Ses3d-NT

www.ses3d-nt.org

Stefan Mauerberger *mauerberger@geophysik.uni-muenchen.de*
Heiner Igel *igel@geophysik.uni-muenchen.de*
Department of Earth and Environmental Sciences (Geophysics)

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

3D seismic wave propagation through spherical sections

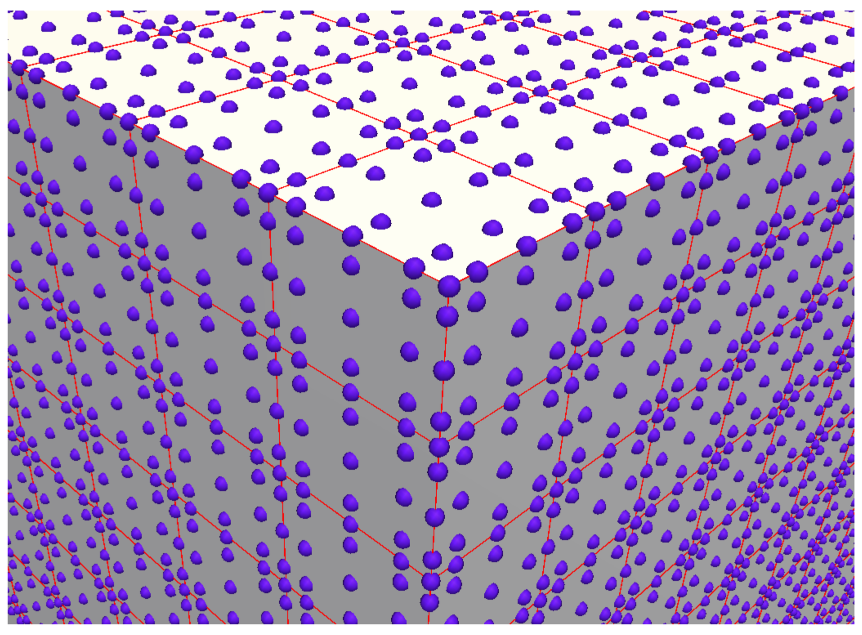
Ses3d-NT is based upon an algorithm and implementation initially developed by Andreas Fichtner. It is the project's key objective to provide an easy to use but flexible regular grid waveform solver. Combining Ses3d-NT with homogenization theory and current FWI problems may promise high efficiency in terms of computational recurses. It is developed as a consistent advancement of Andreas Fichtner's deliberately puristic SES3D providing an user friendly and highly flexible object-orientated framework for seismic wave propagation along with imaging capabilities. The whole development process - including documentation - is open to the public and is therefore driven under strict revision control. In addition, Ses3d-NT's code basis is licensed under the terms of GNU GPL version 3+. Its main features are:

- 3D seismic wave propagation in spherical sections from local to continental scales
- Calculations are carried out in natural spherical coordinates discretized using the spectral element method
- 3D heterogeneous Earth models covering anisotropy and visco-elastic dissipation
- Highly flexible, robust and machine-readable configuration-files
- Computation of sensitivity kernels with respect to model-parameters using the adjoint-method
- Fully transparent partitioning with semi-automatic parallelization scaling up to thousands of cores
- Unpartitioned file in- & output writing SAC-traces, netCDF-files and RAW binary blobs

The Spectral Element Method

The spectral-element method (SEM) is based upon the weak form of the wave equation. Therefore the free surface boundary condition is implicitly fulfilled. The SEM is an elegant way combining the flexibility of a finite-element method with the accuracy of spectral methods. Domain decomposition and discretization:

- Computational domain is split into Elements (red hexahedrons)
- High order polynomial representation sampled at the *Gauss-Lobatto-Legendre* points (blue blobs with polynomial degree $N = 4$)
- Integrations are carried out inside of each element by using Gaussian quadrature
- Spatial derivatives are expressed by the Lagrange polynomial's derivatives



Outline of Configuration Files

Splitting the configuration file into segments, makes the setup of a simulation highly flexible. Each segment encloses a sequence of logically related parameter-names along with their values. To do so, Fortran's well approved NAMELIST-parser is utilized. Each group needs to meet the following outline:

- Comments are indicated by the `!`-character
- Each segment starts with the `&`-character followed by a group-keyword
- A group must enclose a sequence of values assigned to parameter names
- Parameter/value pairs are either separated by commas or new-lines
- Each group must end with the `/`-character

