ABINIT : first-principles approach to material and nanosystem properties.

X. Gonze[⋆ a,b] B. Amadon[d] P.-M. Anglade[e] J.-M. Beuken[a,b]
F. Bottin[d] P. Boulanger[a,p,b] F. Bruneval[s] D. Caliste[j,b]
R. Caracas[m] M. Cote[p] T. Deutsch[j] L. Genovese[i,b]
Ph. Ghosez[ℓ,h] M. Giantomassi[a,b] S. Goedecker[c]
D. R. Hamann[n] P. Hermet[q,ℓ] F. Jollet[d] G. Jomard[d]
S. Leroux[d] M. Mancini[d] S. Mazevet[d] M. J. T. Oliveira[a,b]
G. Onida[o] Y. Pouillon[a,g,b] T. Rangel[a,b] G.-M. Rignanese[a,b]
D. Sangalli[o] R. Shaltaf[a,b] M. Torrent[d] M. J. Verstraete[a,t,g,b]
G. Zerah[d] J. W. Zwanziger[f]

[a]*Université Catholique de Louvain, Louvain-la-Neuve (Belgium)*

[b]*European Theoretical Spectroscopy Facility (ETSF)*

[c]*Department of Physics und Astronomy, Basel University, Basel (Switzerland)*

[d]*CEA, DAM, DIF, F-91297 Arpajon (France)*

[e]*Centre de recherche sur les ions, les matériaux et la photonique, UCBN–ENSICAEN–CNRS–CEA, Caen (France)*

[f]*Department of Chemistry and Institute for Research in Materials, Dalhousie University, Halifax (Canada)*

[g]*Nano-Bio Spectroscopy group and ETSF Scientific Development Centre, Dpto. Física de Materiales, Universidad del País Vasco, Centro de Física de Materiales CSIC-UPV/EHU- MPC and DIPC, 20018 San Sebastián (Spain)*

[h]*European Multifunctional Materials Institute (EMMI)*

[i]*European Synchrotron Radiation Facility, 6 rue Horowitz, BP 220, 38043 Grenoble (France)*

[j]*Commissariat à l'Energie Atomique, INAC/SP2M/L_Sim, Grenoble (France)*

[k]*Institut Néel, CNRS and Université Joseph Fourier, Grenoble (France)*

[ℓ]*Physique Theorique des Matériaux, Université de Liège (B5), B-4000 Liège (Belgium)*

[m]*CNRS, Laboratoire de Sciences de la Terre UMR 5570, Ecole Normale Superieure de Lyon, 46, allée d'Italie, 69364 Lyon (France)*

[n]*Mat-Sim Research LLC, P.O. Box 742, Murray Hill, NJ 07974 (USA)*

[o]*CNISM and Universitá Degli Studi di Milano (Italy)*

[p]*Département de physique, Université de Montréal, Montréal (Canada)*

^{q}*Laboratoire de Physique du Solide, Facultés Universitaires Notre-Dame de la Paix, Namur (Belgium)*

^{r}*LSI, CNRS-CEA, École Polytechnique, Palaiseau (France)*

^{s}*CEA, DEN, Service de Recherches de Métallurgie Physique, F-91191 Gif-sur-Yvette, France*

^{t}*University of York, York (United Kingdom)*

Abstract

ABINIT [1] allows one to study, from first principles, systems made of electrons and nuclei (e.g. periodic solids, molecules, nanostructures, ...), on the basis of Density Functional Theory (DFT) and Many-Body Perturbation Theory. Beyond the computation of the total energy, charge density and electronic structure of such systems, ABINIT also implements many dynamical, dielectric, thermodynamical, mechanical, or electronic properties, at different levels of approximation.

The present paper provides an exhaustive account of the capabilities of ABINIT. It should be helpful to scientists that are not familiarized with ABINIT, as well as to already regular users. First, we give a broad overview of ABINIT, including the list of the capabilities and how to access them. Then, we present in more details the recent, advanced, developments of ABINIT, with adequate references to the underlying theory, as well as the relevant input variables, tests and, if available, ABINIT tutorials.

PACS: 70; 71.15.Mb; 78

Key words: first-principles calculation, electronic structure, density-functional theory, many-body perturbation theory.

Program summary

Program Title: ABINIT
Journal Reference:
Catalogue identifier:
Licensing provisions: GPL [2]
Program summary URL: http://www.abinit.org/about
Programming language: Fortran95, PERL scripts, Python scripts
Computer: All systems with a Fortran95 compiler
Operating System: All systems with a Fortran95 compiler
RAM: ranges from a few Mbytes to several hundred Gbytes, depending on the input file
Number of processors used: sequential, or parallel with proven speed-up up to one thousand processors

---

Email address: `xavier.gonze@uclouvain.be` (X. Gonze*).

