$ s. Maps of geoid height errors when **a** the data are weighted by twice the whitening filter with suppression of edge effects (RMS error 0.31 m, maximum error 2.46 m) and **b** the exact (PCCG-based) data weighting is used (RMS error 0.30 m, maximum error 1.37 m). The circles on the top picture indicate the areas with remaining edge effects

shows that we can safely perform the data weighting in the LORF.

4.5 Comparison of the proposed technique with other approaches

Naturally, it is important to compare the results shown with those obtained by means of the other methods: the energy balance approach and the traditional approach based on the integration of the variational equations.

4.5.1 Energy balance approach

In the energy balance approach, the input data set consists of satellite kinetic energy measurements. In the absence of non-gravitational forces, the kinetic energy per unit mass $K(t)$ can be related to the total gravitational potential $U(t)$ by the energy conservation law (see e.g. Jekeli 1999)

$$K(t) = U(t) - \int_{t_0}^{t} \frac{\partial U(t)}{\partial t} \mathrm{d}t + \mathrm{const} \tag{75}$$

where t_0 is the time when an uninterrupted series of measurements started (each interruption leads to a re-definition of the constant term). The integral term in this expression describes the energy that is 'pumped' in or out because of temporal gravity variations at a given point in an inertial frame. Let us show that a set of kinetic energy measurements leads to nearly the same gravity field model as a set of along-track accelerations, provided that the optimal estimation procedure is followed in both cases.

The kinetic energy K at a given time is a function of the velocity magnitude v: $K(v) = v^2/2$. The differentiation of this relationship yields: $K'(v) = v \neq 0$. Therefore, a set of kinetic energy measurements and a set of velocity magnitude measurements are equivalent in the sense of the optimal estimation (see Appendix A).

Furthermore, the time derivative of the velocity magnitude results in the along-track component of the acceleration vector

$$\frac{\mathrm{d}v}{\mathrm{d}t} = a_x^{(L)} \tag{76}$$

which can be related to the gravitational potential by Newton's second law

$$a_x^{(L)} = \frac{\partial U}{\partial x^{(L)}} \tag{77}$$

We can also derive the relationship of Eq. (77) from Eq. (75) using the expression for the total derivative of the gravitational potential at the satellite location with respect to time

$$\frac{\mathrm{d}U}{\mathrm{d}t} = \frac{\partial U}{\partial x^{(L)}} v + \frac{\partial U}{\partial t} \tag{78}$$

Let us re-write Eq. (76) in the finite-difference form

$$\mathbf{a_x} = \mathbf{D}^{(1)}\mathbf{v} \tag{79}$$

where $\mathbf{v}$ is an uninterrupted series of velocity magnitude measurements; $\mathbf{a_x}$ is the result of the numerical differentiation; and $\mathbf{D}^{(1)}$ is the numerical differentiation matrix

$$\mathbf{D}^{(1)} = \frac{1}{\Delta t}\begin{pmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix} \tag{80}$$

If the matrix $\mathbf{D}^{(1)}$ was square and invertible, we could claim that the data sets v and $\mathbf{a_x}$ were also equivalent in the sense of the optimal estimation (see Appendix B). In reality, this is not the case, which is not surprising: the differentiation leads to a loss of information about the mean level in the original data. We can, however, extend the vector $\mathbf{a_x}$ by just one more value (e.g. the mean velocity), so that the equivalence of the data sets is restored. In practice, this additional measurement would mostly be needed to determine the unknown constant term in Eq. (75); its influence on the gravity field model would be minor. Therefore, a set of velocity magnitudes and a set of along-track accelerations derived by the numerical differentiation are nearly equivalent.

Thus, the kinetic energy measurements contain nearly the same information as measurements of only one – the along-track – acceleration component and should lead to nearly the same optimal model estimation. This has a simple physical interpretation. The presence of the radial and the cross-track components in the acceleration vector is caused by the forces acting in the corresponding directions. Such forces, however, do no work, because they are always perpendicular to the elementary path. Hence by measuring the kinetic energy we collect no information about the forces in the radial and the cross-track directions.

It is now obvious that we can assess the accuracy deliverable by the energy balance approach using the proposed processing technique provided that only the along-track component is retained in the data vector. Naturally, the data weighting can be performed only in the LORF in this case. We have made such a computation with the noisy data set, using $\tau = 180$ s and the exact data weighting scheme. The obtained pattern of geoid height errors turns out to be similar to that in Fig. 5b, but the error level has increased: to 50.34 cm on average and to 232 cm at maximum. This RMS error is 1.70 times larger than that we obtained when all three components of the acceleration vector were used (cf. Table 3). It is very close to the factor $\sqrt{3} \approx 1.73$ that could be expected from a purely statistical point of view (provided that all three acceleration components contribute equally to the solution).

