User’s Manual for CitcomSVE-3.0

Shijie Zhong and Tao Yuan
Department of Physics
University of Colorado at Boulder
Boulder, Colorado 80309
USA

szhong@colorado.edu
tao.yuan@colorado.edu

December 23, 2024

Table of Content

1. Introduction

1.1. What CitcomSVE is used for

CitcomSVE is a finite element package for computing Earth's (or planetary) mantle deformation and stress in response to surface loads and tidal loads. The computational domain is the Earth's mantle that is considered as an either incompressible or compressible viscoelastic medium. The surface loads that drive the deformation can be time-dependent growth and melting of ice sheets (or building-up of a shield volcano). The outputs of CitcomSVE include 3-D displacements, stress, strain rate, dissipation, and viscosity (when stress-dependent viscosity is considered) in the mantle, and surface displacements and gravity anomalies. CitcomSVE can be used for a global scale problem in a spherical shell geometry or for a 3-D regional scale problem in either Cartesian or spherical geometry. CitcomSVE utilizes MPI for parallel computing and works on massively parallel computers with 1000's CPU cores. CitcomSVE-3.0 represents the most updated version of the package.

1.2. History

CitcomSVE is developed from a finite element package CitcomS for mantle convection [Zhong et al., 2000; 2008]. The key difference between CitcomSVE and CitcomS is the mantle deformational property and its associated physical governing equations and numerical techniques. CitcomS considers the mantle as a purely viscous flow medium, while CitcomSVE treats the mantle as a viscoelastic medium. To incorporate elastic deformation, CitcomSVE implements a Lagrangian, deformable grid, while a Eulerian, fixed grid is used in CitcomS. However, CitcomSVE inherits the same 12-block spherical grid, finite element formulation, full multi-grid solver of the linear system of equation, and parallel computing strategies as CitcomS.

The first version of CitcomSVE was developed and used in Zhong et al. [2003] for modeling glacial isostatic adjustment process with laterally varying lithospheric thickness. Subsequently, Paulson et al. [2005] added solutions of sea level equation, polar wander effect and apparent motion of center-of-mass to CitcomSVE, together with 3-D mantle viscosity structure derived from seismic structure. A et al. [2013] further expanded the formulation to include mantle compressibility in CitcomSVE. Zhong et al., [2012] and Qin et al. [2014] implemented tidal-rotational potential forcing to the code. This line of CitcomSVE was developed out of the original version of CitcomS [Zhong et al., 2000] that permitted domain decomposition of the mantle into 12 blocks in horizontal directions for parallel computing purpose, although arbitrary domain divisions were permissible in the radial direction. As a result, CitcomSVE was only used for 12 CPUs in computations (Zhong et al. [2003], and A et al. [2013]). As CitcomS became one of publicly available community codes for mantle convection at CIG and as parallel computers became more powerful, the domain decomposition for CitcomS was improved such that more than 12 CPUs can be used in horizontal directions and that 1000's of CPUs can be used in CitcomS (e.g., Zhong et al. [2008]).

A recent effort was to re-develop and update CitcomSVE using the newer version of CitcomS (version 3.1) such that CitcomSVE can also use 1000's of CPUs [Zhong et al., 2022].

CitcomSVE-2.1 for an incompressible mantle was first released as open-source code to github in the spring of 2022, together with *User's Manual for CitcomSVE v2.1* and benchmark paper [Zhong et al., 2022]. CitcomSVE-2.1 was recently updated at github in October of 2024 with an added feature of calculations of a periodic tidal loading and tidal dissipation [Fienga et al., 2024].

The current document describes CitcomSVE-3.0 that works for both compressible and incompressible mantle on a global scale. This version also includes as pre- and post-processing procedures for a new implementation of sea level equation solver [Kendall et al., 2005], compared with that of earlier versions of CitcomSVE. Benchmark calculations using CitcomSVE-3.0 can be found in Yuan et al., [2024]. Future versions will include other features including grid refinements in horizontal directions and also models on a regional scale.

1.3. Governing equations, numerical method and grids

In CitcomSVE, the mantle is considered as a viscoelastic medium with Maxwell rheology (i.e., a spring and dashpot connected in series). The mantle can be treated as either incompressible or compressible, which will lead to a slightly different governing equations [e.g., Zhong et al., 2003; A et al., 2013]. The governing equations include the conservation law of the momentum (i.e., the equation of the motion), a Poisson's equation for gravitational potential perturbation and a rheological equation between stress and displacement [e.g., A et al., 2013]. For an incompressible medium, an additional equation is the conservation of the mass [e.g., Zhong et al., 2003; 2022].

For details of the governing equations, their finite element analysis and solutions, and calculations for sea-level equations, polar wander, and apparent motion of center-of-mass, readers can find them in Zhong et al. [2022] that provides a comprehensive summary of the developmental work of CitcomSVE in Zhong et al. [2003], Paulson et al. [2005] and A et al. [2013]. For formulation of a compressible mantle, see A et al., [2013]. The updated sea-level equation solver information can be found in Yuan et al., [2024]. For tidal loading calculations, see Zhong et al., [2012] and Fienga et al., [2024] with a step-function and periodic time function, respectively. For 3-D models of a regional scale, see Zhong and Watts [2013] and Bellas and Zhong [2021].

It is helpful to briefly describe the computational domain, domain decomposition and finite grids used in CitcomSVE or CitcomS for the global scale model. The mantle of a spherical shell is divided into 12 spherical blocks or caps, and each cap is further divided into finite elements or grids of similar sizes (Fig. 1, Zhong et al., [2022, 2000]). Each of the 12 caps can be further decomposed into smaller blocks in horizontal and radial directions for parallel computing purpose with each block for a CPU core. Message passing interface (MPI) routine calls were implemented in CitcomS/CitcomSVE for communications between different blocks or CPU cores for parallel computing [Zhong et al., 1998; 2000]. CitcomS/CitcomSVE uses 8-node tri-linear elements, inherited from the original Cartesian Citcom code [Moresi and Gurnis, 1996], which leads to solutions of displacement (or velocity) field to be second-order accurate, as confirmed by error convergence analyses for CitcomS [Zhong et al., 2008] and CitcomSVE [Zhong et al., 2022].

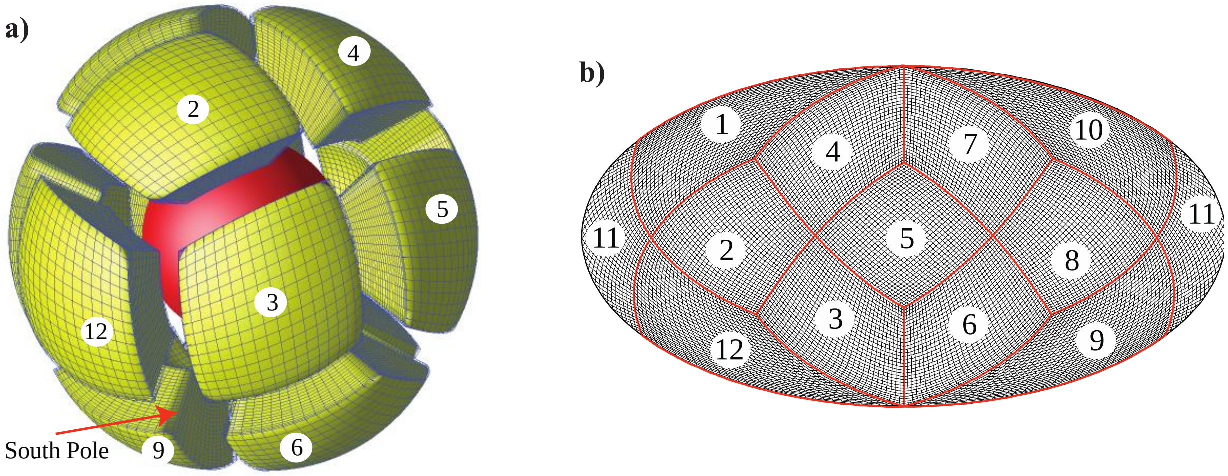

Figure 1. A 3-D view (a) and mapview (b) of the finite element grid with 12 spherical caps covering a spherical shell used in CitcomSVE. Each of the 12 caps in Figure 1a and 1b includes a certain number of finite elements in the radial direction and two azimuthal directions, and each cap can also be further divided into smaller blocks for a parallel computing purpose. The thick red lines in Figure 1b mark the shared boundaries of the 12 caps. In Figure 1a, the caps are separated for a better view. The numbers in Figure 1a and 1b show the ordering of the 12 caps. Figure 1a and 1b are modified from Zhong et al., [2022, 2000].

1.4. Acknowledgement, Citations and User's support.

Some of the important development of CitcomSVE has been done by former and current graduate students in Shijie Zhong's group: Archie Paulson, Geruo A and Tao Yuan. CitcomSVE's development had also been benefitted from the guidance and inspiration of Late Professor John Wahr. NSF, NASA and Packard Foundation have provided financial support for the development of CitcomSVE. The development of CitcomS from the original Citcom code that works for a Cartesian box on a single CPU computer [Moresi and Gurnis, 1996] was benefited from help by Louis Moresi, Michael Gurnis, Allen McNamara and Eh Tan [Zhong et al., 2000; 2008].

Users of CitcomSVE may cite Zhong et al., [2022] as a general reference. To use CitcomSVE-3.0 for a compressible mantle and sea-level calculations, users may cite Yuan et al., [2024] and A et al. [2013], respectively. If you have questions on CitcomSVE or would like to report a bug, you may contact Shijie Zhong at szhong@colorado.edu.

2. Installation of CitcomSVE

CitcomSVE has been tested and used successfully on many different Unix-like computer platforms including PC-clusters and also massively parallel supercomputers (e.g., NCAR's Derecho). CitcomSVE is developed out of CitcomS 3.1, and the installation process is the same as that for CitcomS. Here, a brief description is given on the installation procedures, and it should be sufficient for most users.

To use CitcomSVE effectively, you may want to work on a parallel computer, either a PC-Cluster or a supercomputer. The only system requirements for your computer are: 1) a C compiler and 2) a MPI library (i.e., no need for any other specialized libraries and packages). Most computer systems have them installed already. If not sure, you can ask your computer system administrator.

Here are the three steps to install CitcomSVE on your computer.

1) Download CitcomSVE-3.0 package from https://github.com/shjzhong/CitcomSVE. You will need to download ***three compressed files***: CitcomSVE-3.0.tar.gz, ICE6G_1x1.tar.gz, and Compare_Cookbook.tar.gz. Make a new project directory, move CitcomSVE-3.0.tar.gz to the new directory, and unpack it first. To unpack, use the tar command:

```
$ tar xzf CitcomSVE-3.0.tar.gz
```

The source codes and all the other files in CitcomSVE package are in directory *CitcomSVE-3.0*.