Distribution format: tar.gz
Keywords: first-principles calculation, electronic structure, density-functional theory, many-body perturbation theory
PACS: 70; 71.15.-m; 77; 78
Classification: 7.3 Electronic Structure, 7.8 Structure and lattice dynamics
External routines/libraries:(all optional) BigDFT[3], ETSF_IO[4], libxc [5], NetCDF [6], MPI [7], Wannier90[8].

Nature of problem:
This package has the purpose of computing accurately material and nanostructure properties : electronic structure, bond lengths, bond angles, primitive cell size, cohesive energy, dielectric properties, vibrational properties, elastic properties, optical properties, magnetic properties, non-linear couplings, electronic and vibrational lifetimes, ...
Solution method:
Software application based on Density Functional Theory and Many-Body Perturbation Theory, pseudopotentials, with plane waves, projector augmented-waves (PAW) or wavelets as basis functions.
Running time: From less than one second for the simplest tests, to several weeks. The vast majority of the >600 provided tests run in less than 30 seconds.
References:

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

[2] http://www.gnu.org/copyleft/gpl.txt

[3] http://inac.cea.fr/L_Sim/BigDFT

[4] http://etsf.eu/index.php?page=standardization

[5] http://www.tddft.org/programs/octopus/wiki/index.php/Libxc

[6] http://www.unidata.ucar.edu/software/netcdf

[7] http://en.wikipedia.org/wiki/Message_Passing_Interface

[8] http://www.wannier.org

1 Introduction

Quantum mechanics and electromagnetism are widely perceived as leading to a "first-principles" approach to materials and nanosystems: if the needed software applications and corresponding hardware were available, their properties could be obtained without any adjustable parameter (nuclei characteristics being given). Still, such "first-principles" equations (e.g. N-body Schrödinger equation) are too complex to be handled directly. Indeed, fundamental quantities, like the wavefunctions for N particles, cannot be represented faithfully on the hardware that is available nowadays, for N bigger than about a dozen.

Different methodologies to address this problem have been proposed. We will focus on the Density Functional Theory (DFT), proposed in 1964-65 by Hohenberg, and Kohn [1] and Kohn and Sham [2] (KS), and the Many-Body Perturbation Theory, in particular the so-called GW approximation (GWA) of L. Hedin [3], proposed in 1965. Both significantly reduce the complexity of the "first-principles" approach, at the expense of some approximations. On the basis of such methodologies, in the eighties, it became clear that numerous properties of materials, like total energies, electronic structure, and dynamical, dielectric, mechanical, magnetic, vibrational properties, can be obtained with an accuracy that can be considered as truly predictive (e.g. a few percent on bond lengths, a fraction of an eV for electronic energies...). Many research groups implemented the DFT methodology, using different representations (planewaves, augmented waves, muffin-tin orbitals ..., see the comprehensive book by R.M. Martin [4]). However, it was hardly possible for one research group to gather the expertise to address such wide range of properties.

The ABINIT software project started in 1997, with two basic ideas: (1) the needed expertise should be brought by different teams working together ; (2) taking as example the development of Linux, a "Free Software" licence should be at the heart of such a multi-team effort.

The first publicly available version of ABINIT was released in December 2000. Since then, the developer community has grown from a few early contributors, to about fifty people. The user community counts more than one thousand individuals worldwide. Because of its relatively large developer group, the computational capabilities of ABINIT cover a wide spectrum of properties, that will be described in the present paper. Also, special care has been taken to insure the portability of the application (thanks to non-regression tests) and to guide the user in the learning of ABINIT use. At present, the ABINIT package (31.5 MBytes), includes the main software application and post-processing tools (about 1200 files written in F90, 472,000 lines), automatic tests (657 input files), the corresponding documentation, as well as a complete build system relying on the autotools. The ABINIT project has already been described in

papers published in 2002 (Ref. [5]) and 2005 (Ref. [6]).

