Geophysical Research Abstracts
Vol. 13, EGU2011-7876, 2011
EGU General Assembly 2011

Optimising the finite-volume multigrid code StagYY for thousands of parallel cores

Paul Tackley
ETH Zurich, Institute of Geophysics, Zurich, Switzerland (ptackley@erdw.ethz.ch)

The 3-D finite-volume multigrid code StagYY (Tackley, 2008 PEPI), which models both 3-D spherical shell and Cartesian geometries, has been used for many studies of thermo-chemical mantle convection and plate tectonics. However, until recently production runs were performed on up to only 64 cores, particularly on ETH's Opteron cluster, Brutus, and good scaling was obtained, with up to 1.2 billion unknowns solved for in ∼a few minutes. Here, optimisation of StagYY on up to 8192 cores of a Cray XT-5 is reported. The main problem to address is that the multigrid algorithm requires iterations on grids of different coarseness down to ones with only a few points in each direction, and these coarse grids do not run efficiently on a large number of cores because the execution time is dominated by communication latency. While this may initially seem unimportant because the time for one iteration is still very much smaller on coarse grids than on the finest grid, this is offset by the taking of more iterations on coarse grids in order to improve convergence, particularly on the coarsest grid where solution must be iterated to high convergence to obtain the 'exact' solution. After extensive timing of the iteration time for different grid resolutions on different numbers of cores, it was found that there is a minimum in the iteration time for a particular value of (#cells/core), with this optimum value varying slightly between Cartesian and spherical geometry and between 2-D and 3-D. Thus, in order to minimise solution time the multigrid algorithm was reconfigured such that coarse grids are handled on smaller and smaller subsets of cores, chosen to maintain (#cells/core) at no less than this optimum ratio. Use of MPI communicators simplified the implementation. The resulting code scales on up to 8192 cores in Cartesian geometry and up to 4096 cores in spherical geometry, with grids of up to 768x2304x512x2 (1.8 billion cells; 7.2 billion unknowns) tested. These results are encouraging but two points can be improved in future: (i) While good speedup is obtained using cores on different nodes, cores within the same node do not exhibit such good speedup, presumably due to competition for memory bandwidth. Future work will test different data structures and memory organisation in order to improve this. (ii) A three-dimensional domain decomposition is used. In Cartesian geometry the subdomains are kept as cubic as possible. However, in spherical geometry the Yin-Yang grid is used, in which the sphere is covered by two patches, similar to the construction of a tennis ball. This leads to complicated communication patterns if each patch is split into more than four subdomains, so a parallel decomposition of eight ways azimuthally by many ways radially is used. This becomes inefficient when the number of cores becomes large enough that there are only a few radial levels on each core (or in the extreme case, only one radial level). A future goal is thus to implement arbitrary decomposition in the azimuthal direction. Despite the latter two limitations, StagYY is already an effective tool on hundreds to thousands of CPUs, depending on the resolution.