Use of massive parallel computing libraries in the context of global gravity field determination

Jan Martin Brockmann and Wolf-Dieter Schuh

Institute of Geodesy and Geoinformation, Department of Theoretical Geodesy, University of Bonn, Germany

Abstract

The estimation of the global Earth's gravity field parameterized as a finite spherical harmonic series is computationally demanding. The computational effort depends on the one hand on the maximal resolution of the spherical harmonic expansion (i.e. the number of parameters to be estimated) and on the other hand on the number of the observations (which are several millions for e.g. observations from the GOCE satellite missions). To circumvent these restrictions, a massive parallel software based on high-performance computing (HPC) libraries as ScaLAPACK, PBLAS and BLACS was designed in the context of GOCE HPF WP6000 and the GOCO consortium. A prerequisite for the use of these libraries is that all matrices are block-cyclic distributed on a processor grid comprised by a large number of (distributed memory) computers. Using this set of standard HPC libraries has the benefit that once the matrices are distributed across the computer cluster, a huge set of efficient and highly scalable linear algebra operations can be used.

Grids of distributed memory computers (computer cluster): Naming conventions

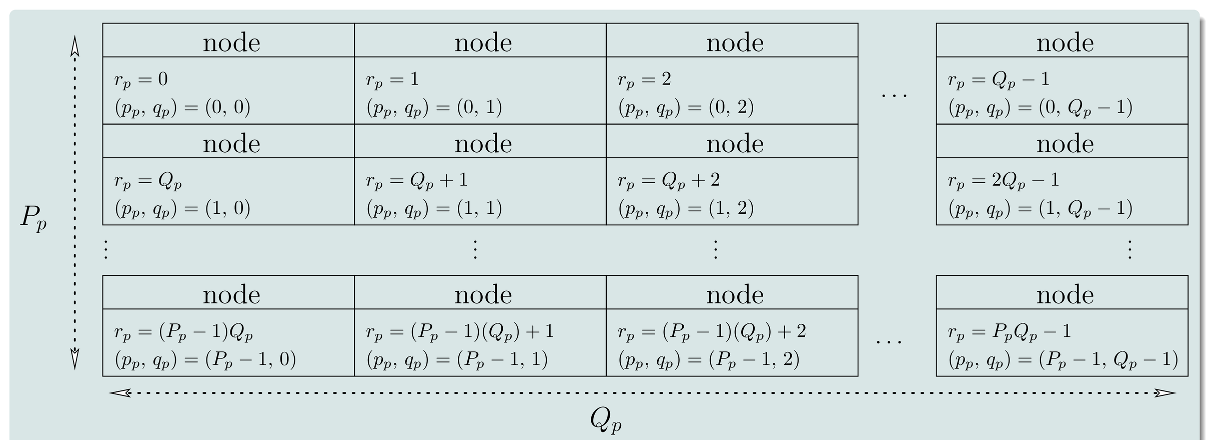

Fig. 1: Computing nodes & their adresses in a rectangular computing grid.

Assume a set of computers, which are connectet via a fast network. These computers form a distributed memory computer cluster. Each of them may contain several CPUs or cores. Every individual core of a computer is called computing node within this context. These computing nodes may be virtually arranged in a rectangular $P_p \times Q_p$ computing grid (cf. Fig. 1). A row of this grid is called node row and the set of nodes in the same column is called node column. These nodes are uniquely addressed via their 2-dimensional Cartesian coordinates (p_r, p_c) in the grid, or by their 1-dimesional rank r_p (counted row-wise, cf. Fig. 1).

Block-cyclic matrix distribution for the use in HPC libraries: An Example

A prerequisite for the use of HPC libraries such as PBLAS or ScaLAPACK is that the matrix to be operated on is block-cyclically distributed over the computing grid. To distribute a general matrix block-cyclically along the grid of computing nodes (cf. Fig. 2) the matrix is divided into patches of a given block size $b_r \times b_c$. These patches are cyclically distributed along a row of the node grid and cyclically along the columns of the grid (Blackford *et al.* (1997)).