2) Go to directory *CitcomSVE-3.0* with CitcomSVE-3.0 source code, configure the compiler, and generate executables.

```
$ cd CitcomSVE-3.0
$ ./configure
$ make
```

An executable called CitcomSFull is generated in *CitcomSVE-3.0/bin*, and it will be used for modeling calculations. Notice that three directories under *CitcomSVE-3.0* are quite useful: *lib, bin* and *DATA*. Most of the source codes (i.e., the C subroutines *.c and head files *.h) are in *CitcomSVE-3.0/lib*, while several others are in *CitcomSVE-3.0/bin*, together with executable CitcomSFull. *CitcomSVE-3.0/DATA* stores some model input files.

3) Move the other two compressed files ICE6G_1x1.tar.gz and Compare_Cookbook.tar.gz to *CitcomSVE-3.0/DATA* and unpack them both. In directory *CitcomSVE-3.0/DATA*, execute these two commands:

```
$ tar xzf ICE6G_1x1.tar.gz
$ tar xzf Compare_Cookbook.tar.gz
```

As a result, two new directories are now generated: *CitcomSVE-3.0/DATA/ICE6G_1x1 and CitcomSVE-3.0/DATA/Compare_Cookbook.* The first directory stores relevant files for ice history model ICE6G and the second one contains files for cookbook example cases. The files from both the directories will be discussed later. Note that the release consists of three compressed files (not a single file), because this practice reduces the file size to fit the limit of 25 Mbytes set by github.

3. Running CitcomSVE

Running CitcomSVE would require input files for model calculations, which can vary depending on the calculations one is interested in running. How to run CitcomSVE also depends on the computer systems you use. The description here focuses on running CitcomSVE on a PC-cluster using mpirun command (A supercomputer center may only allow a user to run jobs in a batch mode and ask a system administrator to help with the batch mode). The example model calculation discussed here is for a glacial isostatic adjustment (GIA) modeling with an ice history model (i.e., ICE-6G), 1-D PREM and viscosity structure (i.e., VM5a). It is case GIA_R2 in Yuan et al., [2024].

Here is an example command line for running executable CitcomSFull on a PC cluster.

Go to directory *CitcomSVE-3.0/DATA* with input files including *inputfile6G_VM5a1* and *machinefile0*.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfile6G_VM5a1
```

Note that "-np 96" indicates that for this run, the number of processors or cores (i.e., np) to be used is 96, and "-machinefile machinefile0" indicates that the run will use compute nodes of the PC cluster that is listed in file *machinefile0*. File *machinefile0* lists 24 compute nodes of the PC-Cluster and contains the follow 24 lines:

node01
node02
…
node24

This example run uses 96 processors in total on 24 compute nodes. Therefore, 4 processors from each of the 24 compute nodes (from node01 to node24) as listed in file *machinefile0* is used for computations. Note that each compute node of this PC-cluster has 4 processors or cores. The number of processors used for each model run is specified in input file *inputfile6G_VM5a1*, for example,

Input file *inputfile6G_VM5a1* contains all the important information about this model including grid, mantle parameters, ice history models, …, all of which will be discussed in detail in next section.

This mpirun command, once issued, will start the CitcomSVE run. Output files (e.g., for surface vertical displacements and gravitational potential) are to be generated and stored during the model run in directories that you specify in *inputfile6G_VM5a1*.

4. Input files

Two types of input files are needed for computing GIA models using CitcomSVE: 1) a general input file (e.g., inputfile6G_VM5a1 in mpirun command line in the last section) and 2) files for ice history models (e.g., ICE6G). This section describes these two types of input files. Note that CitcomSVE package that you have downloaded from gitbub contains sample input files like *inputfile6G_VM5a1* and also ICE6G.

4.1. The input file in the mpirun command line

The input file in mpirun command line (e.g., *inputfile6G_VM5a1*) provides the most essential information for a CitcomSVE model run. This sub-section describes an example of such input file *inputfile6G_VM5a1*. The lines (with italic font) from *inputfile6G_VM5a1* are listed first, and explanation to these lines is provided using regular font. The input information in the input file is divided into 5 groups.

Group 1 is on general information of the modeling.

 # Group 1: General information
datadir="BM_ICE6G"
datafile="case1"

BM_ICE6G is the directory in which output files are stored, and it is under directory *DATA*. Parameter *datafile* (i.e., *case1*) specifies the prefix of output file names. For example, output log file will be *DATA/BM_ICE6G/case1.log*. Directory *BM_ICE6G* should be created in directory *DATA* before the model run starts.

compressible=1

Parameter *compressible=1* sets that the mantle is a compressible medium. When *compressible=0,* the mantle is an incompressible medium.

apply_SLE=1
polar_wander=1
polar_wander_kf=0.952109106687

Parameter *apply_SLE* determines if the sea-level equation is solved (1 means yes, and 0 is no). Parameter *polar_wander* determines if polar wander effect is included (1 means yes, and 0 is no). Parameter *polar_wander_kf* specifies degree-2 fluid Love number used for calculations. For GIA problems, both *apply_SLE* and *polar_wander* should be set as 1.

```
#output information
storage_spacing_surface=10          # write data on the surface every ...
storage_vol=1          #0: no vol. data output, 1: stress/visc, 2: 1+ disp. 3: 1+ stress tensor
storage_spacing_vol=20    #write volume data every .., only relevant for storage_vol=1
```

Parameter *storage_spacing_surface=10* means that the results of surface displacements and gravity anomalies will be output every 10 time-steps. If parameter *storage_vol=0,* then no 3-D volume data (e.g., viscosity, 2nd-invariant of stress, …) will be output. Parameter *storage_vol=1* indicates that 3-D volume data (e.g., viscosity, 2nd-invariant of stress) are output at time step specified by parameter *storage_spacing_vol=20* (i.e., every 20 time steps). Parameter *storage_vol=2* indicates that in addition to viscosity and 2nd invariant of stress, 3-D nodal displacements will be output as well. Parameter *storage_vol=3* indicates that in addition to viscosity and 2nd invariant of stress, 3-D stress-tensor (i.e., 6 components) at each nodal point will be output. Note that 3-D volume data take a lot of disk space and should be output less frequently. Also, note that the final time-step results are always output.

ll_max=32
output_ll_max=32

Parameter *ll_max=32* sets the highest number of spherical harmonic degree and order (i.e., 32 here in this example) with which the gravitational potential anomalies are computed. Parameter *output_ll_max=32* is that for the outputs. In most cases, these two parameters are the same.

Group 2 is on loading and loads.

```
# Group 2: On loads
# only for Heaviside=1 or step function loading
Heaviside=2    # 1: Heaviside   2: ice model e.g., ice6g     3: Sinusoidal in time
```

Parameter *Heaviside* determines loading time history. *Heaviside=1* specifies a step-function in time (i.e., the Heaviside function), and *Heaviside=3* is for sinusoidal function in time, both for loading at a single spherical harmonic, as to be discussed for other loading calculation cases of different input files. *Heaviside=2* is for more complicate loading history, e.g., ICE6G, for which more information is needed as discussed below.

```
# Loading harmonics and amplitude for parameter Heaviside=1 or 3
apply_potential=0      #0: apply surface loads, or 1:apply tidal potential
```

Parameter *apply_potential=0* is for surface loading such as ICE6G loading. For tidal loading with an applied potential, *apply_potential=1*, which is often used together with *Heaviside=1*.

```
# Loading harmonics and amplitude for parameter Heaviside=1 or 3
perturbmag=0.0        # e.g., 1e-6
perturbl=2
perturbm=0
```

These three lines set the loading harmonic degree l in *perturbl* and order m in *perturbm* with loading amplitude in *perturbmag*, for single harmonic loading with *Heaviside=1*. Note that the amplitude is normalized. For surface loading, the amplitude is the normalized height of the load with the density of the mantle (normalized by the radius of the planet). For tidal potential loading, the amplitude is the normalized potential (see Zhong et al., [2022] on how the potential is normalized).

load_stages_time_file="ice6g_26ka_stages_timeA.dat"

load_stages_time_file specifies the file (i.e., *ice6g_26ka_stages_timeA.dat*) that determines the loading history including the time duration of loading (e.g., 26,000 years) and the number of time-steps (e.g., 500 time-steps). See section 4.3.1 for description of this file.

 # ice model information, only useful for Heaviside=2
ice_datafile="ICE6G_1x1/ice6g"
ice_sph_or_grid=0   # 0: regular grid (e.g., 1x1) 1: spherical harmonic Jackson form
ocean_datafile="ICE6G_1x1/ocn6g"

When *Heaviside=2*, i.e., for ice loading, *ice_datafile and ocean_datafile* specify the prefixes of files for spatial and temporal distributions of ice and ocean function, respectively. Parameter *ice_sph_or_grid* sets how the ice model is read between two options: ice thickness on a regular grid (e.g., 1 degree by 1 degree) with *ice_sph_or_grid=0* or as coefficients of spherical harmonic expansion with *ice_sph_or_grid=1.* See section 4.3.2 for description of these files. Ocean functions can only be read on a regular grid. Note that the regular grids for ice models are different from CitcomSVE's finite element grids (Fig. 1), and CitcomSVE code reads the ice history and ocean functions on the regular grids and interpolate them to finite element grids. Note that these ice and ocean function files are in *CitcomSVE-2.1/DATA/ICE6G_1x1* which come from unpacking compressed *ICE6G_1x1.tar.gz*, as discussed in section 2.

Group 3 is on loading and loads.

 # Group 3: On mantle and crustal visco-elastic structure
 # for viscosity/rheology and elastic/density structure
1Dmodel_datafile="model.PREM.vm5a.wrtCrust"   #earth_model
1Dmodel_g_from_file=0

1Dmodel_datafile gives the file that specifies the basic 1-D density, shear modulus, bulk modulus, gravitational acceleration, viscosity structures (e.g., PREM and VM5a) and material group. The details of this file will be described in section 4.2. Parameter *1Dmodel_g_from_file=0* indicates that gravitational acceleration g in the mantle as a function of radius is computed using the mantle densities given in the file set by *1Dmodel_datafile* (i.e., *model.PREM.vm5a.wrtCrust*) and the core density in inputfile6G_VM5a1. If *1Dmodel_g_from_file=1*, gravitational acceleration g in the mantle is set directly by a column in file by *1Dmodel_datafile* (i.e., not computed from the densities). The latter scenario is useful for getting the correct g for simplified mantle density models such as in incompressible mantle models (see example cases in section 6.2).

3d_visc_from_file=0   #0 for 1d layered viscosity; 1 for 3d visc from files
3d_visc_datafile=" VE3D_A/ViscA2x2.dat"

Parameter *3d_visc_from_file* determines the viscosity structure used in the model calculation. *3d_visc_from_file=0* is for 1-D layered viscosity, while *3d_visc_from_file=1* is for 3-D viscosity structure. For the latter case, parameter *3d_visc_datafile="VE3D_A/ViscA2x2.dat"* specifies the file in which 3-D viscosity structure is given. See section 4.2 for description of the viscosity file.

3d_elastic_from_file=0   #0 for 1d layered ; 1 for 3d elastic from file
3d_elastic_datafile="VE3D_A/elastic2x2.dat"

Similarly, parameter *3d_elastic_from_file* determines the elastic and density structure used in the model calculation. *3d_elastic_from_file=0* indicates for 1-D layered structure (e.g., PREM), while *3d_elastic_from_file=1* is for 3-D elastic structures. For this latter case, *3d_elastic_datafile="VE3D_A/elastic2x2.dat"* specifies the file in which 3-D elastic structure is given. See section 4.2 for description of the elastic structure.

VMIN=on
visc_min=1.0e-3
VMAX=on
visc_max=1.0e05

Parameter *VMIN* and *visc_min* specify the dimensionless minimum viscosity cutoff (i.e., 10^{-3} in this example), and *VMAX* and *visc_max* give the dimensionless maximum viscosity cutoff (i.e., 10^5 here), both of which are normalized by the reference viscosity given in this input file (see discussion on dimensional parameters in two pages).

SDEPV=off   # on for stress-dependent viscosity
sdepv_expt=1.0,1.0,3.5,3.5,1.0
sdepv_trns=1.0e6,1.0e6,1.0e6,1.0e6,1.0e6
sdepv_bg=1.0e-6,1.0e-6,1.0e-6,1.0e-6,1.0e-6
sdepv_relax_alpha=0.75

The above segment is for stress-dependent viscosity. Parameter *SDEPV* sets on and off stress-dependent viscosity. When *SDEPV=off*, the subsequent 4 parameters will be ignored. Otherwise, parameters *sdepv_expt*, *sdepv_trns*, and *sdepv_bg* specify the exponents *n* of stress-dependent viscosity, transition stress and background stress for each of the five material groups that are defined in the Earth model file *1Dmodel_datafile="model.PREM.vm5a.wrtCrust"*. Note that these two stresses are in SI units or Pa. In this example, n=1 (i.e., Newtonian viscosity) is set for groups 1, 2, and 5 (i.e., the lower mantle and lithosphere as in the Earth model used here), while n=3.5 (i.e., non-Newtonian) is assigned to groups 3 and 4 (i.e., for the upper mantle). It is possible that the material groups can be re-defined. Parameter *sdepv_relax_alpha* is used as a relaxation parameter for stress-dependent viscosity iterations and it should be between 0 and 1 with 1 meaning no relaxation. In this example, *sdepv_relax_alpha=0.75*.

iteration_tolerance=3.e-3   # relative change cutoff for self-grav iteration

Parameter *iteration_tolerance* sets iteration tolerance or accuracy for stress-dependent viscosity iteration (when the stress-dependent viscosity is applied) and for self-gravitation iteration. *iteration_tolerance=3.e-3* means that the iteration ends when relative change of displacement field between two consecutive iterations is less than 0.3%.

By now, we have described the parts of the input file on general information of the modeling in group 1, the loads and loading in group 2, and viscoelastic mechanical structures in group 3. We will now describe coordinates/geometry/grid of the models in group 4.