One may argue that in practice Eq. (75) is not used for data processing directly. Instead, the integral term is replaced by an approximate estimation that can be obtained from the data as follows (Jekeli 1999)

$$\int_{t_0}^{t} \frac{\partial U(t)}{\partial t}\mathrm{d}t \approx -\omega_e\left(x^{(C)}(t)\frac{\mathrm{d}y^{(C)}(t)}{\mathrm{d}t} - y^{(C)}(t)\frac{\mathrm{d}x^{(C)}(t)}{\mathrm{d}t}\right) + \mathrm{const} \tag{81}$$

where ω_e is the Earth's rotation rate and $(x^{(c)}, y^{(c)})$ are X and Y coordinates of a point in the celestial frame. To make the comparison of the techniques more comprehensive, we have also implemented the energy balance approach explicitly. This required some modifications in the synthesis, co-synthesis, and pre-conditioning algorithms. The data weighting algorithm was almost unchanged but the original covariance matrix was replaced by the square root of it (we leave out a discussion of how justified this is). Some precautions were taken to account for an arbitrary constant term in each uninterrupted data fragment. Data averaging was not included, i.e. we do not distinguish between average and point-wise measurements. Our justification is that we deal with the 1-s sampling and with the residual potential, which is derived from a pair of close orbits. The developed procedure has been applied to the noisy data set too. As before, the results of the computations turn out to be fairly insensitive to the parameter τ. In particular, for $\tau = 180$ s we obtain an RMS geoid height error of 52.16 cm and a maximum error of 236 cm. The pattern of the geoid height errors is similar to that shown in Fig. 5b. This matches the results produced from the along-trade accelerations very well. Thus, we can see that the energy balance approach is indeed about $\sqrt{3}$ times less accurate than the proposed technique, which is based on the satellite accelerations.

4.5.2 Traditional approach (integration of variational equations)

The model obtained from the noise-free data set (Fig. 3) can be directly compared with the one produced in the traditional manner by Ditmar and Klees (2002). As far as the noisy data set is concerned, we have down-sampled the corresponding noisy orbit to 15 s, derived a new set of accelerations, and computed new gravity field

models, both without and with the first-order Tikhonov regularization. In the latter case, the regularization parameter has been selected in such a way as to minimize the RMS geoid height error. The models obtained in this way are comparable with those produced with the traditional approach by Ditmar et al. (2003b).

Results of the comparison in terms of geoid height errors and time expenditures are summarized in Table 4. Importantly, the proposed technique requires only the observation points and the orbit-derived accelerations as input. In applying the traditional technique, on the contrary, the design matrix was calculated by the GEODYN software beforehand; these time expenditures are, however, not included. Furthermore, the import of the computed design matrix is a rather time-consuming operation. Therefore, we compare only the time spent on the data processing itself, leaving out the input/output operations; nevertheless, the proposed technique proves to be much faster.

Furthermore, Table 4 shows that the proposed and the traditional techniques result in models of a very similar quality in all the considered examples. This is not surprising because the orbit data and the accelerations contain nearly the same information, i.e. are practically equivalent. We can easily show this using the same argumentation as in the previous section.

Interestingly, changing the sampling interval in the case of noisy data from 1 s to 15 s increases the RMS model error 3.8 times (from 29.7 to 114 cm). This is very close to the factor $\sqrt{15} \approx 3.9$ that could be expected from the purely statistical point of view. This confirms that the model errors have a purely stochastic origin, i.e. represent propagated data noise.

5 Discussion and conclusions

We have presented a new technique for a high-accuracy computation of the Earth's gravity field from orbit-derived satellite accelerations. The technique is based on the pre-conditioned conjugate gradient method, which allows us to avoid an explicit computation of the normal matrix and even of the design matrix (it is sufficient to have algorithms for multiplication of the design matrix and its transpose to vectors). Thanks to this, the technique is very fast. This is particularly true for long data sets. It has already been shown in the context of SGG data that such a techniques require only $O(N)$ operations per PCCG iteration provided that the number of data N is large (Ditmar et al. 2003a). On the other hand, the traditional technique always requires an explicit computation of the normal matrix. This is mainly because a suitable approximation of the normal matrix that could be used as the pre-conditioner in the PCCG procedure has not been found so far. The assembly of the normal matrix is, however, a time-consuming procedure: it requires $O(NL_{\max}^4)$ operations.