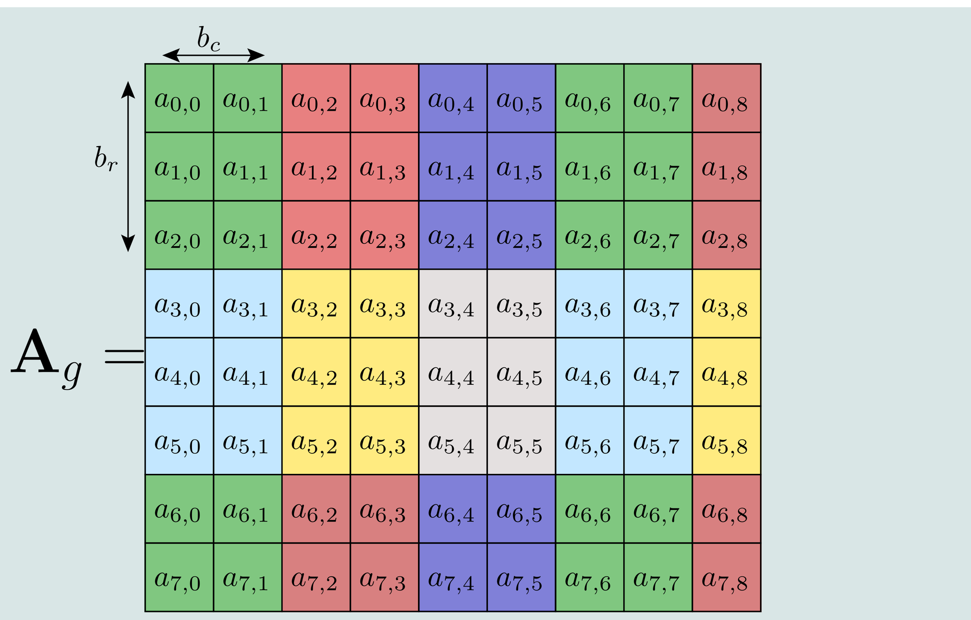

Fig. 2: Example Matrix of Dimension 8×9 which is block-cyclically distributed along a grid of 2×3 nodes with block size of 3×2.

To visualize the concept of the block-cyclic matrix distribution, a general matrix of size 8×9 (cf. Fig. 2) should be block-cyclically distributed on a 2×3 computing grid. The block size of the distribution is chosen as 3×2. The colors in Fig. 2 represent the computing nodes of the grid in Fig. 3 to which the matrix patch is distributed to.

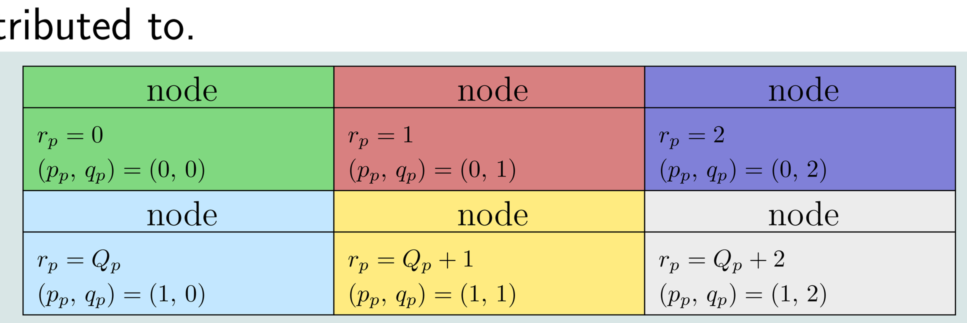

Fig. 3: Example computing grid of 6 computing nodes set up as a rectangular 2×3 computing grid.

The matrix patches are concatenated on each node patch by patch (row- and column-wise) into the local matrix $\mathbf{A}_l$ as sown for the example in Fig. 4.