```
# Group 4: On model domain, numerical grids, parallel processors and dimensional parameters.
nproc_surf=12
nprocx=2
nprocy=2
nprocz=2
```

These four lines set the parallel processors or cores to be used for this case. *nproc_surf=12* indicates 12 spherical caps in the global version CitcomSVE on which the finite element grid is built (Fig. 1). Parameters *nprocx, nprocy, and nprocz* indicate the numbers of processors or cores that each of the cap is further divided and assigned in each of two horizontal directions and radial direction, respectively. In this example case, they are all equal to 2. Therefore, the total number of processors or cores used for this example case run is the product of these four numbers, *nproc_surf**nprocx*nprocy*nprocz* or 96, which should be the same as *-np* in the mpirun command line that we discussed before.

```
nodex=81      # each cap, no of nodes in one horizontal dirc
nodey=81      # each cap, no of nodes in the other horizontal dirc
nodez=49      # each cap, no of nodes in the radial dirc
```

Parameters *nodex, nodey and nodez* specify the number of grid points in each of two horizontal directions and radial direction, respectively for each of the spherical caps. Note that in the way that the grid is designed in CitcomSVE (Fig. 1), *nodex* is always equal to *nodey*, and they are the same for all the 12 caps.

```
mgunitx=5      # multi-grid base level, no. of element
mgunity=5
mgunitz=3
levels=4      # multiplying by 2^(levels-1)
```

These four parameters set up a nested grid for the multi-grid solver of the matrix equation resulting from the finite element discretization of the governing equations. These parameters determine the total number of grid points for the computational domain of each processor. Parameters *mgunitx, mgunity and mgunitz* specify the number of elements in each of two horizontal directions and radial direction, respectively, at the base (i.e., the coarsest grid) level of the nested grid for each domain (or processor). Parameter *levels* indicates the number of levels of the nested grid. Therefore, the number of elements in each direction for each domain (or processor) is, taken x direction as an example, $mgunitx*2^{(levels-1)}$, or 40 in this example case. Therefore, the grid for each domain in this example case is 40*40*24, and this is consistent with the number of processors (e.g., *nprocx=2*) and the number of nodes (e.g., *nodex=81*) in each direction for a spherical cap (note that the number of nodes in each direction is the number of elements plus 1, given that an 8-node brick element is used in CitcomSVE).

 # model domain definition
radius_inner=0.547088
radius_outer=1.0

Parameters *radius_inner* and *radius_outer* are dimensionless radii of the core-mantle boundary and the surface, respectively, both normalized by the radius of the Earth.

 # grid radial layers
r_grid_layers=7
rr=0.547088,0.81635536,0.89483598,0.9356459,0.98430388,0.99058233,1.0
nr=1,17,26,33,45,47,49

These three lines set up radial coordinates of each grid points. Parameter *r_grid_layers* specifies the number of radial grid layers (i.e., *r_grid_layers-1* or 6 in this example case) with each grid layer that can have a different resolution (for the purpose of grid refinement in the radial direction). Parameters *rr* and *nr* determine, starting from the bottom (or inner radius) and up to the surface, radial coordinate and radial nodal index for each grid layer.

 # Dimensional parameters to be used for scalings or normalizations
radius=6371e3
density=4400
shearmodulus=1.4305e11
density_above=0.0
density_below=10895.623
gravacc=9.82
refvisc=1e21

This set of parameters are the reference values all in SI unit that are used to normalize the model parameters. Note that this example case is for incompressible mantle with uniform density.

 # Solver iteration parameters
accuracy=1.0e-3   # dv/v and dp/p error tolerance
tole_compressibility=1e-8   # div(v)/v error tolerance

These two parameters specify solver iteration accuracy or tolerance level (i.e., *dv/v* and *dp/p*, where *v* and *p* are displacement and pressure fields, respectively. Pressure *p* is only relevant for an incompressible medium). Parameter *tole_compressibility* set the tolerance level for incompressibility iteration.

The rest of the parameters in the input file (e.g., inputfile6G_VM5a1) as group 5 rarely need any changes.

4.2. The input files for visco-elastic and density structures of the mantle and crust

This section describes the input files for the visco-elastic structures of the mantle and crust. The background structures are 1-D or depth-dependent and are required for any calculations. Some model calculations may need 3-D viscosity or/and elastic structures. Here we describe how these viscoelastic structures for the mantle and crust can be constructed as input files that can be used for CitcomSVE calculations.

4.2.1. The input file for the background, 1-D visco-elastic structures

Input file *inputfile6G_VM5a1* specifies the file with *1Dmodel_datafile="model.PREM.vm5a.wrtCrust"* that defines the background, 1-D structures. In this example case, the file *model.PREM.vm5a.wrtCrust* is for PREM (i.e., compressible) and VM5a (viscosity) as follow:

```
radius density vp vs grav visc mat_group
100
  3485500    5564    13715     7265  10.70000  3.0950e+21 1
  3519600    5547    13707     7265  10.60000  3.0950e+21 1
  3554200    5529    13699     7265  10.60000  3.0950e+21 1
       …
  5201000    4689    11762     6510   9.95000  3.0950e+21 1
  5201000    4670    11710     6489   9.95000  1.5048e+21 2
       …
  5701000    4381    10754     5948  10.00000  1.5048e+21 2
  5701000    3992    10268     5571  10.00000  4.8530e+20 3
       …
  6360500    3381     8113    4492   9.83000  1.0000e+30 5
  6371000    3381     8113    4492   9.82000  1.0000e+30 5
```

The 1st line indicates that the six columns of this file are (all in SI units) radius, density, Vp, Vs, gravitational acceleration g, viscosity, and material group, respectively. The 2nd line has a single number (100 here) representing the number of lines that describes the structures from the core-mantle boundary at radius of 3485.5 km to the surface at 6371 km (i.e., the final line).

Three remarks on this file:
1) Where two consecutive lines have the same radius indicates a discontinuity in either viscosity or density and elastic properties or both. For example, as above, at radius of 5201000 meters (or 1170 km depth), there is a viscosity jump from 3.0950e+21 Pas to 1.5048e+21 Pas (as in VM5a) and we also define material groups 1 and 2 (the last column) across this radius. At radius 5701000 meters (or 670 km depth), there are changes in density, Vp, Vs and viscosity with material groups from 2 to 3.
2) The 5th column is the gravitational acceleration g. Parameter *1Dmodel_g_from_file* in input file *inputfile6G_VM5a1* determines what g will be used in the modeling calculations. For *1Dmodel_g_from_file=1,* g in this file will be used in the calculations. For *1Dmodel_g_from_file=0,* g will be computed from the density structure (i.e., 2nd column and the core density in *inputfile6G_VM5a1*). The latter is very useful for incompressible models.

3) For modeling calculations with an incompressible mantle, an example model file *model.incomp1* is given as follow (to be used in one of the sample cases later):

radius density vp vs grav visc mat_group
4

3485500	*4604.4*	*1e30*	*5573.877*	*9.80000*	*1.0000e+21*	*1*
6270000	*4604.4*	*1e30*	*5573.877*	*9.80000*	*1.0000e+21*	*1*
6270000	*4604.4*	*1e30*	*5573.877*	*9.80000*	*1.0000e+26*	*2*
6370000	*4604.4*	*1e30*	*5573.877*	*9.80000*	*1.0000e+26*	*2*

Note that the density is a constant of 4604.4 kg/m^3 for incompressible mantle; Vp is set to be an unrealistically large number to be irrelevant; Vs is used to compute the shear modulus which is used in loading calculations; g is set to be 9.8 m/s^2 using *1Dmodel_g_from_file=1*; there is a viscosity discontinuity or jump at 100 km depth where the material groups 1 and 2 are defined.

4.2.2. The input files for 3-D viscosity and elastic structures

For models with 3-D viscosity, users will need to set parameter *3d_visc_from_file* in *inputfile6G_VM5a1* to be 1 and put 3-D viscosity in a file for CitcomSVE to read from. For example,

3d_visc_from_file=1   #0 for 1d layered viscosity; 1 for 3d visc from files
3d_visc_datafile="VE3D_A/ViscA2x2.dat"

where *VE3D_A/ViscA2x2.dat* is the file for 3-D viscosity. In this example 3-D viscosity file, 3-D viscosity is given on a 2^{o} by 2^{o} grid at different depths. Note that the grid for 3-D viscosity structure is completely independent from the 3-D finite element grid in CitcomSVE. CitcomSVE reads in the 3-D viscosity and interpolates it to the finite element grid. Although file *VE3D_A/ViscA2x2.dat* is almost self-explanatory, here are some descriptions of the file.

#The 3D viscosity model has 48 radial layers and is on 2 degree by 2 degree grid.
48 180 90

The first line informs general structure of the viscosity file: 48 radial layers and 2^{o} by 2^{o} grid. The three numbers in the second line (i.e., 48, 180, 90) specify the numbers of radial layers, longitude cells, and co-latitude cells, respectively. In this case, the latter two numbers correspond to 2^{o} by 2^{o} grid.

#The following 48 lines give depth (in km) of viscosity layers.
2.81160e+03
2.70181e+03
2.59202e+03
…
5.45454e+01
3.27272e+01
1.09090e+01

The first line of this segment describes the content of this segment (i.e., depth of each layer). The subsequent 48 lines give the depth of each viscosity layer from the bottom most layer to the topmost layer. The depth interval can be arbitrary and does not need to uniform.

#The following 180 lines give longitudes of viscosity grid.
1.00000e+00
3.00000e+00
5.00000e+00
…
3.57000e+02
3.59000e+02

The first line of this segment describes the content of this segment (i.e., longitudes of the viscosity grid). The subsequent 180 lines define the longitude grid of the viscosity grid, from 1^o to 359^o with increment of 2^o.

#The following 90 lines give latitudes of viscosity grid.
8.90000e+01
8.70000e+01
8.50000e+01
…
-8.70000e+01
-8.90000e+01

The first line of this segment describes the content of this segment (i.e., co-latitudes of the viscosity grid). The subsequent 90 lines define the co-latitude grid of the viscosity grid, from 89^o (near the North Pole) to -89^o (near the South Pole) with increment of 2^o.

#For layer 1 at depth 2.8116e+03 km depth.
8.82480e+19
8.78593e+19
8.77908e+19
…
#For layer 2 at depth 2.7018e+03 km depth.
6.48673e+20
6.45041e+20
6.44448e+20
…
#For layer 48 at depth 1.0909e+01 km depth.
9.42349e+24
9.34908e+24
9.34426e+24
…

Each segment specifies viscosity of a layer on a 2^o by 2^o grid, starting from the bottom most layer (i.e., layer 1) to the topmost layer (i.e., layer 48 in this example case). The viscosity is given in unit of Pas. For each layer, there should be 180x90=16,200 grid points with assigned viscosity. The ordering of the grid points is identical to that of ice model that was discussed in section 4.2.2, i.e., (1º, 89ºN), (3º, 89ºN), …, (359º, 89ºN), (1º, 87ºN), (3º, 87ºN), …, (359º, 87ºS), (1º, 89ºS), (3º, 89ºS), …, (359º, 89ºS).

Similarly, users can also use 3-D elastic structure (e.g., from a seismic tomography model) by setting the following parameters in input file *inputfile6G_VM5a1*:

3d_elastic_from_file=1   #0 for 1d layered ; 1 for 3d elastic from file
3d_elastic_datafile="VE3D_A/elastic2x2.dat"

In bulk of file *elastic2x2.dat,* there are two columns: the bulk and shear moduli (both in SI unit) respectively.

4.3. The input files for temporal and spatial distributions of loads

For any loading calculation, CitcomSVE requires at least one input file that specifies a loading history. For loading with complex temporal and spatial distributions of loads (e.g., glaciation/deglaciation, ICE-6G), additional input files for the spatial distributions of loads at different times are also needed. Here we describe these files' formats so that you can build your own loading history files for your loading models.

4.3.1. Loading history file

Here we first discuss the loading history file, using the same ICE6G/VM5a calculation we discussed in section 4.1 as an example. For this example model run, input file *inputfile6G_VM5a1* specifies the loading history file *ice6g_26ka_stages_timeA.dat* with (see page 9):

load_stages_time_file="ice6g_26ka_stages_timeA.dat"

This model run covers the Earth's last 122 kyrs of glaciation and deglaciation history that is divided into 49 stages. Here are the first few lines and the final two lines in *ice6g_26ka_stages_timeA.dat* that specifies the loading history.

49 1
-122.0 180
-104.0 780
-26.0 10
-25.0 10
…
-0.5 5
0.0 0

In line 1, the first number indicates the number of loading stages (i.e., *49*), and the second number indicates the unit of the time in subsequent lines. If the second number is "*1*" (as in this file), it means that the time is given in terms of 1000 years; if the number is "*0*", it means that the time is given in terms of Maxwell times (i.e., dimensionless).

In line 2, the first number (i.e., *-122.0*) represents the beginning time (i.e., 122 kyrs ago, and the negative sign means the past) of the 1st stage, and the second number (i.e., *180*) specifies the number of time steps to be used to cover this stage.

In line 3, the first number (i.e., *-104.0*) represents the beginning time of the 2nd stage (i.e., 104 kyrs ago) and also the ending time of the 1st stage. This means that the 1st stage lasts for 18 kyrs. And again the second number (i.e., *780*) specifies the number of time steps for the 2nd stage.

This pattern continues to the final stage or 49th stage of this example case that is described by the final two lines in this file. The final stage starts at 0.5 kyrs ago and ends at the present-day (i.e., time 0) with 5 time steps.You may have noticed that for this example case, time increment is 100 years for the entire loading history, and the total number of time steps of the model run is 1220.

As a quick reference, here is a sample loading history file for single harmonic loading calculation with a Heaviside time function (i.e., 1 stage), *single_l_stages_time1.dat.*

```
1 0
-40 200
0.0  0
```

Based on our description above, this file specifies the model calculation with 1 stage loading that starts at 40 Maxwell time ago and ends at time 0 with a total of 200 time-steps.

4.3.2. Load spatial distribution files

Load distribution files are only needed for loading calculations with complex load distributions (i.e., non-single harmonic) in space with *Heaviside=2* as in this example calculation with ICE6G of 49 loading stages. The following three lines in input file *inputfile6G_VM5a1* specify the ice distribution and ocean function files and how the ice model is read:

```
# ice model information, only useful for Heaviside=2
ice_datafile="ICE6G_1x1/ice6g"
ice_sph_or_grid=0                    # 0: regular grid (e.g., 1x1)  1: spherical harmonic Jackson form
ocean_datafile="ICE6G_1x1/ocn6g"
```

Basically, for each stage, one load distribution file is needed. In directory *ICE6G_1x1* which was generated from one of the downloaded compressed files (see section 2), you may find 49 such files that describe ice distribution on a uniform grid (i.e., 1° by 1° in this case): *ice6g.0*, *ice6g.1, ice6g.2*, …, and *ice6g.49*, with the number in the file name indicating the stage number. With parameter *ice_sph_or_grid=0*, CitcomSVE will read the ice distribution on the uniform grid from these files and then interpolate the ice loads to the finite element grid. With parameter *ice_sph_or_grid=1*, CitcomSVE will read the ice distribution as spherical harmonic expansion from similarly names files. We will discuss this ice model input option in the next page after we describe the ice distribution files on a uniform grid.

For every stage with an ice distribution file, there is also an accompanying ocean function file, e.g., *ocn6g.1*, that describes the distribution of oceans and continents. Note that the ocean function distribution is always given on a uniform grid, regardless how *ice_sph_or_grid* is set.

The files for ice distribution on a uniform grid have certain format. For example, here are the beginning and ending lines for near the North and South poles, respectively, and also lines where the latitude changes in *ice6g.1:*

```
360 180
5.0000e-01 8.9500e+01 0.00000e+00
1.5000e+00 8.9500e+01 0.00000e+00
2.5000e+00 8.9500e+01 0.00000e+00
...
3.5850e+02 8.9500e+01 0.00000e+00
3.5950e+02 8.9500e+01 0.00000e+00
5.0000e-01 8.8500e+01 0.00000e+00
1.5000e+00 8.8500e+01 0.00000e+00
...
3.5750e+02 -8.9500e+01 2.85275e+03
3.5850e+02 -8.9500e+01 2.87000e+03
3.5950e+02 -8.9500e+01 2.87000e+03
```

The first line has two numbers indicating the number of grid cells in longitude and latitude directions, respectively (i.e., 360 and 180 for 1^o by 1^o grid). Each of the subsequent lines is for a point on the grid and the 1st, 2nd and 3rd columns are for longitude, latitude and ice height in meters, respectively. It starts from the North Pole region with increasing longitude from 0.5^o to 359.5^o for a fixed latitude of 89.5^o, and then it goes to next latitude of 88.5^o. It continues until reaching to the South Pole region with latitude of -89.5^o.

The ocean function files have the same format as the ice files. The 3rd column is 1 for oceanic region and 0 for land. Given that the ice files and ocean function files are on the same 1^o by 1^o grid, it could have been more economical to only store the 3rd column data in each file.

Now let us describe how to prepare for an ice model in terms of spherical harmonic expansion that can be read in by CitcomSVE with *ice_sph_or_grid=1.* Now the ice model files will take the following prefix *ICE6G_1x1/ice6gsph*, as an example,

