Using SPECFEM3D_Cartesian-3.0 for ASKI - Analysis of Sensitivity and Kernel Inversion, versions 1.0 to 1.2

Florian Schumacher[1]

[1]*Ruhr-Universität Bochum*

Nov 2016

If you use SPECFEM3D_Cartesian for ASKI for your own research, please cite one of our papers [SFL16], or [SF16], as appropriate:

F. Schumacher, W. Friederich and S. Lamara,
"A flexible, extendable, modular and computationally efficient approach to scattering-integral-based seismic full waveform inversion",
Geophysical Journal International, (February, 2016) 204 (2): 1100-1119
http://dx.doi.org/10.1093/gji/ggv505

Schumacher F, Friederich W.
"ASKI: A modular toolbox for scattering-integral-based seismic full waveform inversion and sensitivity analysis utilizing external forward codes".
SoftwareX (2016),
http://dx.doi.org/10.1016/j.softx.2016.10.005

This documentation was written in the hope that it will be useful to the user, but it *cannot be assured* that it is accurate in every respect or complete in any sense. In fact, at some places *this manual is work in progress*.
Please do not hesitate to report any inconsistencies by opening (or adding to) an "issues" topic on https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI or to improve this documentation by incorporating your experiences with SPECFEM3D for ASKI and your personal experience of getting used to it (at best by modifying the source and issuing a pull request on gitHub, in any case let us know about it! Thanks).

Furthermore, I am aware of the poor LaTeXcoding of this document (at the moment, \sloppy is used at the beginning of the document to avoid overfull hboxes in many places). There is a

lot of potential to improve the document style, hence the readability of the manual as a whole, as well as the coding style of the particular `.tex` files. *Please do not hesitate to improve!*

The LaTeXsource files and all related components of this document are available via `https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI` , subdirectory `doc/` of the repository.

Florian Schumacher, Nov 2016

Guide Through This Manual

We assume that you have sufficient knowledge of how to run the regular `SPECFEM3D_Cartesian` software (i.e. without extension for use with `ASKI`).

For details on how to get started by installing everything required for using `SPECFEM3D_Cartesian` with `ASKI` , refer to section 1 (page 6).

Before you start using the code to produce output for `ASKI` , please consider the general comments in section 2 (page 9).

If you are planning to compute a lot of kernels for source-receiver paths (e.g. doing full waveform inversion) it makes sense to use the automated python script `run_specfem3dCartesianForASKI_simulations.py` which conducts a lot of simulations in an automated fashion. Please read section 4 (page 10).

If you want to conduct one single simulation (or just a few ones) producing output for `ASKI` , please read section 3 (page 9).

Section 10 (page 19) is intended to be used as a reference section only.

Bracketed comments starting with "**TODO IN THE FUTURE:**" are intended to mark ideas for future work. So please ignore if you are just applying the code.

Contents

1 Installation and Getting Started

This section explains how to install the SPECFEM3D_Cartesian software (http://geodynamics.org/cig/software/specfem3d) in order to be used as a forward method for ASKI . In general, a regularly installed SPECFEM3D_Cartesian version is extended by certain few modifications so it can produce output for ASKI . So, SPECFEM3D_Cartesian for ASKI basically has the same requirements and dependencies as the SPECFEM3D_Cartesian code, except that it needs a bit more memory and weigh more disc space for output. *Load balancing might not be perfect anymore!!* You should, therefore, have sufficient knowledge of how to run the regular SPECFEM3D_Cartesian software.

1.1 Requirements

1. You require an installation of the ASKI main package, available via https://github.com/seismology-RUB/ASKI :
   `git clone --depth 1 --branch master https://github.com/seismology-RUB/ASKI`
   The directory created by the `git clone` command will be referred to below as ASKI/

2. You need a functioning installation of the SPECFEM3D_Cartesian code, including modifications for usage with ASKI :

   - You can either use the basic extract from the SPECFEM3D_Cartesian master branch (by November 2015) that comes with this package (see section 1.3 (page 7)),
   - or use your running installation of SPECFEM3D_Cartesian and extend it for usage with ASKI , as described below in section 1.4 (page 7).

   In *both* cases you must install this package (section 1.2 (page 6)).

3. You need basic experience in using the regular SPECFEM3D_Cartesian software!

1.2 Installing this package

Clone the latest version of the master branch of the gitHub repository to *the same* directory where you have cloned the ASKI main package to (in the ASKI documentation exemplarily called /your/programs/), i.e. execute
`git clone --depth 1 --branch master https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI`
(in one line, of course) from local path /your/programs/ . This will create subdirectory /your/programs/SPECFEM3D_Cartesian_for_ASKI (also referred to below simply as SPECFEM3D_Cartesian_for_ASKI/) containing the code and documentation of the current release of the extension package SPECFEM3D_Cartesian for ASKI.

Alternatively, go to https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI and download the content of the master branch as a .zip or try executing
`wget https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/archive/master.zip`
(in one line, of course) and extract it in such a way that the code files are contained in /your/programs/SPECFEM3D_Cartesian_for_ASKI/ .

Furthermore you need to compile few more ASKI binaries following these step:

- In SPECFEM3D_Cartesian_for_ASKI/Makefile , set COMPILER appropriately, adjust FFLAGS if required and set the variables BLAS, LAPACK, just as you did in ASKI/Makefile when installing the ASKI main package.
- Execute the command make all from path SPECFEM3D_Cartesian_for_ASKI/

After that, SPECFEM3D_Cartesian_for_ASKI/../ASKI/bin/ should contain the new binaries.

The latest version of the master branch of gitHub repository https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI should be consistent with the latest version of the ASKI repositorie's master branch.