The present paper will start (Sec. 2) with an overview of ABINIT (user's point of view) : a list of capabilities of ABINIT, with explicit reference to internal documentation, some (very limited) information on the structure of its main executable, the structure of the package, how to build ABINIT and how to run ABINIT. Then, Secs. 3-7 will give a detailed presentation of features that have appeared in the last four years (that complements the information already contained in Refs. [5,6]). Successively :
- Sec. 3: Ground-state properties based on projector-augmented waves
- Sec. 4: Other ground-state properties
- Sec. 5: Linear and non-linear responses
- Sec. 6: Excited states
- Sec. 7: Speed-up of the calculation and convergence issues
In these sections, a brief theoretical account will be followed by adequate references to published literature, as well as to internal documentation:
(1) Web pages containing the description of input variables, with entry point, in the package, `doc/input_variables/keyhr.html`;
(2) Example input files (and corresponding output reference files), in the various `tests/*/Input/` (and `tests/*/Refs/`) subdirectories.

The comprehensive list of the capabilities of ABINIT, in Sec. 2.1, will include those that have been described in the early accounts of ABINIT. For such capabilities, there often exists an additional internal source of documentation, in the form of specific tutorials, in the `doc/tutorial/` directory. The underlying theory will often be found in the published literature, especially in the above-mentioned book by R.M. Martin [4]. As the citations to this "Electronic Structure" book (or chapter thereof) will be numerous, we will use the specific abbreviation RM_ES as a shortcut.

Throughout the paper, we refer to version 5.7 of ABINIT, released in January 2009. At the time of writing, a release candidate for ABINIT 5.8 exists, while the planning for ABINIT 6.0 is in preparation. The reference to these forthcoming versions will be kept to a minimum. The reader should however realize that some aspects of the present paper might quickly become obsolete, especially concerning missing capabilities of ABINIT.

The file addresses will always refer to the root of the package, when uncompressed. This location will be reached by issuing the following commands, in the directory where the abinit-5.7.4.tar.gz file resides :

```
gunzip abinit-5.7.4.tar.gz
tar -xvf abinit-5.7.4.tar
cd abinit-5.7.4
```

2 Overview of ABINIT

ABINIT is now presented from the user point of view. Accordingly, the list of capabilities will be rather exhaustive, while the description of the F90 sources will be quite short. In a similar spirit, to the parts of the package that are relevant for the users, the structure of directories will be explained in some detail, while other parts of the package, like the build system, will be nearly neglected. Two final subsections explain how to build and run ABINIT.

2.1 Capabilities of ABINIT

The standard representation of DFT and GWA quantities within ABINIT relies on planewaves, see Chapter 12 and 13 of RM_ES. An alternative representation is provided by a set of wavelets obtained with the BigDFT library, see Sec. 4.2. While very promising, the use of wavelets is not yet possible within the GWA, or for the direct computation of responses to perturbations. Periodic boundary conditions are implied when planewaves are used. In this case, systems that are not inherently periodic, like molecules, nanowires, or surfaces have to be embedded in the appropriate supercell, see Chapter 13 of RM_ES. This is at variance with the case of wavelets, where the available ABINIT/BIGDFT implementation relies on open boundary conditions.

Valence and core electrons are treated on different footings : the core electrons are frozen, and either replaced by norm-conserving pseudopotentials, or treated by the augmentation of plane waves by projectors (Projector Augmented-Wave method - PAW). The norm-conserving technology has been present for a long time in ABINIT, while the PAW methodology is more recent. The latter is described in Sec. 3 of the present paper. For the basic theory of pseudotentials and PAW, see Chapter 11 of RM_ES. It will be signalled whether the different properties can be accessed with the Plane Wave + Norm Conserving Pseudopotential (PW+NCPP) framework, the Projector Augmented-Wave framework (PAW) or even the wavelet (WVL) framework.

The treatment of the spin degree of freedom requires also some information. The easiest case for the implementation of a property arises when the system is non-spin-polarized (NSP): an occupation factor of 2, representing two electrons per state, appears in several DFT or GWA expressions, obtained from scalar wavefunctions. However, for many solids and molecules, there is a spontaneous imbalance between the number of electrons in both spin channels (collinear spin case - COLL), or, even, the wavefunctions have to be treated as spinors (in the presence of spin-orbit interaction, or non-collinear magnetism - SPINOR). Not all capabilities of ABINIT have been implemented in the collinear case or