```
ice_datafile="ICE6G_1x1/ice6gsph"
ice_sph_or_grid=1                  # 0: regular grid (e.g., 1x1)  1: spherical harmonic Jackson form
```

These files are *ice6gsph.0, ice6gsph.1, ice6gsph.2,* …, and *ice6gsph.49*, for 49 stages. The beginning and ending lines for each of these files (e.g., *ice6gsph.1*) are

100
$0\ 0\ C_{0,0}\ S_{0,0}$
$1\ 0\ C_{1,0}\ S_{1,0}$

1 1 $C_{1,1}$ $S_{1,1}$
2 0 $C_{2,0}$ $S_{2,0}$
2 1 $C_{2,1}$ $S_{2,1}$
2 2 $C_{2,2}$ $S_{2,2}$
...
100 99 $C_{100,99}$ $S_{100,99}$
100 100 $C_{100,100}$ $S_{100,100}$

The first line has one number that is the maximum number of spherical harmonic degree l_{max} (i.e., 100 in this case). The subsequent lines give the degree *l* and order *m* and their corresponding expansion coefficients C_{lm} and S_{lm} for *l* from 0 to 100 and for *m* from 0 to *l* for a given *l*. CitcomSVE computes the ice loads directly at the finite element nodal points using these expansion coefficients. Note that the spherical harmonic expansion here uses the spherical harmonic functions that are normalized to 1. See Appendix A for discussion on the spherical harmonic expansion.

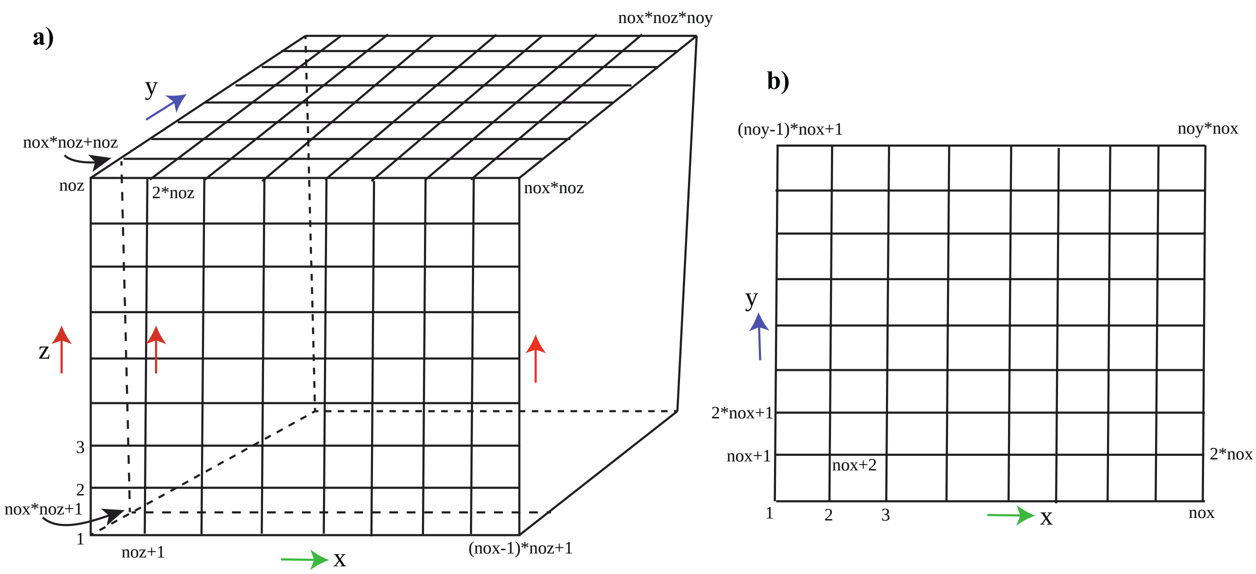

Figure 2. Node numbering for 3-D grid in a computational domain (an arbitrary geometry) (a) and node numbering for its surface grid (b) in CitcomSVE. For the 3-D grid, the numbers of nodes in x, y and z directions are *nox*, *noy* and *noz*, respectively. Therefore, the total number of nodes for this 3-D domain is *nox*noy*noz*, while the number of nodes for the surface is *nox*noy*. In the 3-D domain, the node numbering increases the fastest in *z* or radial direction, the second fastest in *x* or one of the horizontal directions, and the slowest in *y* or the other horizontal direction. For the surface nodes, the node numbering increases the first in *x* direction and then in *y* direction.

5. Output files

Model outputs from CitcomSVE of 3-component displacements, rates of displacements, gravitational potential anomalies and their rates, all on the surface, 2nd-invariant of stress and viscosity throughout the mantle and their horizontal averages are stored in a number of different files. For the example model run with input file *inputfile6G_VM5a1*, these files are in directory

BM_ICE6G, and are *case1.tps_sharm.xx*, *case1.vtps_sharm.x*, *case1.pttl_sharm.x*, *case1.pttldot_sharm.x*, *case1.topo_s.y.x*, *case1.stress_visc.y.x*, and *case1.horiz_ave.z.x*, where *x* is the index for time step (i.e., every 10 time steps for printing surface values and 20 time steps for printing volume values as in *inputfile6G_VM5a1*), *y* is the index for the CPU core (i.e., from 0 to 95, as the model run uses 96 CPU cores), and *z* is the index for the CPU core in vertical direction (i.e., from 0 to 1, as there are two CPU cores in vertical direction with *nprocz=2*). Coordinates of co-latitudes and longitudes for the finite element grid are stored in *case1.coord_s.y*, and radial coordinates can be found in *case1.horiz_ave.z.x*. Note that the displacements on the finite element nodes are relatively small, so we could use the initial coordinates of the nodes as that for all the time in the post-processing (e.g., making plots).

File *case1.log* stores information on convergence, iterations, and CPU times. File case1.time_dep stores global quantities including polar motions.

5.1. Displacements, gravity anomalies, stress and viscosity on finite element grid

Displacements and gravitational potential anomalies are only output for the surface in files *case1.topo_s.y.x* for the CPU cores associated with the surface at a frequency as specified by *storage_spacing_surface=10* (i.e., every 10 time steps in this case) in *inputfile6G_VM5a1*, while 2nd invariant of stress and viscosity are output in files *case1.stress_visc.y.x* at a frequency of every 20 time steps as set by *storage_spacing_vol=20* and *storage_vol=1*. With *storage_vol=2* or *3,* 3-D displacements or stress tensor field will be output for each nodal point. These volume outputs, by producing outputs for all the finite element nodes for each computational domain (or each CPU core) (Fig. 1 and 2), can take up a lot of disk space, and they are produced only if they are really needed.

Files *case1.topo_s.y.x* are only produced for CPU cores or sub-domains that contain the surface. Therefore, for the example model run with a total of 96 CPU cores with 2 in radial direction (i.e., *nprocz=2*), there are 48 files *case1.topo_s.y.x* with CPU core index *y* of 1, 3, …, and 95.

Here are the first several lines of *case1.topo_s.3.20,* one of the 48 files for time step 20.

```
00020 2.00000e+03 1.00000e+02
1.5157e-07 8.6322e-09 5.5421e-08 2.2612e-09 -7.6747e-08 5.0839e-10 -6.7453e-08 -2.7367e-09
1.3524e-07 7.3133e-09 5.3754e-08 2.2063e-09 -8.2838e-08 1.0893e-10 -7.3035e-08 -3.2234e-09
…
```

where line 1 is mostly for information, and each of the subsequent lines is for a node on the surface and there are 41x41=1681 such lines or nodes on the surface, given that *nodex=81, nodey=81*, *nprocx=2 and nprocy=2* in *inputfile6G_VM5a1* for this example case. In line 1, the first three numbers are for time step (i.e., 20), elapsed time in years (i.e., 2,000 years) and incremental time Δt in years (i.e., 100 years), respectively. In each of the subsequent lines, columns 1 to 8 are cumulative (i.e., topography) and incremental displacement in radial direction, total and incremental gravitational potential, cumulative and incremental displacements in co-latitude direction (the South is positive), and cumulative incremental displacements in longitude direction (the East is positive), respectively. Note that all the output quantities are dimensionless (see section 2.5 of Zhong et al., [2022] for normalizations of these variables. To determine the rate of displacement (i.e., velocity), one can simply divide incremental displacement by time increment or Δt. The incremental gravitational potential anomalies can be used to compute rate of gravity change (e.g., for comparing with GRACE data) by dividing time increment or Δt.

Files *case1.stress_visc.y.x* stores 2nd invariant of mantle stress and viscosity for domain or CPU core index *y* at time step *x*, i.e., these files are produced for each of the CPU cores or domains. For the example model run with a total of 96 CPU cores, there are 96 such files for a given time step *x* with *y* of 0, 1, 2, …, 94, and 95.

Here are the first several lines of *case1.stress_visc.3.20,* one of the 96 files for time step 20.

00020 42025 2.00000e+03 1.00000e+02
1.505e+00 3.45835e-07
1.094e+00 2.78889e-07
4.853e-01 1.86236e-07
…
where in line 1, the four numbers are for time step (i.e., 20), the number of grid points or nodes (i.e., 42025) for this domain, elapsed time in years (i.e., 2,000 years) and incremental time Δt in years (i.e., 100 years), respectively; each of the subsequent lines is for mantle viscosity (1st number of the line) and 2nd invariant of stress (2nd number of the line) of the node. For the example case with 41 nodes in each of the two horizontal directions and 25 nodes in radial direction for each domain, there are 41x41x25=42025 nodes in each domain.

The above-mentioned output files require coordinate information for the finite element nodes. The coordinates for surface nodes can be found in files *case1.coord_s.y* (i.e., for the example case *y* can be 1, 3, …, and 95). The ordering of the nodes in these files is identical to that in files *case1.topo_s.y.x*. Note that these coordinate files are produced only once for the first time step to save the disk space.

Here are the first several lines of *case1.coord_s.3,* one of the 48 files (i.e., for the 4th domain).

00001 1.00000e+02
4.8487e-01 6.2832e+00
4.9789e-01 6.2832e+00
5.1090e-01 6.2832e+00
…

where in line 1, the two numbers represent time step (i.e., 1) and elapsed time in years (i.e., 100 years), respectively; each of the subsequent lines is for co-latitude and longitude (both in radium) of a node, respectively. Again, there are 41x41=1681 such nodes or such lines for co-latitude and longitude coordinates in this file.

The radial coordinates of the finite element grid can be found in files *case1.horiz_ave.z.x* for horizontal averages of viscosity and stress, where *z* is the domain or CPU index in radial direction that can be *0, 1,* …, *nprocz-1*, and for our example case with nprocz=2, *z* can be *0,* or *1*. These files are generated for 1st, 10th, …, time steps, for the example case.

Here are the first several lines of *case1.horiz_ave.0.1* (i.e., the bottom domain)

00001 25 1.00000e+02 1.00000e+02
5.47088e-01 3.09500e+00 7.74637e-08
5.63917e-01 3.09500e+00 7.44519e-08
5.80746e-01 3.09499e+00 6.96744e-08
…
8.86116e-01 1.50480e+00 6.20184e-08

where the four numbers in line 1 are for time step (i.e., 1), the number of nodes in radial direction for this domain (i.e., 25), elapsed time in years (i.e., 100 years), and incremental time Δt in years (i.e., 100 years), respectively. Each of the subsequent lines is for a node in radial direction, starting at the core-mantle boundary or dimensionless radius of *0.547088* and ending at the 25th node for radius of *8.86116e-01*, i.e., in the middle of the mantle. The three numbers in each line are the radius, horizontally averaged viscosity and 2nd invariant of stress at the radius, respectively.

File *case1.horiz_ave.1.1* is for the top domain, and has the same format as *case1.horiz_ave.0.1* that we just discussed. Here are the first several lines of *case1.horiz_ave.1.1*.

00001 25 1.00000e+02 1.00000e+02
8.86116e-01 1.50480e+00 6.20184e-08
8.94836e-01 1.09365e+00 6.00619e-08
…
9.95291e-01 1.00000e+05 4.85721e-08
1.00000e+00 1.00000e+05 4.88068e-08

where the 1st line after the information line is for the overlapping node with the bottom domain (i.e., at radius of *8.86116e-01*), and the last line is for the surface where radius is 1.

The radial coordinates from *case1.horiz_ave.0.1* and *case1.horiz_ave.1.1,* together with *case1.coord_s.y* for surface co-latitude and longitude coordinates, can be used to build 3-D finite element grid coordinates that are needed to analyze or plot stress and viscosity with files *case1.stress_visc.y.x*. The nodal ordering in files *case1.stress_visc.y.x* is to increase node numbering along the radial direction from the bottom of the domain to the surface (i.e., 25 nodes in this example run), then shifts horizontally for one node in the same order as in *case1.coord_s.y*, and repeats the increase in node numbering from the bottom of the domain (Figure 2). A given column of nodes in radial direction (i.e., 25 nodes for the domain or 49 nodes in total) have the same co-latitude and longitude coordinates as the corresponding surface node in *case1.coord_s.y*. Likewise, a given layer of nodes in horizontal directions have the same radius as the corresponding node in *case1.horiz_ave.0.1* or *case1.horiz_ave.1.1*.

5.2. Fields with spherical harmonic expansion

Files *case1.tps_sharm.x* and *case1.vtps_sharm.x* are spherical harmonic expansions of cumulative and incremental radial (or vertical) displacements, respectively, at the surface for time step *x*. Files *case1.pttl_sharm.x*, and *case1.pttldot_sharm.x* are for the total and incremental gravitational potential anomalies, respectively, at the surface for time step *x*. These files are generated at a frequency as specified by *storage_spacing_surface=10* in the input file. Again, to determine the rate of displacement (or gravitational potential change), one can simply divide incremental displacement (or incremental gravitational potential) by time increment or Δt. These spherical harmonic expansions are computed from *case1.topo_s.y.x* that we discuss in last section. For definition of the spherical harmonic expansion in CitcomSVE, see equation 30 of Zhong et al., [2022] (or a brief summary in Appendix A of this manual).

Here are the first several lines of *case1.tps_sharm.20.*