1.3 Using Already Extended Extract of SPECFEM3D_Cartesian-3.0 Code

SPECFEM3D_Cartesian_for_ASKI/specfem3d contains a very basic extract of the git release of SPECFEM3D_Cartesian version 3.0 from git repository https://github.com/geodynamics/specfem3d (master branch) by 2015 November 7. Some folders like utils, doc etc. were removed to keep this copy small. Additionally, two important modifications were applied, which were commited to the devel branch on 3 September 2015, or are about to be commited by the developers team (see comments by "FS FS"):

- in src/specfem3D/setup_sources_receivers.f90 , subroutine setup_sources(), l.180 :
  removing USE_FORCE_POINT_SOURCE .or. from the if-clause, i.e. execute (re)definition of t0 only in case of USE_RICKER_TIME_FUNCTION == .true.
- in src/specfem3D/compute_add_sources_viscoelastic.f90 :
  always call function comp_source_time_function_gauss() with half duration hdur_gaussian(isource) instead of fixed value of 5.d0*DT

Re-configure and compile the software on your system according to the compilers you are using etc., e.g. by executing the following commands from path SPECFEM3D_Cartesian_for_ASKI/specfem3d/ :

```
> ./configure FC=gfortran MPIFC=mpif90
> make default
```

In order to produce ASKI output in SPECFEM3D simulations, copy file SPECFEM3D_Cartesian_for_ASKI/Par_file_ASKI to your respective DATA/ path (which is e.g. specfem3d/EXAMPLES/my_example/DATA/ , or specfem3d/DATA/). This file must be adjusted for any specific simulation (just as all other parameter files) and is described in detail in section 10 (page 19).

1.4 Extend Your Own SPECFEM3D_Cartesian-3.0 Code to Produce Output for ASKI

You can use your own running installation of SPECFEM3D_Cartesian and extend it in the following way for usage with ASKI , provided it has the required functionality. This procedure

was tested for SPECFEM3D_Cartesian git master by 2015 Nov 7, extended by the two modifications described above in section 1.3 (page 7):

1. install SPECFEM3D_Cartesian on your system and make it run, gain experience in using it (below, the installation path is refered to as specfem3d/).
2. Copy file SPECFEM3D_Cartesian_for_ASKI/specfem3D_for_ASKI.f90 to specfem3d/src/specfem3D/
3. Replace file specfem3d/src/generate_databases/model_external_values.f90 by SPECFEM3D_Cartesian_for_ASKI/model_external_values.f90
4. Append content of file SPECFEM3D_Cartesian_for_ASKI/parallel_ASKI.f90 to file specfem3d/src/shared/parallel.f90
5. Append content of file SPECFEM3D_Cartesian_for_ASKI/specfem3D_par_ASKI.f90 to file specfem3d/src/specfem3D/specfem3D_par.f90
6. In specfem3d/src/specfem3D/rules.mk : add the following line into the definition of specfem3D_OBJECTS (e.g. before line with $(EMPTY_MACRO))
   [tab_character]$O/specfem3D_for_ASKI.spec.o \
   (be aware that the above line *must* start with an actual TAB character in order to conform to the GNU-make syntax)
7. In specfem3d/src/specfem3D/prepare_timerun.F90 in subroutine prepare_timerun :
   add the following line at the end of the subroutine, before the statistics output is written to main output file by rank 0:
   call prepare_timerun_ASKI()
8. In specfem3d/src/specfem3D/iterate_time.F90 in subroutine iterate_time :
   add the following line just before the ”enddo” of the time loop
   call write_ASKI_output()
9. In specfem3d/src/specfem3D/finalize_simulation.f90 in subroutine finalize_simulation :
   add the following line just before the main output file is closed at the end of the subroutine
   call save_ASKI_output()
10. Set USE_SOURCES_RECVS_Z = .true. in specfem3d/setup/constants.h (or wherever your file constants.h is located).
11. Recompile all SPECFEM3D binaries, possibly by issuing make in directory specfem3d/
12. In order to produce ASKI output in SPECFEM3D simulations, copy file SPECFEM3D_Cartesian_for_ASKI/Par_file_ASKI to your respective DATA/ path (which is e.g. specfem3d/EXAMPLES/my_example/DATA/ , or specfem3d/DATA/). This file must be adjusted for any specific simulation (just as all other parameter files) and is described in detail in section 10 (page 19).

If you have a newer version of SPECFEM3D_Cartesian which does not work with ASKI as thus described, we are happy to hear about it. Please feel free to get in touch with the ASKI developers (via https://github.com/seismology-RUB or http://www.rub.de/aski).

2 General Things to Consider

- parameters FILE_KERNEL_REFERENCE_MODEL and FILE_WAVEFIELD_POINTS of the parameter file for a specific iteration step must be set to some main ASKI output file, which is the basefile name of ASKI_outfile extendet by .main, see 10.2 (page 20). Use the main ASKI output file of some arbitrary ASKI output, e.g. the kernel displacement output of the first source or some kernel green tensor output.

- As there is a fixed order assumed of the ASKI wavefield points (by procs and local element numbering), the computation of many kernels (e.g. for many source-receiver paths in an inversion) can only be consistent, if the *same* mesh decomposition and the *same* number of procs is used at all times (for those kernels you want to use together, e.g. all kernels in your specific iteration step of an inversion). It may, hence, be sensible to use the same MPI Databases for all SPECFEM3D for ASKI simulations (adjust your script process.sh in such a way, that you do not always recompile and decompose MESH, but only call the solver again, with changed parameter files and source files).

- Green tensor simulations are done using the third coordinate in the station definition as the FORCESOLUTION depth value, in order to allow receivers to be located not only on the surface. It, hence, is strongly recommended to thoroughly use USE_SOURCES_RECVS_Z = .true. in specfem3d/setup/constants.h.

- When using the provided SPECFEM3D version in subdirectory SPECFEM3D_Cartesian_for_ASKI/specfem3d, or using file SPECFEM3D_Cartesian_for_ASKI/model_external_values.f90 for extending your own SPECFEM3D version, the functionality controled by flag COUPLE_WITH_EXTERNAL_CODE does not work anymore, so COUPLE_WITH_EXTERNAL_CODE should be switched to .false. in Par_file. If you want to produce ASKI output with COUPLE_WITH_EXTERNAL_CODE switched on, you need to incorporate the coupling funktionality into model_external_values.f90 accordingly.

- As coordinates of wavefield points (and, hence, inversion grid coordinates) SPECFEM3D for ASKI uses:
  First coordinate = X, second coordinate = Y, third coordinate = Z.

- You must use PRINT_SOURCE_TIME_FUNCTION = .true. in the SPECFEM3D Par_file in order to ensure correct functionality (relevant for cases ASKI_DECONVOLVE_STF = .true. in Par_file_ASKI).

3 One Single Simulation

As usual, you need to do (external or internal) meshing for the appropriate (current) background model. See section 5 (page 12) for details on how to import the current model of an inversion (the inverted model of the last iteration step) into SPECFEM3D.

