

MAR14-2013-008907

Abstract for an Invited Paper
for the MAR14 Meeting of
the American Physical Society

ABINIT: Plane-Wave-Based Density-Functional Theory on High Performance Computers

MARC TORRENT, CEA, DAM, DIF, F-91297 Arpajon, France

For several years, a continuous effort has been produced to adapt electronic structure codes based on Density-Functional Theory to the future computing architectures. Among these codes, ABINIT [1] is based on a plane-wave description of the wave functions which allows to treat systems of any kind. Porting such a code on petascale architectures pose difficulties related to the many-body nature of the DFT equations. To improve the performances of ABINIT – especially for what concerns standard LDA/GGA ground-state and response-function calculations – several strategies have been followed: A full multi-level parallelisation MPI scheme has been implemented, exploiting all possible levels and distributing both computation and memory. It allows to increase the number of distributed processes and could not be achieved without a strong restructuring of the code. The core algorithm used to solve the eigen problem (“Locally Optimal Blocked Congugate Gradient”), a Blocked-Davidson-like algorithm, is based on a distribution of processes combining plane-waves and bands. In addition to the distributed memory parallelization, a full hybrid scheme has been implemented, using standard shared-memory directives (*openMP/openACC*) or porting some consuming code sections to Graphics Processing Units (GPU). As no simple performance model exists, the complexity of use has been increased; the code efficiency strongly depends on the distribution of processes among the numerous levels. ABINIT is able to predict the performances of several process distributions and automatically choose the most favourable one. On the other hand, a big effort has been carried out to analyse the performances of the code on petascale architectures, showing which sections of codes have to be improved; they all are related to Matrix Algebra (diagonalisation, orthogonalisation). The different strategies employed to improve the code scalability will be described. They are based on an exploration of new diagonalization algorithm, as well as the use of external optimized librairies. Part of this work has been supported by the european Prace project (PaRtnership for Advanced Computing in Europe) [2] in the framework of its workpackage 8.

[1] <http://www.abinit.org>

[2] <http://www.prace-ri.eu>