One may argue that the normal matrix is a valuable product of data processing on its own (e.g. because it can be used for computation of the model covariance matrix). However, we find it very uneconomical to generate the normal matrix in every data processing attempt. Furthermore, the technique we propose can easily be adapted to the computation of the normal matrix as well. First of all, we have to define a 'unit model vector', i.e. a model with all the coefficients set to 0 except for one, which equals 1. A sequential application of the synthesis, data weighting, and co-synthesis just results in one column of the normal matrix. All the columns can be computed in this way, one by one. At present, such a computation is a standard option of our software. Importantly, the number of operations per column is equal to that at one iteration of the PCCG scheme. Thus, the whole normal matrix can be assembled in $O(NL_{\max}^2)$ operations. As a result, the proposed technique remains, for large N, faster than the traditional one. A more detailed discussion about the computation of the normal matrix with the fast synthesis/co-synthesis algorithms can be found in Ditmar et al. (in press). It is worth adding that the covariance matrix, which is the inverse of the normal matrix, is usually required with a much lower accuracy than a model of the gravity field. This opens the door for various approximations in computing the normal matrix. Our technique offers a number of ways to speed up the computations at the expense of the accuracy: (1) using relatively large cells in the synthesis/co-synthesis on a grid; (2) setting the orbit radius constant, thanks to which the 3-D interpolation in the synthesis/co-synthesis reduces to a 2-D one; and (3) usage of an approximate

Table 4. Comparison of the proposed technique with the traditional one (based on the integration of the variational equations). The time expenditure refers only to the data processing itself, without input/output operations. In the case of the proposed technique, the exact data weighting is applied in an inertial frame with $\tau = 180$ s

Data set	Proposed technique				Traditional technique		
	RMS geoid error (cm)	Max geoid error (cm)	CPU time, on a Pentium IV 2.4-GHz laptop PC (s)	Wall-clock time, SGI Origin 38000 (s)	RMS geoid error (cm)	Max geoid error (cm)	Wall-clock time, SGI Origin 38000 (s)
Noise-free	0.044	0.26	50	7 (10 CPUs)	0.044	0.22	169 (10 CPUs)
Noisy (15-s sampling, without regularization)	114	492	379	26 (30 CPUs)	111	534	3422 (30 CPUfs)
Noisy (15-s sampling, with regularization)	46.1	337	345	23 (30 CPUs)	46.0	396	3374 (30 CPUs)

data weighting. Finally, we would like to point out that the entries in a covariance matrix can also be approximately estimated by means of a Monte-Carlo method (Gundlich et al. 2003).

The conducted numerical study has shown that the proposed technique is about $\sqrt{3}$ times more accurate than the energy balance approach. However, this result must be interpreted with caution. On the one hand, the performance of the energy balance approach can be even worse if the estimation procedure is not optimal (e.g. without a data weighting at all). On the other hand, the contribution of different components to the solution may be not the same. If, for some reason, the along-track measurements are much more accurate than the measurements related to the other directions, the energy balance approach may even deliver the same accuracy as the other methods. Furthermore, in practice the solutions are usually computed with a regularization, which can be considered as an additional source of information. Therefore, the relative loss of information due to the use of the energy balance approach can also be somewhat less than in the examples above. Finally, errors in the solution may be of a systematic rather than of a stochastic origin (e.g. because of unaccounted temporal gravity field variations). Switching from the energy balance approach to another data processing technique will not help to minimize those errors at all.

The proposed technique suggests that the input data – average satellite accelerations – are derived from a precise orbit with the three-point scheme. The proper adaptation of the functional model is performed by incorporation of the averaging filter into the processing procedure. Naturally, there is a more obvious way to define the functional model: to assume that the measured accelerations are point-wise (Reubelt et al. 2003a). We believe, however, that in practice such a functional model is inferior. First, derivation of (approximately) point-wise accelerations requires a high-order differentiation scheme. Therefore, even one missing observation in terms of orbit data causes a relatively broad gap in terms of accelerations. Second, designing an accurate and efficient data weighting procedure would be more difficult in the case of a high-order differentiation.