Set the regular SPECFEM3D files Par_file, CMTSOLUTION / FORCESOLUTION and STATIONS (standard SPECFEM3D functionality; only if you want to record any seismograms).

Additionally, you need to set file Par_file_ASKI to desired values. The file is described in detail in section 10 (page 19).

After that, you are ready to run the code. Since all relevant information for producing ASKI output are read on runtime, you do not need to recompile the SPECFEM3D code every time you run a SPECFEM3D simulation for ASKI , you just need to set the above listet parameter files.

4 Using Automated Python Script for Doing Several Simulations

As usual, you need to do (external or internal) meshing for the appropriate (current) background model. See section 5 (page 12) for details on how to import the current model of an inversion (the inverted model of the last iteration step) into SPECFEM3D.

Python script run_specfem3dCartesianForASKI_simulations.py (provided in directory SPECFEM3D_Cartesian_for_ASKI/) conducts the specified kernel simulations (as described inside the script on the top) by running SPECFEM3D simulations one after another, setting all parameter files before each simulation appropriately. You need to edit the first lines of the script and set all variables defined there to appropriate values, as described in the comments in the script

(**TODO IN THE FUTURE:** maybe it is better to have an input (file?) mechanism to this script. But then: more overhead/extra requirements (packages, arguments handling) to cope with on cluster machines . . .)

The python script may not be suitable for the HPC system you are using. If you are not able to adapt it in a way which makes it possible to be used, you might have to figure out an analogous way yourself how to perform the tasks done by this script.

In case of using the provided python script run_specfem3dCartesianForASKI_simulations.py, some parameters in SPECFEM3D files CMTSOLUTION, FORCESOLUTION, Par_file and in file Par_file_ASKI are automatically changed, while the script conducts the SPECFEM3D simulations one after another.

In the following, only those parameters/lines are listed, which, if necessary, need to be set *manually* before running this python script. All other parameters are set by the script.

4.1 Manually Setting Par_file_ASKI

The following Par_file_ASKI parameters need to be set manually before running the python script, since they are not changed/set by the script.

- USE_ASKI_BACKGROUND_MODEL,FILE_ASKI_BACKGROUND_MODEL
- IMPOSE_ASKI_INVERTED_MODEL,FILE_ASKI_INVERTED_MODEL
- ASKI_INVERTED_MODEL_INTERPOLATION_TYPE,ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS
- ASKI_MAIN_FILE_ONLY (must be set to .false.!)

- OVERWRITE_ASKI_OUTPUT
- ASKI_DECONVOLVE_STF
- ASKI_DFT_double
- ASKI_DFT_apply_taper, ASKI_DFT_taper_percentage
- in case of define_ASKI_output_volume_by_inversion_grid = False in the python script, you need to manually set all parameters concerning the inversion grid, i.e. ASKI_type_inversion_grid, ASKI_(cw)(xyz), ASKI_rot_(XYZ)

It is important to notice, that the flag ASKI_DECONVOLVE_STF is not changed by the automated python script. Although it is required for Green functions ("gt" simulations) to use ASKI_DECONVOLVE_STF = .true. in order to get the displacement field w.r.t. a Dirac impulse source-time function, the situation might arise, that for the forward wavefields ("displ" simulations) emanated by the seismic events you want to use a pre-defined source wavelet (e.g. Ricker wavelet) that *should not* be deconvolved from the wavefields. For now, This can only be realized by doing two separate runs with the python script, one for all "gt" simulations (setting ASKI_DECONVOLVE_STF = .true.) and one for the "displ" simulations (setting ASKI_DECONVOLVE_STF = .false.).

4.2 Manually Setting FORCESOLUTION

The python script *always* automatically sets "latorUTM:", "longorUTM:", "depth:", "factor force source:", "component dir vect source E:", "component dir vect source N:", "component dir vect source Z_UP:". In case of "displ" and "gt" simulations, additionally "f0:" is set to $5 \times DT$. So, if you wish to do a "data" simulation for single force sources, using a different "f0:" value, you should conduct those in a separate run of the python script.

It is strongly recommended (if not necessary) to use USE_SOURCES_RECVS_Z = .true. in specfem3d/setup/constants.h, see comment in section 2 (page 9).

4.3 Manually Setting CMTSOLUTION

The python script *always* automatically sets "latorUTM:", "longorUTM:", "depth:", "Mrr:", "Mtt:", "Mpp:", "Mrt:", "Mrp:", "Mtp:". In case of "displ" simulations, additionally "half duration:" is set to "0.". So, if you wish to do a "data" simulation for moment tensor sources using a different "half duration:" value, you should conduct those in a separate run of the python script.

It is strongly recommended (if not necessary) to use USE_SOURCES_RECVS_Z = .true. in specfem3d/setup/constants.h, see comment in section 2 (page 9).

4.4 Manually Setting STATIONS

In the upper part of the python script, the flag create_specfem_stations can be set to True. In this case, the SPECFEM3D STATIONS file is automatically generated from the ASKI file FILE_STATION_LIST.

If you do not use this flag to automatically generate the SPECFEM3D STATIONS file, you must provide it manually. The standard SPECFEM3D STATIONS file should contain the definition of stations as in the ASKI file FILE_STATION_LIST, in consistend SPECFEM3D notation, i.e. coordinate columns being lat (= Y, third column of STATIONS and fourth column of FILE_STATION_LIST) and lon (= X, fourth column of STATIONS and third column of FILE_STATION_LIST) and elev (= Z, sixth column of STATIONS and fifth column of FILE_STATION_LIST).

You must also assure to use the very same station names and network codes in file STATIONS as in ASKI file FILE_STATION_LIST!

It is strongly recommended (if not necessary) to use USE_SOURCES_RECVS_Z = .true. in specfem3d/setup/constants.h, see comment in section 2 (page 9).

5 Importing external models into SPECFEM3D , e.g. simple background model or currently inverted model for next iteration step

There are two types of external models that can be put (in combination) into SPECFEM3D, using a special implementation of the SPECFEM3D module model_external_values:

Simple 1D layered gradient background models can overwrite the default background model (coming from the mesher, e.g. Trelis). Exported .kim files (as produced by ASKI program exportKim with option -otxt) may be superimposed onto the background model (default Trelis model or ASKI 1D background model) and used as a model for the new iteration of full waveform inversion of ASKI .