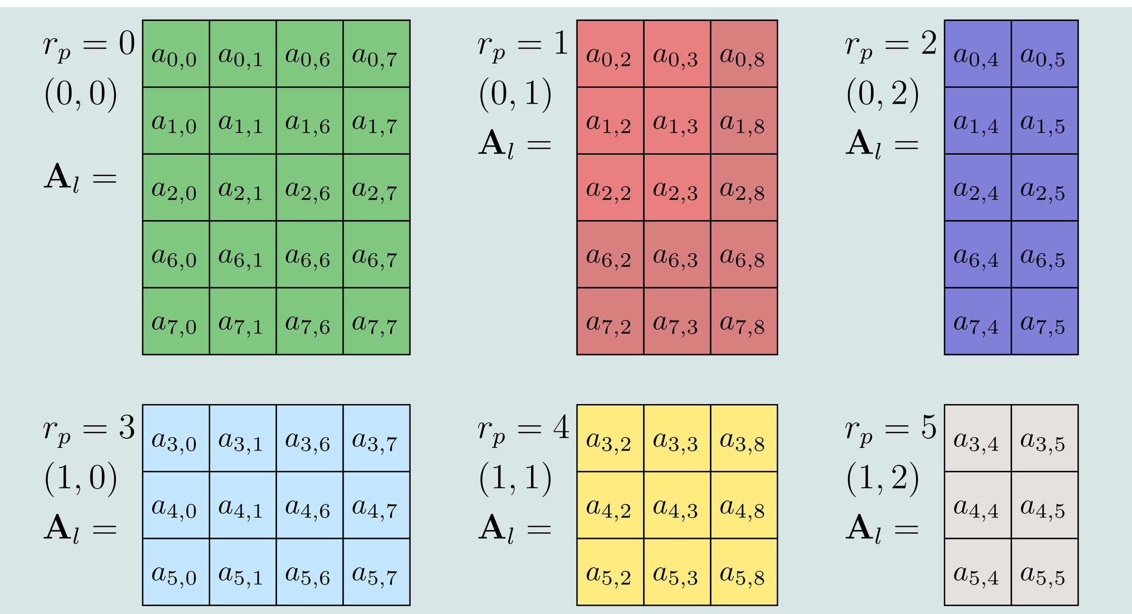

Fig. 4: Local matrices on each of the nodes (p_r, p_c) for the example.

On each node, the two-dimensional matrix needs to be mapped to the linear main memory of the computer, which is done by column major order (CMO), storing the matrix column by column to a one-dimensional ordinary array $\_\mathbf{a}_l$. For the example matrix, the resulting local arrays on each node are shown in Fig. 5.

Fig. 5: Local arrays $\_\mathbf{a}_l$ on each of the nodes (p_r, p_c) for the choosen matrix and distribution.

Description of a block-cyclically distributed matrix: Interface to PBLAS/ScaLAPACK

The data of the distributed matrix is stored locally in the 1-dimensional arrays $\_\mathbf{a}_l$. Thus, memory of every node of the cluster is used to store the complete matrix, thus huge matrices can be stored, although each node has just severel GB of main memory.

Interface to ScaLAPACK, PBLAS routines:

- Access to matrix data: address in the main memory of the 1st element of the local array: $\_\mathbf{a}_l(0) \Rightarrow$ differs on each node
- Description of distribution parameters, 9-element int-array: $\&\_\mathbf{d}_{\mathbf{A}_g} \Rightarrow$ same on each node (except 8th entry).

Estimating combined gravity fields from preprocessed normal equations using HPC libraries

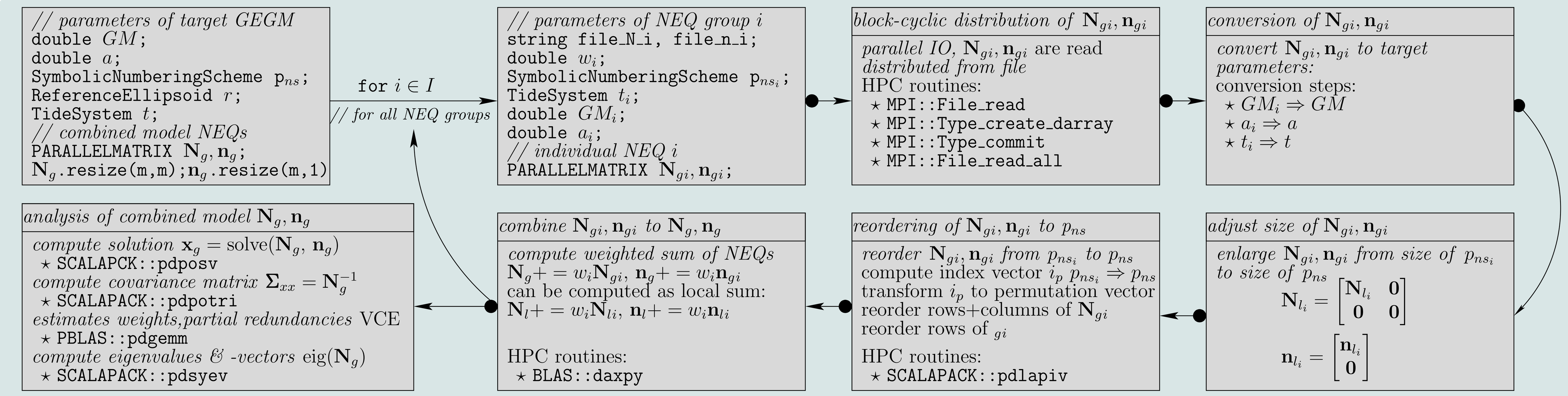

Fig. 6: Scheme of massive parallel programm for the combination of preprocessed gravity field normal equations.