```
20 9.02863e+00 tps
 ll   mm     cos      sin
  0    0 9.663792e-13 0.000000e+00
  1    0 9.007467e-07 0.000000e+00
  1    1 1.845651e-07 2.104532e-07
  2    0 -1.173747e-06 0.000000e+00
  2    1 3.988329e-09 -8.460380e-09
  2    2 6.046514e-08 3.599005e-08
...
 32  31 -3.180959e-10 -2.146502e-09
 32  32 -1.182004e-10 -1.196040e-09
```

Lines 1 and 2 are for information, and the 1st and 2nd numbers of line 1 are for the time step (20) and time (in terms of Maxwell time), respectively. The subsequent lines are the spherical harmonic expansion from degree 0 and order 0 to degree 32 and order 32 (as set by *output_ll_max=32* in the input file). Starting from line 3, 1st and 2nd numbers of each line are spherical harmonic degree *l* and order *m*, respectively, and 3rd and 4th numbers of the line are the cosine and sine coefficients of the expansion.

The other three harmonic expansion files *case1.vtps_sharm.x*, *case1.pttl_sharm.x*, and *case1.pttldot_sharm.x* have the same format as that above.

6. Cookbook example cases

Directory *DATA* includes file *README_sample_cases* that describes 14 sample cases. This directory also stores 14 input files for these sample cases including input file *inputfile6G_VM5a1* that was discussed extensively in previous sections. Here we will discuss different sample cases as cookbook example cases.

6.1. Example cases on GIA: 1-D and 3-D viscosity

6.1.1. A case with VM5a (1-D) viscosity

Input file *inputfile6G_VM5a1* was discussed extensively in previous section that uses ice model ICE6G and viscosity model VM5a. From file *README_sample_cases*, this case uses CitcomSVE grid with 12x48x80x80 (i.e., 3,686,400 elements and ~81 km horizontal resolution at the surface) spatial resolution and incremental time of 100 years per time step and is computed on 96 CPU cores. This case should compute for a total of 1,220 time steps for a total of 122,000 years of glaciation and deglaciation history until the present day. Read sections 3, 4 and 5 for details on running this case, input and output files.

Some of key input parameters in input file *inputfile6G_VM5a1:*

```
datadir="BM_ICE6G"
datafile="case1"
apply_SLE=1
polar_wander=1
Heaviside=2    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
load_stages_time_file="ice6g_26ka_stages_timeA.dat"
ice_datafile="ICE6G_1x1/ice6g"
ocean_datafile="ICE6G_1x1/ocn6g"
storage_spacing_surface=10      # write data on the surface every ...
storage_vol=1           #0: no vol. data output, 1: stress/visc, 2: 1+ disp. 3: 1+ stress tensor
storage_spacing_vol=20    #write volume data every .., only relevant for storage_vol!=0
1Dmodel_datafile="model.PREM.vm5a.wrtCrust"  #"earth_model1"
3d_visc_from_file=0      #0 for 1d layered viscosity; 1 for 3d visc from files
```

For details in *inputfile6G_VM5a1*, read section 4.1 on input file. Note that sea level equation solve (*apply_SLE=1*) and polar wander (*polar_wander=1*) effects are all included as for calculations with realistic ice models.

To run this case, go to directory *CitcomSVE-3.0/DATA* with input files including *inputfile6G_VM5a1* and *machinefile0*, and issue the following command to run this case on a PC cluster.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfile6G_VM5a1
```

Depending on your computer's speed, it may take ~2 hours to complete the model run. Output files are case1.* in subdirectory *DATA/BM_ICE6G*. Surface output files (e.g., cumulative and incremental surface displacement and gravity) are generated every 10 time steps, and the volume output files (e.g., stress, viscosity and their horizontal averages) are generated every 20 time steps.

Note that directory *CitcomSVE-3.0/DATA/Compare_Cookbook* was generated from one of the downloaded compressed file *Compare_Cookbbook.tar.gz*. This directory includes two subdirectories *BM_ICE6G* and *BM_LoveNumbers* that stores some selective output files. In subdirectory *BM_ICE6G,* you may find some selective output files case1.*. Note that to reduce file size for *Compare_Cookbbook.tar.gz*, some output files were excluded (e.g., all of those files for stress and viscosity on the 3-D finite element grid and some files for surface displacement and gravity on the finite element grid). Nevertheless, you can compare your output files of results for this example case with those files, if they exist, in *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_ICE6G/case1.**, to verify your calculations.

6.1.2. A case with 3-D viscosity

The second GIA example case uses 3-D mantle viscosity but is otherwise identical to the first example case (e.g., ICE6G). The input file is *inputfile6G_V3D* which differs from input file *inputfile6G_VM5a1* only in two places: datafile="case2" and 3d_visc_from_file=1 in file *inputfile6G_V3D* while file *inputfile6G_VM5a1* has datafile="case1" and 3d_visc_from_file=0. The 3-D viscosity is specified in file *VE3D_A/ViscA2x2.dat* as defined by 3d_visc_datafile="VE3D_A/ViscA2x2.dat" in *inputfile6G_V3D.* See section 4.1 on input file and section 4.2 for details of this 3-D viscosity file. This 3-D viscosity is from A et al., [2013].

Issue the following command in directory *CitcomSVE-3.0/DATA* to run this case on a PC cluster.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfile6G_V3D
```

Output files are case2.* in subdirectory *DATA/BM_ICE6G* (i.e., the same place where case1.* files were stored). Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_ICE6G/case2.** are the results for this sample case, and you can compare your results with these files. Again, not all the results for this cookbook case were included.

6.1.3. Four other cases with VM5a

File *README_sample_cases* lists four more GIA sample cases with their input files *inputfile6G_VM5a2*, *inputfile6G_VM5a3*, *inputfile6G_VM5a4*, and *inputfile6Gc_VM5a* in *CitcomSVE-3.0/DATA.* All these four cases use VM5a viscosity. The first two cases (cases 3 and 4 using *inputfile6G_VM5a2* and *inputfile6G_VM5a3*) are identical to case 1 except for having different spatial and temporal resolutions than case 1 and comparing their input files with *inputfile6G_VM5a1* for case 1 will help see their setups.

Case 5 using *inputfile6G_VM5a4* is for an incompressible mantle with constant mantle density. This case is the same as case GIA_R2 in Zhong et al., [2022]. Here are four key parameters in *inputfile6G_VM5a4* for case 5 that are different from those for the compressible model case 1 (note that there are a few other minor differences including the radii for the CMB and the surface, and the core density).

```
compressible=0
polar_wander_kf=1.11664062
1Dmodel_datafile="model.incomp2"  #"earth_model"
1Dmodel_g_from_file=1
```

Parameter *compressible* is set to be zero for this incompressible model case 5. Parameter *polar_wander_kf* is different from that for case 1 because the fluid Love number depends on the mantle density structure which is different in case 5 from the compressible models. File *model.incomp2* defines the 1-D background viscoelastic and density structures for case 5. The file is given as following:

```
radius density vp vs grav visc mat_group
12
  3485500     4604.4    1e30    5573.877  9.80000  3.0951e+21 1
  5200000     4604.4    1e30    5573.877  9.80000  3.0951e+21 1
  5200000     4604.4    1e30    5573.877  9.80000  1.5048e+21 2
  5700000     4604.4    1e30    5573.877  9.80000  1.5048e+21 2
  5700000     4604.4    1e30    5573.877  9.80000  4.8531e+20 3
  5960000     4604.4    1e30    5573.877  9.80000  4.8531e+20 3
  5960000     4604.4    1e30    5573.877  9.80000  4.8531e+20 4
  6270000     4604.4    1e30    5573.877  9.80000  4.8531e+20 4
  6270000     4604.4    1e30    5573.877  9.80000  1.0000e+22 5
  6310000     4604.4    1e30    5573.877  9.80000  1.0000e+22 5
  6310000     4604.4    1e30    5573.877  9.80000  1.0000e+26 6
  6370000     4604.4    1e30    5573.877  9.80000  1.0000e+26 6
```

Note that for an incompressible mantle, mantle density (2^{nd} column) is a constant and Vp (3^{rd} column) is infinitely large (or set to be 1e30 m/s here). For case 5, Vs (4^{th} column) and g (5^{th} column) are set to be constants here (but they can be set to be depth-dependent). Importantly, with parameter *1Dmodel_g_from_file=1,* g is given from this file (i.e., 9.8 m/s^2), not from calculations using the density. Together with the density, Vs determines the shear modulus which is used for the loading calculations. The 6^{th} column is for viscosity or VM5a, and note that viscosity jumps are set together with the radius (1^{st} column), as explained in section 4.2.1). The last column defines material groups (not used for Newtonian models like discussed here).

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_ICE6G/case5.** are the results for this sample case, and you can compare your results with these files. Again, not all the results for this cookbook case were included.

Case 6 using *inputfile6Gc_VM5a* is the same as case 1 except for using ICE-6Gc with a more complicated glaciation history prior to the last glacial maximum given by Peltier et al., (2015).

Case 6 is the same as case GIA_R2 in Yuan et al., [2024]. Here are three key parameters in *inputfile6G_VM5a5* for case 6 that are different from those in *inputfile6G_VM5a1* for case 1:

load_stages_time_file="ice6g_122ka_stages_timeB.dat"
ice_datafile="ICE6G_1x1_preprocessed/ice6g"
ocean_datafile="ICE6G_1x1_preprocessed/ocn6g"

Basically, the differences are related to ice temporal and spatial distributions. Note that the ice and ocean-continent files in *ICE6G_1x1_preprocessed* are too big in file sizes to be included. You may obtain them following Peltier et al. (2015) or contact us.

6.1.4. Remarks on solution of the sea level equation and treatment of floated ice

Kendall et al. (2005) introduced an iterative scheme to solve the sea level equation to ascertain the model present-day bedrock topography after a GIA model run to be consistent with the observed. In this scheme, the floated ice distribution is determined from the bedrock topography and is excluded from being considered as ice loads to the solid Earth. In this scheme, a complete GIA modeling calculation consists of several repeated or iterative GIA calculations until the model present-day bedrock topography is consistent with the observed. The first iteration of GIA calculation starts with a guessed set of ocean-continent function at different times, and the model outputs of bedrock topography and sea level change are used to modify the ocean-continent functions and floated ice at different times. The iteration continues until the model present-day bedrock topography is in an agreement with the observed. Yuan et al. (2024) implemented Kendall et al.'s scheme as pre- and post-processing processes in the semi-analytical GIA model of Han and Wahr (1995) and A et al. (2013). The pre- and post-processing processes determine ocean-continent functions and floated ice at different times based on the present-day observed bedrock topography and a given ice model (e.g., ICE-6Gc). Yuan et al. (2024) showed that depending on the objective of a given study, 3 iterations are often sufficient, and for some studies no iteration may be needed (Yuan et al., 2024). These pre- and post-processing codes can be used for CitcomSVE-3.0, because they only involve determinations of ocean-continent functions and ice loads (i.e., excluding the floated ice) at different times. For example, for case 6 above, those ocean-continent function and ice load files for ICE6Gc in *ICE6G_1x1_preprocessed* are generated from the pre-processing process.

The pre- and post-processing codes are also included in the CitcomSVE-3.0 package and can be used together with CitcomSVE-3.0.

6.2. Example cases on single harmonic surface loading calculations

File *README_sample_cases* lists five sample cases for computing surface loading problems. The loads are given at a single spherical harmonic with a Heaviside function (i.e., step-function) in time. No ocean (or sea level change) is considered for these calculations. The input files like *inputfileSLl8m4_c* are in *CitcomSVE-3.0/DATA,* where in the file name, letter "*SL*" indicates for surface load Love number calculations, "*l8m4*" stands for spherical degree l=8 and order m=4, and letters "*c*" and "*i*", if present, represent compressible and incompressible mantles, respectively. As indicated in file *README_sample_cases*, these 5 sample cases can be found in Yuan et al., [2024] for compressible cases or Zhong et al., [2022] for incompressible cases.

Here only three sample cases are described, and their input files are *inputfileSLl8m4_c*, *inputfileSLl2m1_c*, and *inputfileSLl2m0_i*.

6.2.1. A surface loading case with input file inputfileSLl8m4_c

This case is for computing the displacements and gravitational potentials at the surface and the 2nd invariant of the stress and dissipation if needed, caused by single harmonic loading at spherical harmonic degree *l=8* and order *m=4*, with a Heaviside function loading history and resolution of 12x64x64x64 using 96 CPU cores. The 1-D mantle viscoelastic structure is given by PREM (i.e., compressible) and VM5a as in model file *model.PREM.vm5a.wrtCrust*. Input file *inputfileSLl8m4* contains all the relevant information for this case and see section 4.1 to understand the details of this input file.

Some key input parameters in *inputfileSLl8m4_c* are:

```
datadir="BM_LoveNumbers"
datafile="caseC"
compressible=1
apply_SLE=0
polar_wander=0
polar_wander_kf=0.952109106687
storage_spacing_surface=5     # write data on the surface every ...
storage_vol=0
Heaviside=1    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
apply_potential=0      #0: apply surface loads, or 1:apply tidal potential
perturbmag=1e-6        # e.g., 1e-6
perturbl=8
perturbm=4
load_stages_time_file="single_l_stages_time1.dat"
1Dmodel_datafile="model.PREM.vm5a.wrtCrust"
```

Loading history file *single_l_stages_time1.dat,* as specified above in *inputfileSLl8m4_c,* indicates that this case computes for 40 Maxwell times in 200 time steps. Note that polar wander, sea level equation, and applied potential are all turn off (i.e., *polar_wander_kf=0.952109106687* is irrelevant here). With *l=8* and *m=4*, we do not expect any polar wander effect anyway. However, it should be pointed out that even if polar wander effect is turn on for this case, the results are the same. Parameter *perturbmag=1e-6* is the dimensionless height or amplitude of the topography of the load at the surface (i.e., 6.371 m if it scaled for the Earth) of the given harmonic *l=8* and *m=4.*

Issue the following command in directory *CitcomSVE-3.0/DATA* to run this case on a PC cluster.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfileSLl8m4_c
```

where file machinefile0 lists 24 compute nodes from node01 to node24. Output files are caseC.* in subdirectory *DATA/BM_LoveNumbers*. As specified in input file inputfileSLl8m4_c, output results are stored every 5 time-steps for surface displacements and gravity and no output for volume data such as mantle stress and viscosity. See section 5 for description of these output files. Love numbers, *h*, *k*, and *l*, as a function of time or time step, are stored in output file *caseC.Love_numbers*, and the first several lines in this file are given as following:

0 0.0000e+00 -1.234779e+00 -5.722059e-04 -7.711194e-02 -1.172631e-02 3.026602e-02
5 1.0000e+00 -2.262729e+00 -6.553741e-04 -1.919690e-01 -4.109571e-03 2.930632e-02
10 2.0000e+00 -3.086887e+00 -7.069213e-04 -2.871667e-01 -2.393847e-03 7.258717e-02
15 3.0000e+00 -3.778451e+00 -7.546375e-04 -3.676183e-01 -1.622684e-03 1.022129e-01
20 4.0000e+00 -4.367761e+00 -7.989682e-04 -4.363441e-01 -1.179299e-03 1.218212e-01
…

Each line is for a time step (i.e., every 5 time-steps). The 1st and 2nd columns are for time step and time (i.e., in terms of Maxwell time), respectively. The 3rd – 7th columns are for *h*, *h*'s dispersion error, *k, k*'s dispersion error, and |*l*|, respectively. See Zhong et al., [2022] for how the Love numbers are defined and computed.

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_LoveNumbers/caseC.** are some of the results for this sample case (again, to save the storage space, only some selective files are given).

<u>*6.2.2. A surface loading case with polar wander effect and input file inputfileSLl2m1_c*</u>

This case is the same as that for the last example case except for spherical harmonic degree *l=2* and order *m=1*, with Heaviside function loading history and resolution of 12x64x64x64 using 96 CPU cores. The 1-D mantle viscoelastic structure is given by PREM (i.e., compressible) and VM5a as in model file *model.PREM.vm5a.wrtCrust*. Given the loading harmonic degree *l=2* and order *m=1*, polar wander is excited, and its effect is calculated. Input file *inputfileSLl2m1_c* contains all the relevant information for this case and see section 4.1 for the details of this input file.

Some key input parameters in *inputfileSLl2m1_c* are:

```
datadir="BM_LoveNumbers"
datafile="caseB"
compressible=1
apply_SLE=0
polar_wander=1
polar_wander_kf=0.952109106687
Heaviside=1    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
apply_potential=0      #0: apply surface loads, or 1:apply tidal potential
perturbmag=1e-6        # e.g., 1e-6
perturbl=2
perturbm=1
```

Note that given the loading harmonic degree *1=2* and order *m=1*, polar wander is generated and *polar_wander=1*.

Issue the following command in directory *CitcomSVE-3.0/DATA* to run this case on a PC cluster.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfileSLl2m1_c
```

Output files are caseB.* in subdirectory *DATA/BM_LoveNumbers*. See section 5 for description of these output files. Love numbers, *h*, *k*, and *l*, as a function of time or time step, are stored in output file *caseB.Love_numbers*.

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_LoveNumbers/caseB.** are the partial results for this sample case, and you can compare your results with these files when they are present.

6.2.3. A surface loading case with incompressible mantle and input file inputfileSLl2m0_i

This case is for loading at degree *l=2* and order *m=0* for incompressible mantle (i.e., “i” at the end of the file name). Otherwise this case is similar to that in section 6.2.1 (i.e., with Heaviside function loading history, resolution of 12x64x64x64 using 96 CPU cores, and no polar wander). The 1-D mantle viscoelastic structure is given in model file *model.incomp1* (this file is described in section 4.2.1 and page 15). Input file *inputfileSLl2m0_i* contains all the relevant information for this case and see section 4.1 for the details of this input file.

Some key input parameters in *inputfileSLl2m0_i* are:

```
datadir="BM_LoveNumbers"
datafile="caseE"
compressible=0
apply_SLE=0
polar_wander=0
Heaviside=1    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
apply_potential=0      #0: apply surface loads, or 1:apply tidal potential
perturbmag=1e-6        # e.g., 1e-6
perturbl=2
perturbm=0
1Dmodel_datafile="model.incomp1"  #"earth_model1"
1Dmodel_g_from_file=1
```

Parameter *compressible=0* indicates an incompressible case. Model file *model.incomp1* indicates an incompressible model with constant density and constant shear modulus and with a 100 km elastic lithosphere over a constant viscosity mantle with 10^{21} Pas. Parameter *1Dmodel_g_from_file=1* indicates that the gravitational acceleration g is read from file *model.incomp1* (i.e., 9.8 m/s^2).

Issue the following command in directory *CitcomSVE-3.0/DATA* to run this case on a PC cluster.

$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfileSLl2m0_i

Output files are caseE.* in subdirectory *DATA/BM_LoveNumbers*. See section 5 for description of these output files. Love numbers, *h*, *k*, and *l*, as a function of time or time step, are stored in output file *caseE.Love_numbers*.

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_LoveNumbers/caseE.** are the partial results for this sample case, and you can compare your results with these files when they are present. Note that this incompressible case is the same as one of the cases in Zhong et al., [2022].

6.3. Example cases on single harmonic tidal loading calculations

File *README_sample_cases* lists 2 sample cases for computing tidal loading problems with input files *inputfileTLHl2m0_c* and *inputfileTLHl2m0_i* in *CitcomSVE-3.0/DATA,* where in the file name, "*TL*" indicates for tidal loading Love number calculations, and "*H*" and "*P*" are for Heaviside and periodic function loading history, respectively.

6.3.1. A Heaviside time function tidal loading case with input file inputfileTLHl2m0_c

This case is for computing the displacements and gravitational potentials at the surface and the 2nd invariant of the stress and dissipation if needed, caused by tidal loading at single harmonic loading at spherical harmonic degree *l=2* and order *m=0*, with a Heaviside function loading history and resolution of 12x64x64x64 using 96 CPU cores. The 1-D mantle viscoelastic structure is given by PREM (i.e., compressible) and VM5a as in model file *model.PREM.vm5a.wrtCrust*. Input file *inputfileTLHl2m0_c* contains all the relevant information for this case and see section 4.1 to understand the details of this input file.

Some key input parameters in *inputfileTLHl2m0_c* are:

```
datadir="BM_LoveNumbers"
datafile="caseF"
compressible=1
apply_SLE=0
polar_wander=0
polar_wander_kf=0.952109106687
storage_spacing_surface=5     # write data on the surface every ...
storage_vol=0
Heaviside=1    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
apply_potential=1      #0: apply surface loads, or 1:apply tidal potential
perturbmag=1e-6        # e.g., 1e-6
perturbl=2
perturbm=0
load_stages_time_file="single_l_stages_time1.dat"
1Dmodel_datafile="model.PREM.vm5a.wrtCrust"
```

Differing from previous cases with surface loading, now *apply_potential=1* means that tidal potential is applied. With loading history file *single_l_stages_time1.dat*, this case computes for 40 Maxwell times in 200 time steps. Note that the non-dimensional applied potential amplitude (1e-6) is normalized by a scale for gravitational potential as discussed in Zhong et al., [2022].

Issue the following command in directory *CitcomSVE-3.0/DATA* to run this case on a PC cluster.

```
$ mpirun -np 96 -nolocal -machinefile machinefile0 ../bin/CitcomSFull inputfileTLHl2m0_c
```

Output files are caseF.* in subdirectory *DATA/BM_LoveNumbers*. As specified in input file inputfileTLHl2m0_c, output results are stored every 5 time steps for surface displacements and gravity and horizontal averages of volume data, but no outputs are generated for the volume data. See section 5 for description of these output files. Love numbers, *h'*, *k'*, and *l'*, as a function of time or time step, are stored in output file *caseF.Love_numbers*.

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_LoveNumbers/caseF.** are subset of the results for this sample case, and you can compare your results with these files.

6.3.2. A Heaviside time function tidal loading case with input file inputfileTLHl2m0_i

This tidal loading case is nearly the same as that in section 6.3.1 (caseF), except for incompressible mantle. The tidal loading is at single harmonic loading at spherical harmonic degree *l=2* and order *m=0*, with a Heaviside function loading history and resolution of 12x64x64x64 using 96 CPU cores. The 1-D mantle viscoelastic structure is given by model file *model.incomp1*.

Some key input parameters in *inputfileTLHl2m0_i* are:

```
datadir="BM_LoveNumbers"
datafile="caseG"
compressible=0
storage_spacing_surface=5     # write data on the surface every ...
storage_vol=0
Heaviside=1    # 1: Heaviside   2: ice model e.g., ice5g    3: Sinusoidal in time
apply_potential=1      #0: apply surface loads, or 1:apply tidal potential
perturbmag=1e-6        # e.g., 1e-6
perturbl=2
perturbm=0
load_stages_time_file="single_l_stages_time2.dat"
1Dmodel_datafile="model.incomp1"
1Dmodel_g_from_file=1
```

Parameter *compressible=0* indicates an incompressible case. Model file *model.incomp1* indicates an incompressible model with constant density and constant shear modulus and with a 100 km elastic lithosphere over a constant viscosity mantle with 10^{21} Pas. Parameter *1Dmodel_g_from_file=1* indicates that the gravitational acceleration g is read from file *model.incomp1* (i.e., 9.8 m/s^2). With loading history file *single_l_stages_time2.dat*, this case computes for 400 Maxwell times in 800 time steps.

Output files are caseG.* in subdirectory *DATA/BM_LoveNumbers*. As specified in input file inputfileTLHl2m0_i, output results are stored every 5 time steps for surface displacements and gravity and horizontal averages of volume data, but no outputs are generated for the volume data. See section 5 for description of these output files. Love numbers, *h'*, *k'*, and *l'*, as a function of time or time step, are stored in output file *caseG.Love_numbers*.

Files *CitcomSVE-3.0/DATA/Compare_Cookbook/BM_LoveNumbers/caseG.** are subset of the results for this sample case, and you can compare your results with these files.

Note that calculations for tidal forcing with periodic time function and mantle as in Fienga et al., [2024] for a compressible mantle are not implemented in CitcomSVE-3.0 at the moment. Such calculations for incompressible mantle are in CitcomSVE-2.1 released at the same github site in October of 2024.

7. References

A, G.R., J. Wahr, and S.J. Zhong (2013). Computations of the viscoelastic response of a 3-D compressible Earth to surface loading: an application to glacial isostatic adjustment in Antartica and Canada, Geophys. J. Int., 192, 557-572.

Bellas, A., and S.J. Zhong (2021). Seismic Strain Rate and Flexure at the Hawaiian Islands Constrain the Fictional Coefficient, Geochemistry, Geophysics, Geosystems, 22, e2020GC009547.https://doi.org/10.1029/2020GC009547

Fienga, A., S.J. Zhong, A. Memin, A. Briaud (2024). Tidal dissipation with 3D finite element deformation code CitcomSVE v2.1: comparisons with the semi-analytical approach, in review, Celestial Mechanics and Dynamical Astronomy.

Kang, K.S., S.J. Zhong, G.R., A, W. Mao (2022), The effects of non-Newtonian rheology in the upper mantle on relative sea level change and geodetic observations induced by glacial isostatic adjusted process, Geophys. J. Int., 228, 1887–1906, https://doi.org/10.1093/gji/ggab428.

Moresi, L.N., and V.S. Solomatov (1995), Numerical investigation of 2D convection with extremely large viscosity variations, Phys. Fluids, 7, 2154-2162.

Paulson, A., S.J. Zhong, and J. Wahr (2005), Modeling post-glacial rebound with lateral viscosity variations, Geophys. J. Int., 163, 357-371.

Qin, C., S.J. Zhong, and J. Wahr (2014). A perturbation method and its application: elastic tidal response of a laterally heterogeneous planet, Geophys. J. Int., 199, 631-647.

Yuan, T., S.J. Zhong, and G.R. A, (2024). CitcomSVE 3.0: A Three-dimensional Finite Element Software Package for Modeling Load-induced Deformation for an Earth with Viscoelastic and Compressible Mantle, EGUsphere, https://doi.org/10.5194/egusphere-2024-3200.

Zhong, S.J., K.X. Kang, G.R. A, and C. Qin (2022). CitcomSVE: A Three-Dimensional Finite Element Software Package for Modeling Planetary Mantle's Viscoelastic Deformation in Response to Surface and Tidal Loads, Geochem. Geophys. Geosyst, 23, e2022GC010359, https://doi.org/10.1029/2022GC010359.

Zhong, S.J., and A.B. Watts (2013). Lithospheric deformation induced by loading of the Hawaiian Islands and its implications for mantle rheology, J. Geophys. Res., 118, 6025-6048, doi:10.1002/2013JB010408.

Zhong, S.J., C. Qin, G.R. A, and J. Wahr (2012). Can tidal tomography be used to unravel the long-wavelength structure of the lunar interior? Geophys. Res. Lett., 39, L15201, doi:10.1029/2012GL052362.

Zhong, S.J., A.K. McNamara, E. Tan, L. Moresi, and M. Gurnis (2008). A benchmark study on mantle convection in a 3-D spherical shell using CitcomS, Geochem. Geophys. Geosyst, 9, Q10017, doi:10.1029/2008GC002048.

Zhong, S.J., A. Paulson, and J. Wahr (2003). Three-dimensional Finite Element Modeling of Earth's Viscoelastic Deformation: Effects of Lateral Variations in Lithospheric Thickness, Geophys. J. Int., 155, 679-695.

Zhong, S.J., Zuber, M.T., Moresi, L., Gurnis, M. (2000). Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection. J. Geophys. Res.-Solid Earth 105, 11063–11082. doi:10.1029/2000JB900003.

Appendix A: Spherical harmonic expansion in CitcomS/CitcomSVE

The spherical harmonic expansion of a function $f(\theta,\varphi)$ is

$$f(\theta,\varphi)=\sum_{l=0}^{l_{max}}\sum_{m=0}^{l}(2-\delta_{m0})[\mathrm{C}_{lm}\cos(m\varphi)-\mathrm{S}_{lm}\sin(m\varphi)]p_{lm}^{J}(\theta), \tag{1}$$

where $p_{lm}^{J}(\theta)$ is related to the associated Legendre function $P_{lm}(\theta)$ as

$$p_{lm}^{J}(\theta)=\sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}}P_{lm}(\theta). \tag{2}$$

Such defined spherical harmonic functions are normalized to 1. Note that this spherical harmonic expansion differs from that in CitcomS/CitcomSVE [see Zhong et al., 2008, 2022]. CitcomS/CitcomSVE uses $p_{lm}^{SVE}(\theta)$ that is defined as

$$p_{lm}^{SVE}(\theta)=\sqrt{\frac{(2l+1)(l-m)!}{2\pi(1+\delta_{m0})(l+m)!}}P_{lm}(\theta). \tag{3}$$

The expansion is

$$f(\theta,\varphi)=\sum_{l=0}^{l_{max}}\sum_{m=0}^{l}[C_{lm}^{SVE}\cos(m\varphi)+S_{lm}^{SVE}\sin(m\varphi)]p_{lm}^{SVE}(\theta). \tag{4}$$

The expansion coefficients can be related to each other as:

$$C_{lm}^{SVE}=\sqrt{\frac{2}{(1+\delta_{m0})}}\mathrm{C}_{lm},\ \ S_{lm}^{SVE}=-\sqrt{\frac{2}{(1+\delta_{m0})}}\mathrm{S}_{lm}. \tag{5}$$