These two types of external modesl are explained in the following. To be able to use *any* of the two (or both in combination), you *must* set MODEL = external in Par_file !

5.1 Overwrite background model by simple 1D layered gradient model

The logical flag USE_ASKI_BACKGROUND_MODEL in Par_file_ASKI indicates whether SPECFEM3D_Cartesian should use the 1D reference model as defined in the text file with name given by FILE_ASKI_BACKGROUND_MODEL , relative to DATA/ . This mode will overwrite model values on all GLL points, dependent on depth (or rather Z). *A model like this will not affect the meshing of spectral elements or any internal boundaries created by the meshing process!*

The 1D model is defined by a list of model values at given depth nodes between which a spline interpolation is done. A template of such a background model file, containing documenting commentary, is given by file SPECFEM3D_Cartesian_for_ASKI/ASKI_background_model_template. The specific format of this text file is described now in the following:

the first line is ignored, this line may contain a short description of the model or can be empty.

the second line contains one real value defining the maximum Z value z_{max} of the model domain (since internally, Z values are processed and there is no knowledge about "depth", hence the Z-value of the following depth nodes are computed as $z_{\text{node}} = z_{\text{max}} - \text{depth}_{\text{node}}$).

the third and fourth line simply define the characteristics of the depth nodes which are defined in the table (for convenience when reading the file by the program):
The third line must contain the number of layers, between which discontinuities are allowed in the 1D model. At the boundary of any two layers, there should be a "double node", i.e. two lines with *same* depth. There is no spline interpolation done accross any layer boundaries, i.e. over any double node. Different model values on either side of a double node will be interpreted as a discontinuity in the model. You can also set the same model value on either side of a double node, e.g. if you want to have a half space of the same model values as a gradient coming from above, etc.
The fourth line contains as many integer values (separated by white space) as there are layers (as defined by line three) and gives for each layer the number of nodes. amust contain the number of nodes inside each layer.

starting from line 4, each line defines a depth node giving (isotropic) model values at this depth. The columns are separated by white space and assume the meaning:
depth [m] density [Kg/m^3] vp [m/s] vs [m/s] Qmu Qkappa
The depth is assumed to be monotonically *increasing*, the first line should have depth 0.

Everything below the expected number of lines is ignored, so you can also add commentary below the model definition.

5.2 Impose exported `.kim` model onto background model

This functionality is controled by logical flag `IMPOSE_ASKI_INVERTED_MODEL` along with the parameters `FILE_ASKI_INVERTED_MODEL`, `ASKI_INVERTED_MODEL_INTERPOLATION_TYPE`, `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS` in `Par_file_ASKI`.

`FILE_ASKI_INVERTED_MODEL` provides the filename (relative to directory `DATA/`) of the exported `.kim` file (text file as produced using option `-otxt` of `ASKI` executable `exportKim`).

`ASKI_INVERTED_MODEL_INTERPOLATION_TYPE` and `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS` control the method of interpolating the given inverted model (defined on an `ASKI` internal inversion grid) onto the GLL points used in your `SPECFEM3D` simulation. At the moment, an unstructured 3D interpolation after Shepard [She68] is supported which is founded on inverse-distance weighting and accounts for issues of nearby points, direction and slope. `ASKI_INVERTED_MODEL_INTERPOLATION_TYPE` can be either set to `shepard_standard` or to `shepard_factor_radius` .

In case of type `shepard_factor_radius` , the factor given by `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS` controls the influence of neighbouring control nodes on the interpolation (larger factor will include more control nodes (further away) for the interpolation). For a particular GLL point, first the closest control node of the inverted model (center of inversion grid cell) is found. Then this distance is multiplied by `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS` to yield a radius within which all contained control nodes of the inverted model will be taken into account to compute the interpolated value for that GLL point.

Method `shepard_standard` is the same as using `shepard_factor_radius` with `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS = 2.0`. This factor proved to be a good choice. When setting the method to `shepard_standard`, any value given for `ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS` is ignored.

6 Preparing Synthetic Data as Expected by ASKI

Use executable transformSpecfem3dCartesianSyntheticData, as described below.

It is assumed that a copy of the content of the OUTPUT_FILES directory (without the MPI_DATABASES files etc...) of all involved SPECFEM3D simulations (which contain the standard seismograms files) can be found at the path as choosen by the automated python script (see 4 (page 10)), i.e. filename of the kernel displacement file for the respective event with the extension _OUTPUT_FILES. The synthetic data then is written in the required form to path PATH_SYNTHETIC_DATA/, where the filenames are by convention synthetics_EVENTID_STATIONNAME_COMPONENT. Make sure that the ASKI frequency discretization as defined by the ASKI main parfile and iter parfile is correctly set! Also, all other objects used for an ASKI iteration step (like wavefield points file, inversion grid etc.) must be in place, since for executing transformSpecfem3dCartesianSyntheticData the basic requirements for an iteration step are initiated (compare ASKI manual, section "Initiate Basic Requirements").

Executing transformSpecfem3dCartesianSyntheticData (without arguments) will print a help message how to use it and will list the required positional arguments and mandatory options and optional options. Those are described in more detail in the following:

Positional arguments

main_parfile Main parameter file of inversion.

Mandatory options

-bicode band_instrument_code band_instrument_code is a character string consisting of bandcode and instrument code, i.e. the first two characters before the component in seismogram filename, e.g. "LH" if your filenames look like "network.staname.LH?.semd"

-ori orientation orientation is either "NEZ" or "XYZ", indicating the component characters following band_instrument_code.

-dt time_step Gives the time step of the SPECFEM3D seismograms (as in the SPECFEM3D Par_file).

-nstep number_of_time_Steps Defines the number of samples NSTEP as in the SPECFEM3D Par_file

-ocomp "comp_1 ... comp_n" Vector of station components for which synthetic data is produced. Valid components are CX , CY , CZ , N , S , E , W , UP , DOWN (also see ASKI user manual, section on "Data in ASKI ").

Optional options

-evid event_id `event_id` indicates a single event for which synthetic data is produced, otherwise synthetic data is produced for *all* events (as defined by the `ASKI FILE_EVENT_LIST`).

-dconv If set, the source time function will be deconvolved from the `SPECFEM3D` seismograms. This option is consistend with setting `ASKI_DECONVOLVE_STF = .true.` in `Par_file_ASKI`.