Application: Estimated eigenvalues and eigenvectors for different GOCE based combination models

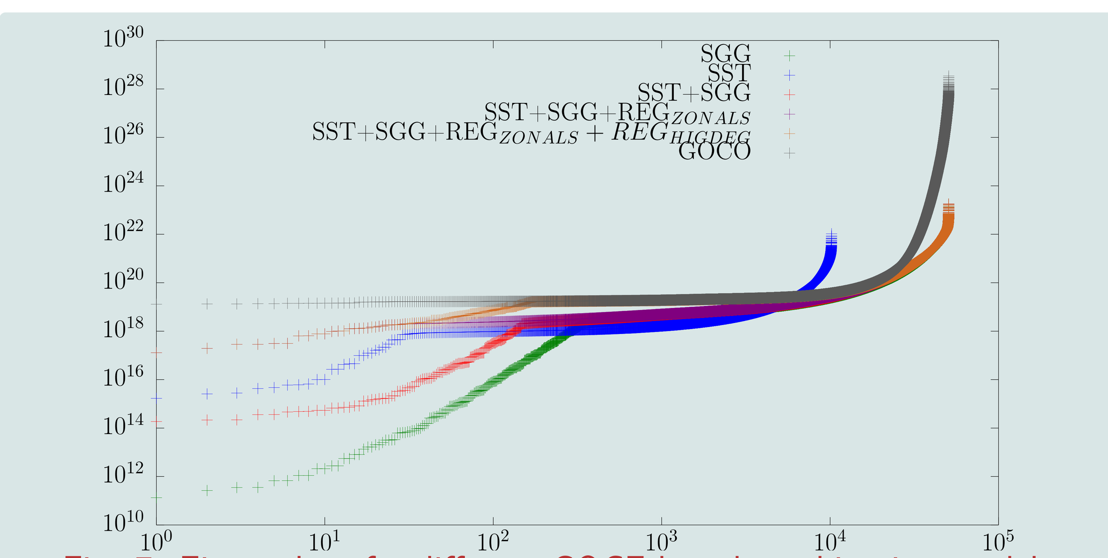

Fig. 7: Eigenvalues for different GOCE based combination models.

model:		
A	SGG	$\kappa = 1.3 \cdot 10^{12}$
B	SGG+SST	$\kappa = 9.6 \cdot 10^{8}$
C	SGG+SST+REG$_{\text{zonals}}$	$\kappa = 1.4 \cdot 10^{6}$
D	SGG+SST+REG$_{\text{zonals}}$+REG$_{\text{higdeg}}$	$\kappa = 1.4 \cdot 10^{6}$
E	GOCO	$\kappa = 2.6 \cdot 10^{9}$

- SGG, SST ≙ GOCE time-wise gradiometry & SST NEQ (RL01, 71days, Pail *et al.* (2010b))
- REG$_{\text{zonals}}$ ≙ Regularization of (near) zonal coefficients
- REG$_{\text{higdeg}}$ ≙ Regularization of high degree (> 170) coefficients
- GRACE ≙ ITG-Grace2010s NEQ (Meyer-Gürr *et al.* (2010))
- GOCO ≙ GOCO01S NEQ (Pail *et al.* (2010a))