An important feature of orbit-derived accelerations is correlated noise. If the noise in orbit data is white, the noise in accelerations increases, roughly speaking, in proportion to the frequency squared. For a small sampling rate, the noise in the range of highest frequencies can exceed the signal by orders of magnitude. This explains why an accurate data weighting proved to be a key to the success of the proposed technique. Furthermore, our numerical study has shown that the quality of the obtained models is not very sensitive to the filter half-width τ. However, our results are probably valid only when data errors match the considered stochastic model. In practice, the situation may be more complicated due to various reasons, e.g.: (1) non-stationary (and, possibly, correlated) noise in the kinematic orbit; (2) colored noise in measurements of the non-gravitational satellite accelerations; and (3) systematic errors caused by an imperfect functional model. A stochastic model corresponding to arbitrary stationary colored noise has been already considered by Schuh (1996), Klees and Broersen (2002), Klees et al. (2003), and Klees and Ditmar (in press), who used recursive (ARMA) filters for the data weighting. More complicated stochastic models are, however, a matter for future studies.

In order to obtain the residual satellite accelerations, we have to subtract the reference accelerations from the observed ones. In our numerical study, we have computed the reference accelerations from an auxiliary orbit, which corresponds to the reference gravity field. The reference gravity field, however, can also be used for the direct computation of the reference accelerations. In doing so, we must compute the point-wise accelerations at the observation points and then apply the averaging. The simplest way to do the latter operation is to use the averaging filter as discussed in Sect. 3.1.2. We should bear in mind, however, that inaccuracies of the filtering procedure will be proportional to the signal filtered and that the total accelerations are orders of magnitude larger than the residual accelerations. Therefore, the statements made above about tolerable approximations in the averaging filtering will no longer be valid: a high-order filter will be a must. The direct computation of the reference accelerations is probably preferable when the motion of a satellite is influenced by non-gravitational forces, as is the case for the CHAMP and GRACE missions.

Another important aspect of the proposed technique is the convergence of the PCCG procedure. Our numerical study shows that the block-diagonal pre-conditioner described in Sect.3.3 leads to a sufficiently fast convergence. It may have been noticed, however, that repeat orbits were used in the numerical examples considered. Then, the question arises whether a fast convergence can also be reached for a non-repeat orbit as well. Our latest experience tells that this is indeed the case, provided that the data set is sufficiently long. This can be explained as follows. The block-diagonal pre-conditioner works efficiently when most of the normal matrix elements vanish, so that the structure of the matrix itself becomes nearly block-diagonal. Each element of the normal matrix is a scalar product of two filtered design matrix columns, each of which can be approximated by a linear combination of sinusoidal harmonics. A normal matrix element becomes equal to zero if (1) the sets of frequencies in the two columns do not coincide and (2) an integer number of cycles fit into one column for every frequency. It is just the repeat orbit configuration which makes the latter condition satisfied. If, however, the orbit is sufficiently long, so that the number of cycles is large, the product of two harmonics of different frequencies is close to zero anyway, no matter whether the number of cycles in a column is integer or not. On the other hand, it is important to point out that the pre-conditioner we have used so far is built under the assumption that the list of unknowns is formed by the spherical harmonic coefficients only. In practice, it may be necessary to include other unknown parameters into the inversion scheme, in which case it

will be more difficult to find a suitable pre-conditioner. This is also a subject of further investigations.

The developed technique has been incorporated into the GOCESOFT software, which is intended, first of all, to process the data that will be acquired by the GOCE satellite. This does not, however, exclude consideration of data from other satellite missions. In particular, processing of real SST data acquired by the CHAMP satellite is currently in progress. The obtained results will be presented and discussed in a forthcoming publication.

Appendix A

Invariance of a linearized least-square solution with respect to functional transformations of the data

Assume that a model vector $\mathbf{m}$ of length M has to be computed on the basis of a data vector $\mathbf{y}$ of length N, provided that there is a certain (possibly, non-linear) functional relationship between the model and the data: $\mathbf{y} = \mathbf{\Phi}(\mathbf{m})$. This relationship can be linearized by means of the Taylor expansion, where the terms of the second and higher order of smallness are neglected

$$\mathbf{y} = \mathbf{y_0} + \mathbf{A}(\mathbf{m} - \mathbf{m_0}) \tag{A1}$$

with $\mathbf{m_0}$ an initial model; $\mathbf{y}_0 = \mathbf{\Phi}(\mathbf{m_0})$, and $\mathbf{A}$ the matrix of partial derivatives (design matrix)

$$A_{ij} = \left.\frac{\partial \Phi_i(\mathbf{m})}{\partial m_j}\right|_{\mathbf{m}=\mathbf{m_0}} \tag{A2}$$

Equation (A1) can be re-written in a more compact form as

$$\mathbf{Ax} = \mathbf{d} \tag{A3}$$

where $\mathbf{x}$ is the model correction to be found, $\mathbf{x} = \mathbf{m} - \mathbf{m_0}$, and d are the data residuals; $\mathbf{d} = \mathbf{y} - \mathbf{y_0}$. Let $\mathbf{C_d}$ be the data covariance matrix. Then, the optimal estimation of the model can be found as follows [cf. Eqs. (11) and (12)]

$$\hat{\mathbf{x}} = (\mathbf{A^T C_d^{-1} A})^{-1} \mathbf{A^T C_d^{-1} d} \tag{A4}$$