-bin Indicates whether the `SPECFEM3D` trace files are binary files or not. For ascii output simply do not set option -bin .

-ext extension *This option is not needed for standard functionality, only use this if you know what your're doing*. Standard functionality (i.e. *not* setting `-ext`) will produce *displacement* spectra. If `-ext` is set, however, the specific file extension `extension` is forced to be used. `extension` represents *anything* following the orientation character, including *all* dots etc., e.g. ".semv" if the filenames that should be used look like "network.staname.FX?.semv".

-diffts *This option is not needed for standard functionality, only use this if you know what your're doing*. Standard functionality (i.e. *not* setting `-diffts`) will produce *displacement* spectra. If set, the time series will be differentiated (by simple first order central differences) before further processing.

7 Preparing synthetically computed "measured" data as expected by **ASKI**

For synthetic tests, you might want to treat synthetic data computed by SPECFEM w.r.t. some perturbed earth model as (noise-free) measured data. `SPECFEM3D for ASKI` "data" simulations (e.g. produced by automated python script, 4 (page 10)) will produce these data in standard SPECFEM time-domain output formats. In order to transform these time series to frequency-domain measured data files in the form required by `ASKI` there are two possibilities:

First of all, the `ASKI` executable `transformMeasuredData` can be utilized for this purpose (e.g. choosing seismic Unix output for the SPECFEM simulations, or renaming the text output trace files as required for input of `transformMeasuredData` option `-txt`).

As an alternative (providing more possibilities of data processing like scaling, filtering, differentiating), you can use executable `transformSpecfem3dCartesianMeasuredData`, as described below. It is assumed that a copy of the content of the `OUTPUT_FILES` folder (without the `MPI_DATABASES` files etc...) of the "data" simulations (which contain the standard seismograms files) can be found in respective directory `PATH_MEASURED_DATA/data_EVENTID_OUTPUT_FILES`. The measured data files then are written in the required form to path `PATH_MEASURED_DATA/`, where the filenames are by convention `data_EVENTID_STATIONNAME_COMP`. Make sure that the frequency discretization of `ASKI` measured data as defined by the `ASKI` main parfile is correctly set, as well as the measured data path!

Executing `transformSpecfem3dCartesianMeasuredData` (without arguments) will print a help message how to use it and will list the required positional arguments and mandatory options and optional options (described in more detail in the following). Note that only necessary things will be read/initialized on the basis of the `ASKI` main parameter file given (e.g. everything not needed, like integration weights, inversion grid, will *not* be initialized at this point, i.e. those quantities are not required to exist already! Hence, you may use this executable before starting to solve the forward problem in the first iteration of `ASKI` FWI).

Positional arguments

`main_parfile` Main parameter file of inversion.

Mandatory options

`-bicode band_instrument_code` `band_instrument_code` is a character string consisting of bandcode and instrument code, i.e. the first two characters before the component in seismogram filename, e.g. "LH" if your filenames look like "network.staname.LH?.semd"

`-ori orientation` `orientation` is either "NEZ" or "XYZ", indicating the component characters following `band_instrument_code`.

`-dt time_step` Gives the time step of the `SPECFEM3D` seismograms (as in the `SPECFEM3D Par_file`).

`-nstep number_of_time_Steps` Defines the number of samples `NSTEP` as in the `SPECFEM3D Par_file`

`-ocomp "comp_1 ... comp_n"` Vector of station components for which measured data is produced. Valid components are `CX, CY, CZ, N, S, E, W, UP, DOWN` (also see `ASKI` user manual, section on "Data in `ASKI` ").

Optional options

`-ext extension` *This option is not needed for standard functionality, only use this if you know what your're doing*. Standard functionality (i.e. *not* setting `-ext`) will produce *displacement* spectra. If `-ext` is set, however, the specific file extension `extension` is forced to be used. `extension` represents *anything* following the orientation character, including *all* dots etc., e.g. ".semv" if the filenames that should be used look like "network.staname.FX?.semv".

`-filter` If set, the respective event filters and station (component) filters as defined by the `ASKI` main parfile will be applied to the spectra before writing them to file. I.e. if in the `ASKI` main parfile any filtering is switched off (by respective flags), *no* filtering will by applied by executable `transformSpecfem3dCartesianMeasuredData` ! If in `ASKI` main parfile, only event filters are enabled, then this option `-filter` will cause the executable only to apply the event filters etc.

-evid event_id event_id indicates a single event for which measured data is produced, otherwise measured data is produced for *all* events (as defined by the ASKI FILE_EVENT_LIST).

-cfreq forward_method Instead of the ASKI standard definition of real-valued frequencies by $f_k = k \cdot \Delta f$ (for frequency indices k and frequency step Δf), the data is produced w.r.t. complex frequencies consistent with the given forward method forward_method. For instance, setting -cfreq GEMINI will produce Gemini-consistent spectral data at complex frequencies with additional constant imaginary part $\sigma = -5\Delta f/2\pi$, thus implicitely using the actual frequencies $f_k = k \cdot \Delta f + i \cdot \sigma$ (with i being the imaginary unit, also see ASKI user manual, section on "Data in ASKI "). Any filter values (in case of -filter is set) are assumed to be given at those complex frequencies, too!

-dconv If set, the source time function will be deconvolved from the SPECFEM3D seismograms. This option is consistend with setting ASKI_DECONVOLVE_STF = .true. in Par_file_ASKI.

-diffts *This option is not needed for standard functionality, only use this if you know what your're doing*. If set, the time series will be differentiated (by simple first order central differences) before further processing.

-scale ts_scale_factor If set, the time series are scaled with factor ts_scale_factor (must be different from 0) after reading in, before transforming to frequency domain.

-bin Indicates whether the SPECFEM3D trace files are binary files or not. For ascii output simply do not set option -bin .

8 Create ASKI filter file from source time function file

Using the executable createSpectralFilters contained in the ASKI main package, spectral filter files as required by ASKI can be produced from a time-domain trace file (e.g. the output txt file plot_source_time_function.txt). This functionality may be useful for synthetic tests: After producing "measured" data by simulation with a specific source-time function (e.g. a Ricker wavelet), spectral filters containing the source-time-function information should be used in ASKI .

By option -eventf you must provide a parameter file, particularly specifying the wavelet by STF_FILE, STF_COLUMN_OF_TRACE, STF_DT, STF_NSTEP . Please refer to the ASKI user manual for any details.