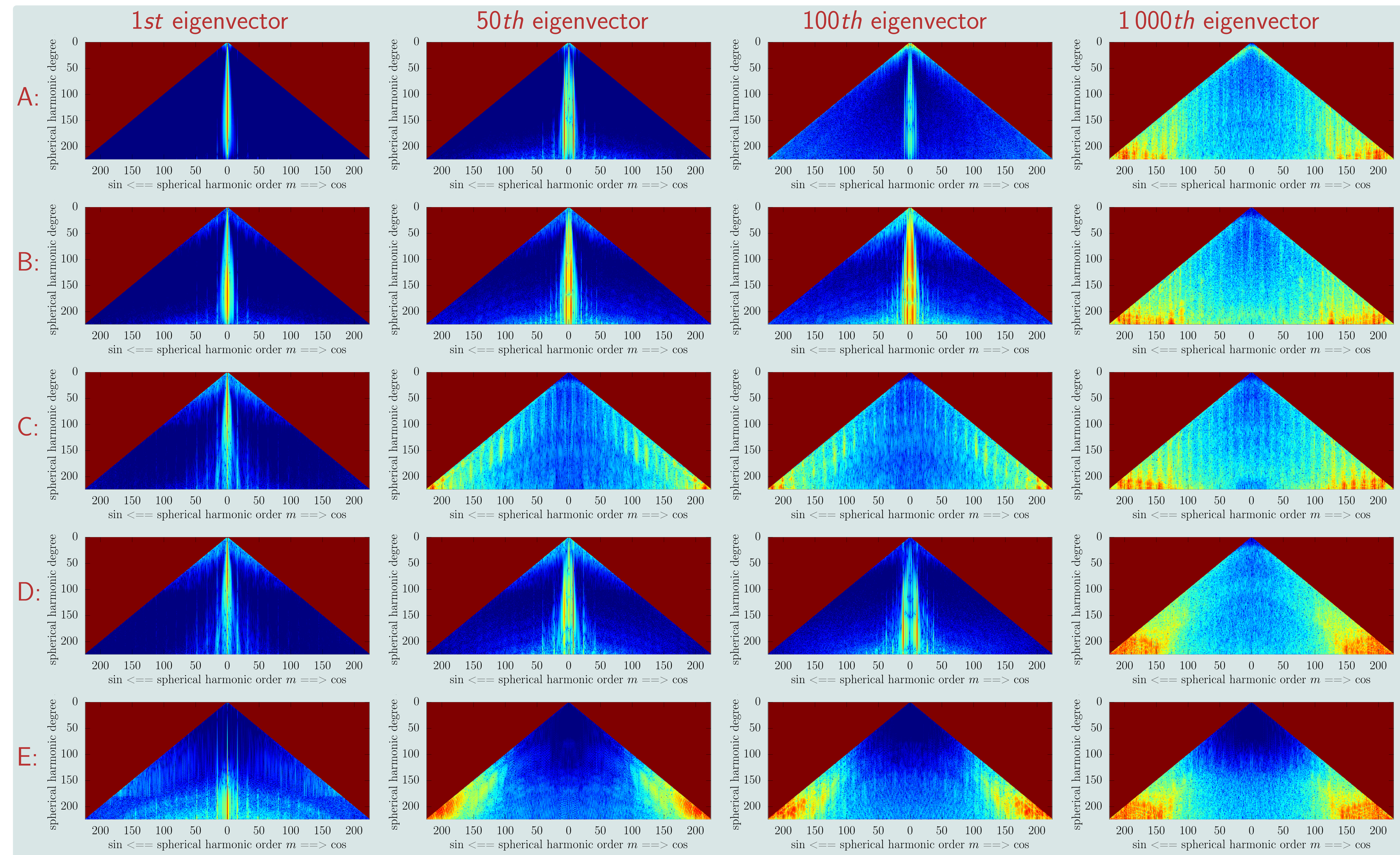

Fig. 8: Logarithmic elements of *i*. normalized eigenvector in the coefficient triangle for models A–E.

Acknowledgement

The authors acknowledge the European Space Agency for the provision of the GOCE data. Parts of this work were financially supported by the BMBF Geotechnologien program REAL-GOCE. The computations were performed on the JUROPA supercomputer at FZ Jülich, computing time was granted by John von Neumann Institute for Computing (project HBN15).

References

Blackford, L. S. et-al. (1997). *SCALAPACK Users Guide*. SIAM, Philadelphia, 2 edition.

Pail, R. et-al. (2010a).Combined satellite gravity field model GOCO01S derived from GOCE and GRACE. *Geophys. Res. Lett.*, 37, L20314.

Pail, R. et-al. (2010b). GOCE gravity field model derived from orbit and gradiometry data applying the time-wise approach. In *ESA Living Planet Symposium Bergen, Proceedings*.

Meyer-Gürr, T. et al. (2010). *ITG-Grace2010*. www.igg.uni-bonn.de/apmg/.

Institute of Geodesy and Geoinformation
Department of Theoretical Geodesy
University of Bonn

Contact:
Jan Martin Brockmann
brockmann@geod.uni-bonn.de

Nussallee 17
D-53115 Bonn
www.igg.uni-bonn.de/tg/