Assume that a certain functional transformation is applied to the data prior to the model estimation

$$\tilde{y}_i = f(y_i) \quad i = 1, 2, \ldots, N \tag{A5}$$

where $\tilde{y}_i$ is a transformed observation and $f(y)$ is an arbitrary function (possibly non-linear).Variations of the transformed data $\boldsymbol{\delta}\tilde{\mathbf{y}}$ can be related to those of the original data $\boldsymbol{\delta}\mathbf{y}$ as $\boldsymbol{\delta}\tilde{\mathbf{y}} = \mathbf{D}\boldsymbol{\delta}\mathbf{y}$, where $\mathbf{D}$ is a diagonal matrix of derivatives

$$\mathbf{D} = \begin{pmatrix} f'(y_1) & & & \\ & f'(y_2) & & \\ & & \ddots & \\ & & & f'(y_N) \end{pmatrix} \tag{A6}$$

In the first-order approximation, the transformed data residuals $\tilde{\mathbf{d}}$ can be related to the original data residuals $\mathbf{d}$ as

$$\tilde{\mathbf{d}} = \mathbf{Dd} \tag{A7}$$

Furthermore, the transformed data are characterized by the following covariance matrix:

$$\mathbf{C_{\tilde{d}}} = \mathbf{DC_d D} \tag{A8}$$

The covariance matrix $\mathbf{C_{\tilde{d}}}$ is invertible if all the derivatives $f'(y_1), \ldots, f'(y_N)$ are not equal to zero

$$\mathbf{C_{\tilde{d}}}^{-1} = \mathbf{D}^{-1} \mathbf{C_d^{-1}} \mathbf{D}^{-1} \tag{A9}$$

The transformed data can be related to the model as

$$\tilde{\mathbf{y}} = \tilde{\mathbf{\Phi}}(\mathbf{m}) = \mathbf{f}(\mathbf{\Phi}(\mathbf{m})) \tag{A10}$$

Furthermore, a linearized relationship can be written by analogy with Eq. (A1)

$$\tilde{\mathbf{y}} = \tilde{\mathbf{y}}_\mathbf{0} + \tilde{\mathbf{A}}(\mathbf{m} - \mathbf{m_0}) \tag{A11}$$

where $\tilde{\mathbf{A}}$ is the transformed matrix of partial derivatives. By using the chain rule of differentiation, we can represent the elements of the matrix $\tilde{\mathbf{A}}$ as follows:

$$\tilde{A}_{ij} = \left.\frac{\partial \tilde{\Phi}_i(\mathbf{m})}{\partial m_j}\right|_{\mathbf{m}=\mathbf{m_0}} = f'(y_i) A_{ij} \tag{A12}$$

Equation (A12) can be re-written with the matrix notation as

$$\tilde{\mathbf{A}} = \mathbf{DA} \tag{A13}$$

The optimal model that can be derived from the transformed data is equal, by analogy with Eq. (A4), to

$$\hat{\tilde{\mathbf{x}}} = (\tilde{\mathbf{A}}^T \mathbf{C}_{\tilde{\mathbf{d}}}^{-1} \tilde{\mathbf{A}})^{-1} \tilde{\mathbf{A}}^T \mathbf{C}_{\tilde{\mathbf{d}}}^{-1} \tilde{\mathbf{d}} \tag{A14}$$

Substitution of Eqs. (A7), (A9), and (A13) into Eq. (A14) yields

$$\hat{\tilde{\mathbf{x}}} = \hat{\mathbf{x}} \tag{A15}$$

Thus, a linearized least-square solution is invariant with respect to functional transformations of the data, provided that the derivative of the transformation function is not equal to zero for all the data.