9 Experimental feature: Convolving impulsive synthetics with given source-time function

Attention: this is an experimental feature, the executable convolveWithStf *does not yet produce satisfying results, i.e. it must be debugged!!*

The synthetic data computed for an `ASKI` iteration step is computed w.r.t. an inpulsive source-time function and the actual source-time function of the measured data is modelled by applying spectral filters to the frequency-domain synthetic data. This, however, does not allow to compare the current data fit in that iteration visually by looking at time series (comparing measured and synthetic data).

The executable `convolveWithStf` (*not yet debugged!!, do not believe its output!*) aims at convolving the synthetic seismograms (produced by SPECFEM for an `ASKI` iteration) with a given source time function (e.g. the one which is assumed for the measured data, or the one that was used in synthetic test inversions for computing the measured data). Optionally, the original source time function that was used in the SPECFEM simulation for synthetic data (i.e. the "kernel displacement" simulations), which is a thin Gaussian in case of a single force source or a steep error function in case of a moment tensor source, may be devonvolved from the synthetics before convolving with the given source time function.

The idea of this executable is, to enable the user to have a look at the time-domain data fit *without* having to run another forward simulation for each source using the given source-time function (that was used for producing measured data).

However: it is not clear to the author (Florian Schumacher, August 2016) whether such a process can be successful at all when the synthetic data was modelled to be stable only up to a quite low maximum frequency (e.g. in the first iteration steps), since the higher frequencies that are contained in the measured data are *not* stably contained in the syntheticy!

(**TODO IN THE FUTURE:** this executable could support to simply apply the `ASKI` spectral filters *but* these are usually given only in a small frequency band and are given in a frequency discretization different from that used for Fast Fourier Transform.)

In the following, there is a short description of the required positional arguments and mandatory options and optional options of executable `convolveWithStf`.

Positional arguments

`stf_file` File containing the source time function by which the synthetics are to be convolved. It is assumed that it is a text file containing *two* columns: time (first column) and source-time function values (second column). The first column is used to get the time step of the given source time function (by subtracting first from last time value and dividing by the number of time intervals).

`main_parfile` Main parameter file of inversion.

Mandatory options

`-bicode band_instrument_code` `band_instrument_code` is a character string consisting of bandcode and instrument code, i.e. the first two characters before the component in seismogram filenames, e.g. "LH" if your filenames look like "network.staname.LH?.semd"

`-ori orientation` `orientation` is either "NEZ" or "XYZ", indicating the component characters following `band_instrument_code`.

-dt time_step Gives the time step of the SPECFEM3D seismograms that was used in the current ASKI iteration (as in the SPECFEM3D Par_file).

-nstep number_of_time_Steps Defines the number of samples NSTEP of the SPECFEM3D seismograms that was used in the current ASKI iteration (as in the SPECFEM3D Par_file).

Optional options

-ext extension *This option is not needed for standard functionality, only use this if you know what your're doing*. Standard functionality (i.e. *not* setting -ext) will produce *displacement* spectra. If -ext is set, however, the specific file extension extension is forced to be used. extension represents *anything* following the orientation character, including *all* dots etc., e.g. ".semv" if the filenames that should be used look like "network.staname.FX?.semv".

-evid event_id event_id indicates a single event for which the convolution should be done, otherwise it will be done for *all* events (as defined by the ASKI FILE_EVENT_LIST).

-bin Indicates whether the SPECFEM3D trace files are binary files or not. For ascii output simply do not set option -bin .

-dconv If set, the source time function will be deconvolved from the SPECFEM3D seismograms. This option is consistend with setting ASKI_DECONVOLVE_STF = .true. in Par_file_ASKI.

-opath output_path If set, the character string output_path will be used to write the convolved seismogram files. It is assumed relative to the respective specfem seismograms path, i.e. path PATH_KERNEL_DISPLACEMENTS/kernel_displ_eventID_OUTPUT_FILES/ . If not set, the default value "convolved/" is used (indicating the subdirectory "convolve/").

10 File Par_file_ASKI

File Par_file_ASKI is, just like the file Par_file, located in directory DATA/ of your current SPECFEM3D example. It basically controls ASKI functionality SPECFEM3D if used along with an ASKI extended SPECFEM3D installation. If in such an installation file Par_file_ASKI is not present, no ASKI output is produced and SPECFEM3D runs with standard functionality.

In the following, we give a short description of the functionality of parameters defined in file Par_file_ASKI.

10.1 ASKI external model

The following parameters will *only* have any effect, when you set MODEL = external in Par_file.

First the SPECFEM3D model is set, as defined by standard SPECFEM mechanisms. Then, *only if indicated* by flag USE_ASKI_BACKGROUND_MODEL, this model is overwritten by the ASKI 1D background model at all depths where this background model is defined (see 5.1 (page 12)). After that, *only if indicated* by flag IMPOSE_ASKI_INVERTED_MODEL an ASKI inverted model is superimposed to the then existing model values (will set absolute model values, but at the boundaries of the inversion domain it will smooth out to the existing model, see 5.2 (page 13)).

USE_ASKI_BACKGROUND_MODEL, FILE_ASKI_BACKGROUND_MODEL

Logical flag USE_ASKI_BACKGROUND_MODEL indicates whether at all to use a 1D background model and

FILE_ASKI_BACKGROUND_MODEL , defines a filename relative to DATA/ from which the 1D model is read. For the required format of this text file, see 5.1 (page 12).

IMPOSE_ASKI_INVERTED_MODEL, FILE_ASKI_INVERTED_MODEL, ASKI_INVERTED_MODEL_INTERPOLATION_TYPE, ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS

Logical flag IMPOSE_ASKI_INVERTED_MODEL indicates whether at all to impose an ASKI inverted model onto the existing model (standard background or standard background plus ASKI 1D background).

FILE_ASKI_INVERTED_MODEL gives the filename relative to DATA/ where to find the file containing the the exported .kim file (text file as produced using option -otxt of ASKI executable exportKim).

Parameters ASKI_INVERTED_MODEL_INTERPOLATION_TYPE, ASKI_INVERTED_MODEL_FACTOR_SHEPARD_RADIUS control the way of interpolating the model values given on control nodes of an ASKI inversion grid onto the GLL points of the current SPECFEM3D mesh. For their meaning see 5.2 (page 13).