Admissible groups-names are `time`, `grid`, `model`, `general` and others. Groups as `source`, `receiver` or `ouput_netcdf` may occur multiple times.

```
! Just a comment followed by a blank line

&time
    nt = 700 ! Number of time-steps
    dt = 0.75 ! Time-increment
              ! yyyy, mm, dd, dm_GMT,  h,  m, s, ms
    date_time = 2011, 2, 24,      0, 10, 15, 0,  0
/

&grid
    nx = 16 ! Number of elements in theta-direction
    ny = 26 ! Number of elements in phi-direction
    nz = 11 ! Number of elements in r-direction
    lpd = 4 ! Lagrange-Polynomial degree
    pml = 3 ! Width of relaxing boundaries (elements)
/

&receiver
  network='IU', station='PAB', location='00',
  lon=-4.3499, lat=39.5446, depth=0.0,
  attributes='N' 'E' 'Z'
/
```

Enhancements

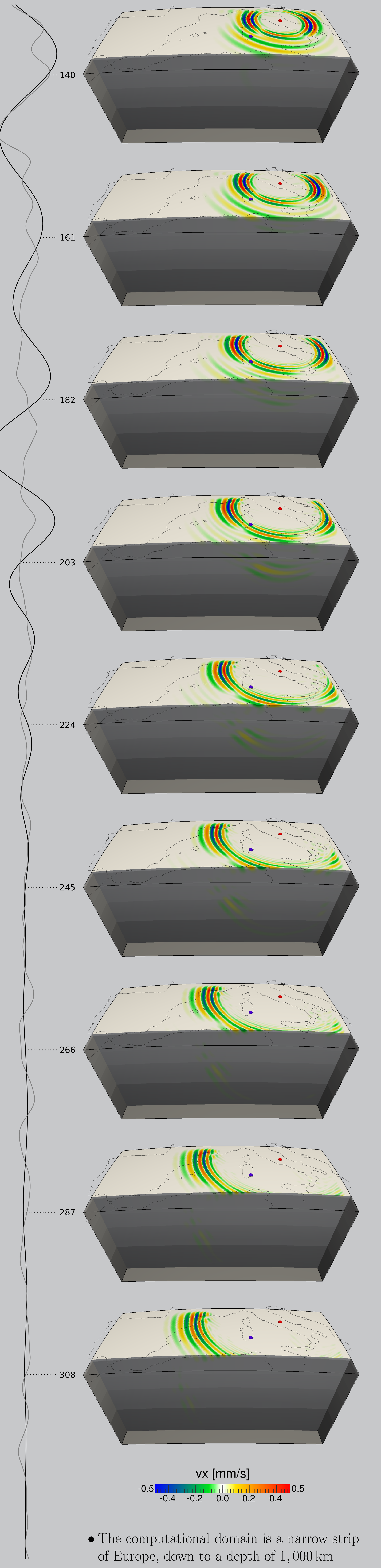
In comparison to SES3D, Ses3d-NT comes with a whole bunch of improvements. The code basis got restructured into modules and classes, received a elemental breakdown and got refactored with features provided by modern Fortran.

- Linkage between partitioning and parallelization got significantly weekend
- Parallelization is done semi-automatically
- Memory consumption got reduced by 1/4
- Allocation of arrays/fields is done dynamically
- Due to various optimization a speedup of 25% can be expected
- File input and output is based upon MPI Parallel I/O
- Volume snapshots can be written in netCDF-file format
- SES3D's *boxfiles* are no longer used
- Synthetics are written as SAC-trace files
- Easy to process, non-partitioned binary in- and output (model parameters, snapshots, kernels)
- Numeric portability through parameterized real types
- The number of sources and receivers may be chosen arbitrarily
- Dynamic and individual directory structures are supported
- Python framework for custom model generation and data pre- and post-processing

Future Plans

Within the upcoming year Ses3d-NT is expected to become part of VERCE. Accordingly it will be available on selected PRACE sites. In addition to documentation and tutorials, the following features will be implemented over time:

- Highly flexible and powerful (sub-domain) imaging workflow
- Wider variety of observables including stress, strain, acceleration and displacement along with their spatial derivatives, the curl and divergence
- Exhaustive netCDF-output covering multiple time-steps, receiver & source locations, vector- and tensor-fields
- Node-level optimizations of spatial derivatives (this is promising a huge performance boost)
- Nonblocking MPI communication may improve scalability
- Proper treatment of discontinuities by exchanging the flux amongst elements
- Improvement of stability of visco-elasticity and APMLs
- Extension of the python-toolbox for custom-model generation, pre- and post-processing
- A collection of test- and benchmark-cases
- Custom hexahedral but regular grids



- The computational domain is a narrow strip of Europe, down to a depth of 1,000 km
- As underlying earth model elastic, isotropic PREM has been chosen
- The grid-spacing is in the order of 65 km
- The exciting signal is a double-couple source with Ricker-pulse time evolution
- Simulated frequencies are in the band of ≈ 0.5 to 80 mHz