Appendix B

Invariance of a least-square solution with respect to linear transformations of the data vector

Assume that a data vector $\mathbf{d}$ is characterized by the covariance matrix $\mathbf{C_d}$ and related to an unknown model vector $\mathbf{x}$ by a linear functional model represented by Eq. (A3). Assume further that this functional model is subject to linear transformation, so that the transformed functional model is

$$\tilde{\tilde{\mathbf{A}}}\mathbf{x} = \tilde{\tilde{\mathbf{d}}} \tag{B1}$$

where $\tilde{\tilde{\mathbf{A}}}$ is the transformed design matrix and $\tilde{\tilde{\mathbf{d}}}$ is the transformed data vector

$$\tilde{\tilde{\mathbf{A}}} = \mathbf{BA}, \quad \tilde{\tilde{\mathbf{d}}} = \mathbf{Bd} \tag{B2}$$

with $\mathbf{B}$ an arbitrary square non-singular matrix, so that the matrix $\mathbf{B}^{-1}$ exists. The covariance matrix of the transformed data $\mathbf{C}_{\tilde{\tilde{\mathbf{d}}}}$ is

$$\mathbf{C}_{\tilde{\tilde{\mathbf{d}}}} = \mathbf{B}\mathbf{C}_{\mathbf{d}}\mathbf{B}^T \tag{B3}$$

whereas the inverse of it is equal to

$$\mathbf{C}_{\tilde{\tilde{\mathbf{d}}}}^{-1} = (\mathbf{B}^T)^{-1}\mathbf{C}_{\mathbf{d}}^{-1}\mathbf{B}^{-1} \tag{B4}$$

The optimal model that can be derived from the transformed data is given by

$$\hat{\tilde{\tilde{\mathbf{x}}}} = \left(\tilde{\tilde{\mathbf{A}}}^T \mathbf{C}_{\tilde{\tilde{\mathbf{d}}}}^{-1} \tilde{\tilde{\mathbf{A}}}\right)^{-1} \tilde{\tilde{\mathbf{A}}}^T \mathbf{C}_{\tilde{\tilde{\mathbf{d}}}}^{-1} \tilde{\tilde{\mathbf{d}}} \tag{B5}$$

Substitution of Eqs. (B2) and (B4) into Eq. (B5) yields

$$\hat{\tilde{\tilde{\mathbf{x}}}} = \hat{\mathbf{x}} \tag{B6}$$

where $\hat{\mathbf{x}}$ is the optimal solution related to original functional model, as given by Eq. (A4). Thus, a least-square solution is invariant with respect to a linear transformation of the data vector provided that the transformation matrix is square and not singular.

Acknowledgments. Professor R. Klees is thanked for support of the project and for numerous fruitful discussions. The authors are also thankful to Dr. J. Kusche for useful remarks and to Dr. E. Schrama – his solid background in satellite geodesy proved to be very helpful. A large number of valuable comments were made by Dr. S.-C. Han, Dr. P. Schwintzer, and an anonymous reviewer; their contribution is greatly acknowledged. The satellite orbits used in the numerical study were kindly provided by Dr. P. Visser (Aerospace Department, Delft University of Technology). Access to the SGI Origin 3800 computer was provided by Stichting Nationale Computerfaciliteiten (NCF), grant SG-027.

References

Barriot JP, Balmino G (1992) Estimation of local planetary gravity fields using line of sight gravity data and an integral operator. Icarus 99: 202–224

Bertsekas DP (1982) Constrained optimization and Lagrange multiplier methods. Academic Press, New York

Betti B, Sansò F (1989) The integrated approach to satellite geodesy. In: Sansò F, Rummel R (eds) Theory of satellite geodesy and gravity field determination. Lecture Notes in Earth Sciences, vol. 25. Springer, Berlin Heidelberg New York, pp 373–416

Colombo OL (1986) Notes on the mapping of the gravity field using satellite data. In: Sünkel H (ed) Mathematical and numerical techniques in physical geodesy. Lecture Notes in Earth Sciences, vol. 7. Springer, Berlin Heidelberg New York, pp 261–316

Colombo OL (1989) Advanced techniques for high-resolution mapping of the gravitational field. In: Sansò F, Rummel R (eds) Theory of satellite geodesy and gravity field determination. Lecture Notes in Earth Sciences, vol. 25. Springer, Berlin Heidelberg New York, pp 335–369

Davis PJ (1979) Circulant matrices. John Wiley, New York

Ditmar P, Klees R (2002) A method to compute the Earth's gravity field from SGG/SST data to be acquired by the GOCE satellite (available at: http://www.geo.tudelft.nl/fmr/publications-presentations/delftuniversitypress/book_GOCE_data_inversion.pdf). Delft University Press, Delft

Ditmar P, Klees R, Kostenko F (2003a) Fast and accurate computation of spherical harmonic coefficients from satellite gravity gradiometry data. J Geod 76: 690–705

Ditmar P, Visser P, Klees R (2003b) On the joint inversion of SGG and SST data from the GOCE mission (available at: http://www.copernicus.org/egu/adgeo/2003/l/adg-l–87.pdf). Adv Geosci 1: 87–94