10.2 ASKI output

COMPUTE_ASKI_OUTPUT, ASKI_MAIN_FILE_ONLY, OVERWRITE_ASKI_OUTPUT

Parameter COMPUTE_ASKI_OUTPUT controls whether at all ASKI output is produced by the SPECFEM3D solver (i.e. kernel green tensor kernel displacement main or frequency files).

If COMPUTE_ASKI_OUTPUT = .true., then logical flag ASKI_MAIN_FILE_ONLY controls whether to produce only the .main output file at the beginning of a simulation and immendiately terminate. No frequency output files and no SPECFEM seismograms will be produces in this case. This functionality is useful, if you want to check the resolution of wavefield points with regard of your chosen inversion grid or you want to look at the kernel reference

model (background model used by SPECFEM) *before* running all your simulations for an iteration step of ASKI waveform inversion. With one single .main output file available, namely, you can execute the ASKI executable initBasics and check for everything related to your wavefield points and inversion grid.

Logical flag OVERWRITE_ASKI_OUTPUT controls if the ASKI output files files shall be overwritten if existend or not. If set to .false. and any of those files exist, the SPECFEM3D solver will terminate raising an error message.

Setting COMPUTE_ASKI_OUTPUT = .false. will *not* prevent the SPECFEM3D mesher from setting an ASKI external model, if MODEL = external in Par_file and any external model is defined as described above in section 10.1 (page 20) ! So you can use an ASKI external model along with a standard SPECFEM3D simulation.

ASKI_outfile, ASKI_output_ID

ASKI_outfile defines the absolute base file name of ASKI output files. The actual output files of this simulation will be this base name appended by file extensions .main (for main output file) and .jf###### for each frequency (e.g. .jf000013 for frequency index 13).

ASKI_output_ID is a character string of maximum lenght as defined by parameter length_ASKI_output_ID in file SPECFEM3D_Cartesian_for_ASKI/specfem3D_par_ASKI.f90 with which all output files of the current simulation will be taged, and it will be used to check consistency of the files (could be a timestamp, eventID, station name, component etc).

ASKI_DECONVOLVE_STF

Logical flag ASKI_DECONVOLVE_STF indicates whether to deconvolve (the derivative of) the source time function from the wavefield spectra before writing them to files. Select .true. for any Green function computations! Even if a Heaviside source time function is used, the velocity field is not exactly a Green function (i.e. displacement wavefield w.r.t. an impulse source time function), since a steep error function is used by SPECFEM to resemble a quasi-Heaviside function. This steep error function, furthermore, is dependent on timestep DT! Hence, only by deconvolution of (the derivative of) this quasi-Heaviside source time function, the real Green function (generated by an impulsive Dirac source time function), which is independent of the time step can be computed.

Dependent on the type of source mechanism (single force, or moment tensor), SPECFEM uses a Gaussian (single force) or a Heaviside (moment tensor) in case of USE_RICKER_TIME_FUNCTION = .false.. ASKI takes care about, which wavefield to store and which source wavelet to deconvolve. ASKI always produces displacement spectra w.r.t. a Dirac impulse time function (if ASKI_DECONVOLVE_STF = .true.).

10.3 Frequency discretization and Fourier transform

The double precision df [Hz] and integer values jf have the following meaning: The spectra are saved for all frequencies f = (jf)*df [Hz].

ASKI_df, ASKI_nf, ASKI_jf

ASKI_df is a predefined frequency step that is used to evaluate the spectrum. In case we want to do an inverse FT in case of time-domain sensitivity kernel computation, we need to choose ASKI_df with care as ASKI_df = 1/length_of_time_series and suitably high frequency indices (dependent on frequency content). Otherwise we could lose periodicity (if in exp(-i2pi(k)(n)/N) N is no integer, these are no roots of 1 anymore). The spectra are saved for frequencies f = (ASKI_jf)*ASKI_df (ASKI_nf many).

ASKI_DFT_method

ASKI_DFT_method defines which numerical method is used to do the on-the-fly Fourier transform of wavefield and strain components (at the wavefield points) in order to produce spectral output for ASKI . At the moment, two values are supported:

- ASKI_DFT_method = EXPLICIT_SUMMATION
  on-the-fly summation of complex values $s(t)\, e^{-i\,2\pi\, f\, t}$ (where $s(t)$ represents displacement or strain time series); slightly more memory efficient than GOERTZEL_STANDARD

- ASKI_DFT_method = GOERTZEL_STANDARD
  using Goertzel's algorithm (as in [Goe58], only adapted for time-forward time series, also compare ASKI developer's manual, section on adding support for new forward codes, subsection on choosing a method of discrete Fourier transform); compared with EXPLICIT_SUMMATION requires only half the number of multiplications for Fourier transform operations. *However*, the overall performance improvement is not noticeable for small tested examples, as these operations comprise only a part of the additional operations done when producing output for ASKI and other things like array/memory access also play an important role.

ASKI_DFT_double

Choose precision of Discrete Fourier Transform. If there is enough memory available, it is highly recommended to use ASKI_DFT_double = .true. in which case double complex spectra are hold in memory (single precision is written to file, though, but less roundoffs during transformation). Otherwise choose ASKI_DFT_double = .false. in which case single precision spectra will be used in memory. The transformation coefficients exp(-i*2pi*f*t) are always in double complex precision!

ASKI_DFT_apply_taper, ASKI_DFT_taper_percentage

Decide whether the (oversampled, noisy, ...) time series should be tapered by a hanning taper (on tail) while applying the discrete fourier transform (on-the-fly). If ASKI_DFT_apply_taper = .true., the value of ASKI_DFT_taper_percentage (between 0.0 and 1.0) defines the amount of total time for which the hanning taper will be applied at the tail of the time series.

10.4 Inversion grid

ASKI_type_inversion_grid

ASKI supports several types of inversion grids for FORWARD_METHOD = SPECFEM3D. ASKI_type_inversion_grid =

1. (TYPE_INVERSION_GRID = schunkInversionGrid)
   NOT TO BE USED WITH SPECFEM3D Cartesian!
   ASKI internal, but SPECFEM independent simple spherical inverison grid

2. (TYPE_INVERSION_GRID = scartInversionGrid)
   ASKI internal, but SPECFEM independent cartesian inversion grid:
   The values for ASKI output are stored at all inner GLL points of spectral elements which lie inside the block volume defined below by parameters ASKI_(cw)(xyz). ASKI loactes the coordinates of those points inside the inversion grid cells and computes integration weights for them.

