Specfem3D_Globe

Cédric Jourdain – 08/01/2020

Summary Version

v0.1

Purpose of Benchmark

The software package SPECFEM3D_GLOBE simulates three-dimensional global and regional seismic wave propagation and performs full waveform imaging (FWI) or adjoint tomography based upon the spectral-element method (SEM).

The test cases simulate the earthquake of June 1994 in Northern Bolivia at a global scale with the global shear-wave speed model named s362ani.

Test Case A is designed to run on Tier-1 sized systems (up to around 1,000 x86 cores, or equivalent), Test Case B is designed to run on Tier-0 sized systems (up to around 10,000 x86 cores, or equivalent) and finally the test case C is designed to run on PCP prototypes (up to around 100 cores, or equivalent).

Mechanics of building

To configure and compile Specfem3D_Globe:

1. Download Specfem3D_Globe software package :

   $ git clone https://github.com/geodynamics/specfem3d_globe.git

   Then use a fixed and stable version of specfem3D_globe (the one of October 31, 2017 for example, see https://github.com/geodynamics/specfem3d_globe/commits/master)

   $ cd specfem3d_globe

   $ git checkout b1d6ba966496f269611eff8c2cf1f22bcdac2bd9

2. Define the environment:

   a. You will need a Fortran and a C compiler and a MPI library and it is recommended that you explicitly specify the appropriate command names for your Fortran compiler in your .bashrc or your .cshrc file. To be exhaustive here are the relevant variables to compile the code: FC, MPIFC,CC.

   b. To be able to run on GPUs, you must define the CUDA environment by setting the following two variables: CUDA_LIB and CUDA_INC.

   c. To define the optimization specific to the target architecture, you will need the environment variables FCFLAGS and CFLAGS.

3. The “configure” script assumes that you will compile the code on the same kind of hardware as the machine on which you will run it. To configure specfem3D_Globe use the configure script with the “--enable-openmp” option. The “configure” script assumes that you will compile the code on the same kind of hardware as the machine on which you will run it.On GPU platform you will need to add the following arguments to the configure “--build=ppc64 --with-cuda=cuda5” and you will need to set the GPU_MODE flag to .true. in the parameter file “Par_file”

4. Finally, use “make clean && make all” to compile the code.

Mechanics of running

Input for the mesher (and the solver) is provided through the parameter file Par_file, which resides in the subdirectory DATA. Before running the mesher, a number of parameters need to be set in the Par_file. The solver calculates seismograms for 129 stations, and simulations are run for a record length of 3 minutes 30 for test case A, 10 minutes for test case B and one minute for test case C.

The different test cases correspond to different meshes of the earth. The size of the mesh is determined by a combination of following variables: NCHUNKS, the number of chunks in the cubed sphere (6 for global simulations), NPROC_XI, the number of processors or slices along one chunk of the cubed sphere and NEX_XI, the number of spectral elements along one side of a chunk in the cubed sphere. These three variables give us the number of degrees of freedom of the mesh and determine the amount of memory needed per core. The Specfem3D solver must be recompiled each time we change the mesh size because the solver uses a static loop size and the compilers know the size of all loops only at the time of compilation and can therefore optimize them efficiently.

- Test case A runs with 96 MPI tasks using hybrid parallelization (MPI+OpenMP or MPI+OpenMP+Cuda depending on the system tested) and has the following mesh characteristics: NCHUNKS=6, NPROC_XI=4 and NEX_XI=384.
- Test Case B runs with 1536 MPI tasks using hybrid parallelization and has the following mesh characteristics: NCHUNKS=6, NPROC_XI=16 and NEX_XI=384.
- Test Case C B runs with 6 MPI tasks using hybrid parallelization and has the following mesh characteristics: NCHUNKS=6, NPROC_XI=1 and NEX_XI=64.

Once the parameter file is correctly define, to run the test cases copy the Par_file, STATIONS and CMTSOLUTION files into the SPECFEM3D_GLOBE/DATA directory. Then use the xmeshfem3D binary (located in the bin directory) to mesh the domain and xspecfem3D to solve the problem using the appropriate command to run parallel jobs (srun, ccc_mprun, mpirun…).

Verification of results

The relevant metric for this benchmark is time for the solver. Using slurm, it is easy to gather as each mpirun or srun is interpreted as a step which is already timed. So the command line sacct -j <job_id> allows you to catch the metric. The output of the mesher (“output_mesher.txt”) and of the solver (“output_solver.txt”) can be find in the OUTPUT_FILES directory. These files contains physical values and timing values that are more accurate than those collected by slurm.