Ditmar P, Kusche J, Klees R (2003c) Computation of spherical harmonic coefficients from gravity gradiometry data to be acquired by the GOCE satellite: regularization issues. J Geod 77: 465–477

Ditmar P, Klees R, Kostenko F (in press) Fast spherical harmonic synthesis and co-synthesis with applications in gravity field modeling. In: Proc Hotine–Marussi workshop, Matera, Italy, 17–21 June 2002

European Space Agency (1999) Gravity field and steady-state ocean circulation missions. Reports for mission selection. The four candidate Earth explorer core missions, SP-1233(1). European Space Agency, Noordwijk

Fengler MJ, Freeden W, Michel V (submitted) The Kaiserslautern multiscale geopotential model SWITCH-03 from orbit perturbations of the satellite CHAMP and its comparison to the models EGM96, UCPH2002_02_0.5, EIGEN-1s, and EIGEN-2. Geophys J Int

Gerlach C, Földvary L, Švehla D, Gruber T, Frommknecht B, Oberndorfer H, Peters T, Rotacher M, Rummel R, Sneeuw N, Steigenberger, P, Wermuth M (2003a) A CHAMP-only gravity field model from kinematic orbits using the energy balance approach. Poster presented at EGS-AGU-EUG Joint Assembly, Nice, 6–11 April

Gerlach C, Sneeuw N, Visser P, Švehla D (2003b) CHAMP gravity field recovery using the energy balance approach (available at: http://www.copernicus.org/egu/adgeo/2003/l/adg-l–73.pdf). Adv Geosci 1: 73–80

Gundlich B, Koch K-R, Kusche J (2003) Gibbs sampler for computing and propagating large covariance matrices. J Geod 77: 519–528

Han S-C, Jekeli C, Shum CK (2002) Efficient gravity field recovery using in situ disturbing potential observables from CHAMP. Geophys Res Lett 29(16): 36.1–36.4

Heiskanen WA, Moritz H (1984) Physical geodesy. Institute of Physical Geodesy, Technical University Graz, Austria

Hestenes MR, Stiefel E (1952) Methods of conjugate gradients for solving linear systems. J Res Nat Bur Stds 49: 409–436

Howe E, Stenseng L, Tscherning CC (2003) Analysis of one month of CHAMP state vector and accelerometer data for the recovery of the gravity potential (available at: http://www.copernicus.org/egu/adgeo/2003/l/adg-l–l.pdf). Adv Geosci 1: 1–4

Jekeli C (1999) The determination of gravitational potential differences from satellite-to-satellite tracking. Celest Mech Dynam Astron 75: 85–101

Kaula WM (1966) Theory of satellite geodesy. Blaisdell, Waltham, MA

Klees R, Broersen P (2002) How to handle colored noise in large least-squares problems. Building the optimal filter. Delft University Press, Delft

Klees R, Ditmar P (in press) How to handle colored noise in large least-squares problems in the presence of data gaps? In: Proc Hotine–Marussi workshop, Matera, Italy, 17–21 June 2002

Klees R, Ditmar P, Broersen P (2003) How to handle colored observation noise in large-scale least-squares problems. J Geod 76: 629–640

Klees R, Ditmar P, Kusche J (in press) Numerical techniques for large least-squares problems with application to GOCE. In: Proc Hotine–Marussi workshop, Matera, Italy, 17–21 June 2002

Koop R (1993) Global gravity field modeling using satellite gravity gradiometry. Publ Geodesy, New Series, no. 38, Nederlands Geodetic Comission, Delft

Koop R, Visser P, van den IJssel J, Bouman J, van Gelderen M (2000) Simulation of gravity gradients. In: GOCE end-to-end closed loop simulation. Final report. NIVR NRT no. 2703 TU. Delft Institute for Earth-Oriented Space Research (DEOS), Delft University of Technology, Delft, pp 25–33

Kusche J, Klees R (2002a) On the regularization problem in gravity field determination from satellite gradiometric data. In: Adam J, Schwarz K-P (eds.) Vistas for geodesy in the new millennium. International Association of Geodesy Symposia, vol 125. Springer, Berlin Heidelberg New York, pp 175–180

Kusche J, Klees R (2002b) Regularization of gravity field estimation from satellite gravity gradients. J Geod 76: 359–368

Lemoine FG, Kenyon SC, Factor JK, Trimmer RG, Pavlis NK, Chinn DS, Cox CM, Klosko SM, Luthcke SB, Torrence MH, Wang YM, Williamson RG, Palvis EC, Rapp RH, Olson TR (1998) The development of the joint NASA GSFC and the National Imagery and Mapping Agency (NIMA) geopotential model EGM96. NASA/TP-1998-206861 NASA GSFC, Greenbelt, MD