3. (TYPE_INVERSION_GRID = ecartInversionGrid)
   External inversion grid provided e.g. by Trelis, which may contain tetrahedra, as well as hexahedra. As in case of ASKI_type_inversion_grid = 2, ASKI output is stored at all inner GLL points of elements which are inside the volume defined by ASKI_(cw)(xyz). ASKI locates the wavefield points inside the inversion grid and computes weights.

4. (TYPE_INVERSION_GRID = specfem3dInversionGrid)
   Use SPECFEM elements as inversion grid:
   Wavefield points are *all* GLL points of an element for elements which are (at least partly) inside the volume defined by ASKI_(cw)(xyz). Additionally store the jacobians for all wavefield points. Assume ncell = ntot_wp/(NGLLX*NGLLY*NGLLY) as the number of inversion grid cells, and the order of wavefield points accordingly (do k=1, NGLLZ; do j=1,NGLLY; do i=1,NGLLX; ip=ip+1 ....)

5. (TYPE_INVERSION_GRID = chunksInversionGrid)
   NOT TO BE USED WITH SPECFEM3D Cartesian!
   ASKI internal, but SPECFEM independent more elaborate spherical inverison grid

ASKI_(cw)(xyz), ASKI_rot_(XYZ)

Dependent on ASKI_type_inversion_grid, (a selection of) the following parameters may be used to define a volume within which wavefield points are searched for:

First, ASKI_wx,ASKI_wy,ASKI_wz define the total width of a block which is centered in x=y=z=0 E.g. the total block extension in x-direction covers all points with
x >= - 0.5*ASKI_wx and x <= 0.5*ASKI_wx.
Then, ASKI_rot_X,ASKI_rot_Y,ASKI_rot_Z define rotation angles in degrees by which the block is rotated (anti-clockwise) about the Z, Y and X coordinate axis, before ASKI_cx,ASKI_cy,ASKI_cz define a vector by which the rotated block is shifted (new center of block).

Be aware:

- the parameters for rotation angles `ASKI_rot_(XYZ)` *must always* be assinged to values! Set to `0.` if no rotation should be applied.
- `scartInversionGrid` only supports `ASKI_rot_Z` and uses a different definintion of the z-coverage.
- `ecartInversionGrid` and `specfem3dInversionGrid` use *all* rotation angles `ASKI_rot_(XYZ)` (again, set angles to zero if no rotation is desired).

References

[Goe58] Gerald Goertzel. An algorithm for the evaluation of finite trigonometric series. *The American Mathematical Monthly*, 65(1):34–35, 1958. https://doi.org/10.2307/2310304.

[SF16] Florian Schumacher and Wolfgang Friederich. ASKI: A modular toolbox for scattering-integral-based seismic full waveform inversion and sensitivity analysis utilizing external forward codes. *SoftwareX*, pages –, 2016.

[SFL16] Florian Schumacher, Wolfgang Friederich, and Samir Lamara. A flexible, extendable, modular and computationally efficient approach to scattering-integral-based seismic full waveform inversion. *Geophysical Journal International*, 204(2):1100–1119, 2016.

[She68] Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference*, ACM '68, pages 517–524, New York, NY, USA, 1968. ACM.

History

This is a section on the history of this document, i.e. its previously published versions, as referred to by the GNU Free Documentation License (version 1.3).

Using SPECFEM3D_Cartesian-3.0 for ASKI versions 1.0 to 1.2, Aug 2016

Recognizable snippet from the title page (scaled):

Using SPECFEM3D_Cartesian-3.0 for
ASKI - Analysis of Sensitivity
and Kernel Inversion, versions 1.0 to 1.2
Aug 2016

Title: Using SPECFEM3D_Cartesian-3.0 for ASKI versions 1.0 to 1.2, Aug 2016
Year: 2016
Authors: Florian Schumacher (Ruhr-Universität Bochum, Germany)

This version of this document is provided for download (as of August 2016) at
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/tag/v1.2
Direct link (as of August 2016):
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/download/v1.2/SPECFEM3D_Cartesian_for_ASKI_manual_aug-2016.pdf

Using SPECFEM3D_Cartesian-3.0 for ASKI version 1.0, December 2015

Recognizable snippet from the title page (scaled):

Using SPECFEM3D_Cartesian-3.0 for
ASKI - Analysis of Sensitivity
and Kernel Inversion, version 1.0
December 2015

Title: Using SPECFEM3D_Cartesian-3.0 for ASKI version 1.0, December 2015
Year: 2015
Authors: Florian Schumacher (Ruhr-Universität Bochum, Germany)

This version of this document is provided for download (as of August 2016) at
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/tag/v1.0
Direct link (as of August 2016):
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/download/v1.0/SPECFEM3D_Cartesian_for_ASKI_1-0_manual_dec-2015.pdf

Using SPECFEM3D_Cartesian-2.1 for ASKI version 0.3, September 12, 2013

Recognizable snippet from the title page (scaled):

Using SPECFEM3D_Cartesian-2.1 for
ASKI - Analysis of Sensitivity
and Kernel Inversion, version 0.3
September 12, 2013

Title: Using SPECFEM3D_Cartesian-2.1 for ASKI version 0.3, September 12, 2013
Year: 2013
Authors: Florian Schumacher (Ruhr-Universität Bochum, Germany)

This version of this document is provided for download (as of August 2016) at
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/tag/v0.3
Direct link (as of August 2016):
https://github.com/seismology-RUB/SPECFEM3D_Cartesian_for_ASKI/releases/download/v0.3/SPECFEM3D_Cartesian_for_ASKI_manual.pdf

GNU Free Documentation License

Version 1.3, 3 November 2008

<http://fsf.org/>

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "**Document**", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "**you**". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "**Modified Version**" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "**Secondary Section**" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "**Invariant Sections**" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this

License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "**Cover Texts**" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "**Transparent**" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "**Opaque**".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "**Title Page**" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "**publisher**" means any person or entity that distributes copies of the Document to the public.

A section "**Entitled XYZ**" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "**Acknowledgements**", "**Dedications**", "**Endorsements**", or "**History**".) To "**Preserve the Title**" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may

accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only

one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be

placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See `http://www.gnu.org/copyleft/`.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you

have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.