O'Keefe JA (1957) An application of Jacobi's integral to the motion of an Earth satellite. Astron J 62(1252): 265–266

Overhauser AW (1968) Analytic definition of curves and surfaces by parabolic blending. Tech rep SL68-40, Scientific research staff publication, Ford Motor Company, Detroit, MI

Pavlis DE, Moore D, Luo S, McCarthy JJ, Luthcke SB (1997) GEODYN operations manual, 5 vols. Hughes STX, Greenbelt, MD

Press WH, Flannery BP, Teukolsky SA, Vetterling WT (1992) Numerical recipes in Fortran: the art of scientific computing. Cambridge University Press, Cambridge

Reigber C, Balmino G, Schwintzer P, Biancale R, Bode A, Lemoine J-M, König R, Loyer S, Neumayer H, Marty J-C, Barthelmes F, Perosanz F, Zhu SY (2002) A high-quality global gravity field model from CHAMP GPS tracking data and accelerometry (EIGEN-1S). Geophys Res Lett 29(14): 37.1–37.4

Reigber C, Bock R, Forste C, Grunwaldt L, Jakowski N, Lühr H, Schwintzer P, Tilgner C (1996) CHAMP phase B executive summary. GeoForschungsZentrum, Potsdam, STR96/13

Reubelt T, Austen G, Grafarend EW (2003a) Harmonic analysis of the Earth's gravitational field by means of semi-continuous ephemerides of a low Earth orbiting GPS-tracked satellite. Case study: CHAMP. J Geod 77: 257–278

Reubelt T, Austen G, Grafarend EW (2003b) Space gravity spectroscopy– determination of the Earth's gravitational field by means of Newton interpolated LEO ephemeris. Case studies on dynamic (CHAMP Rapid Science Orbit) and kinematic orbits (available at: http://www.copernicus.org/egu/adgeo/2003/l/adgl–127.pdf). ARdv Geosci 1: 127–135

Ries JC, Eanes RJ, Shum CK, Watkins MM (1992) Progress in the determination of the gravitational coeffficient of the Earth. Geophys Res Lett 19(6): 529–531

Rowlands D, Marshall JA, McCarthy J, Moore D, Pavlis D, Rowton S, Luthcke S, Tsaoussi L (1995) GEODYN II system description, vols. 1–5, Contractor report. Hughes STX, Greenbelt, MD

Schäfer C (2001) Space gravity spectroscopy. Reihe C, Heft Nr 534, Deutsche Geodätische Kommission, München

Schrama E (1990) Gravity field error analysis: applications of GPS receivers and gradiometers on low orbiting platforms. NASA tech memo 100769, Greenbelt, MD

Schuh WD (1996) Tailored numerical solution strategies for the global determination of the Earth's gravity field. Folge 81, Mitteilungen der geodätischen Institute der Technischen Universität Graz, Graz

Seeber G (1993) Satellite geodesy. Walter de Gruyter, Berlin

Sneeuw NJ (1992) Representation coefficients and their use in satellite geodesy. Manuscr Geod 17: 117–123

Sneeuw NJ, Gerlach C, Švehla D, Gruber C (2002) A first attempt at time-variable gravity recovery from CHAMP using the energy balance approach (available at: http://olimpia.topo.auth.gr/gg2002/session3/sneeuw.pdf). In: Tziavos In (ed) Proc 3rd Meeting of the International Gravity and Geoid Commission, Thessaloniki, 26–30 August

Tapley BD (1997) The gravity recovery and climate experiment (GRACE). EOS Trans Am Geophys Un Suppl 78(46): 163

Tapley BD, Watkins MM, Reis JC, Davis GW, Eanes RJ, Poole SR, Rim HG, Schutz BE, Shum CK, Nerem RS, Lerch FJ, Marshall JA, Ulosko SM, Palvis NK, Williamson RG (1996) The joint gravity model 3. J Geophys Res B 101(B12): 28 029–28 049

Visser PNAM, van den IJssel J (2000) GPS-based precise orbit determination of the very low Earth-orbiting gravity mission GOCE. J Geod 74: 590–602

Visser PNAM, van den IJssel J, Koop R, Klees R (2001) Exploring gravity field determination from orbit perturbations of the European gravity mission GOCE. J Geod 75: 89–98

Voevodin VV, Tyrtyshnikov EE (1987) Vychislitel'nye processy s Toeplitzevymi matritsami (Computations with Toeplitz matrices, in Russian). Nauka, Moscow