6

Capability	Methodology and entry point
Total energy, charge density and forces for finite systems (molecules, clusters ...)	PW+NCPP (using a supercell) : doc/tutorial/lesson_1.html PAW (using a supercell) : Sec. 3.1 WVL (open boundaries) : Sec. 4.2 Open boundaries to treat the electrostatics for PW+NCPP and PAW : Sec. 4.3
Total energy, charge density and forces for periodic insulating systems (bulk solids, slabs, supercell...)	PW+NCPP and PAW : doc/tutorial/lesson_3.html, and, for PAW, Sec. 3.1
Total energy, charge density and forces for periodic metallic systems (bulk solids, slabs, supercell...)	PW+NCPP and PAW : doc/tutorial/lesson_4.html and, for PAW, Sec. 3.1
Geometry optimization or molecular dynamics	PW+NCPP, PAW and WVL : doc/tutorial/lesson_1.html and ionmov input variable (Broyden algorithm, viscous damping, Nosé thermostat, Langevin dynamics ...)
Stresses and primitive cell optimization	PW+NCPP, and PAW : doc/tutorial/lesson_3.html, with ionmov and optcell input variables (full optimization, uniform scaling, fixed volume optimization, fixed stress optimization ...)
Total energy, charge density, forces and molecular dynamics for high-temperature plasmas	PW+NCPP (local potentials only) : Sec. 4.4
Macroscopic polarization (Berry phase)	PW+NCPP : doc/tutorial/lesson_ffield.html, PAW : Sec. 3.4
Periodic systems under finite electric field (Berry phase)	PW+NCPP : doc/tutorial/lesson_ffield.html,
Collinear magnetization	PW+NCPP, PAW and WVL : doc/tutorial/lesson_spin.html
Non-collinear magnetization	PW+NCPP, PAW : doc/tutorial/lesson_spin.html
Antiferromagnetism	PW+NCPP, PAW and WVL : doc/tutorial/lesson_spin.html
Electric field gradients	PAW : Sec. 3.6
Mössbauer Isomer Shift	PAW : Sec. 3.6
Fermi contact interaction	PAW : Sec. 3.6
Positron lifetime	PW+NCPP, PAW : Sec. 4.1
Bader partitioning of density	PW+NCPP : doc/users/aim_help.html
Hirshfeld charges	PW+NCPP : doc/users/cut3d_help.html

Table 1
List of capabilities of ABINIT, arising from basic DFT. See text for the abbreviations. By default, the applicability is NSP, COLL and SPINOR for PW+NCPP, and NSP, COLL for WVL (spinors are not yet implemented in WVL).

spinor case, and this will be signalled. Explanations about the spin treatment can be found in Chapter 8 and 10 of RM_ES.

We will present in separate tables the capabilities of ABINIT that arise from :

7

DFT level of approximation	Availability and entry point
Local (Spin) Density Approximation (LDA)	PW+NCC, PAW, and WVL : 9 native ABINIT functionals, ixc input variable ; 16 Libxc functionals, Sec. 4.6
Generalized Gradient Approximation (GGA)	PW+NCC, PAW, and WVL : 10 native ABINIT functionals, ixc input variable ; 41 Libxc functionals, Sec. 4.6 (23 functionals for exchange, 8 functionals for correlation, 10 functionals for combined exchange and correlation)
LDA or GGA + U	PAW : Sec. 3.2
Hybrid functional	PAW : local PBE0, Sec. 3.3

Table 2

List of approximations available with DFT. See chapter 8 of RM_ES.

Capability	Methodology and entry point
KS electronic structure	PW+NCPP, PAW and WVL : doc/tutorial/lesson_1.html and doc/tutorial/lesson_3.html See Chapter 7 of RM_ES for the meaning of KS structure.
KS electronic Density of States	PW+NCPP : doc/tutorial/lesson_3.html PAW : Sec. 3.1
Atomic and angular momentum resolved KS electronic Density of States	PW+NCPP, PAW : doc/tutorial/lesson_3.html PAW : Sec. 3.1
Energy-resolved charge density, for Tersoff-Hamann approach to STM	PW+NCPP, PAW : prtstm input variable
Frequency-dependent optical and conductivity response	PW+NCPP : doc/users/conducti_manual.pdf or doc/tutorial/lesson_optic.html PAW : Sec. 3.5
Frequency-resolved non-linear optical response	PW+NCPP : doc/tutorial/lesson_optic.html
Wannier functions	PW+NCPP : doc/tutorial/lesson_wannier90.html
Excitation energies from Time-Dependent Density Functional Theory	PW+NCPP, for finite systems only, no SPINOR : doc/tutorial/lesson_tddft.html and Sec. 6.6

Table 3

List of basic capabilities of ABINIT, arising from the knowledge of Kohn-Sham (KS) electronic structure and from Time-Dependent Density Functional Theory. See text for the abbreviations. By default, the applicability is NSP, COLL and SPINOR for PW+NCPP, and NSP, COLL for WVL (spinors are not yet implemented in WVL).

- a direct implementation of DFT (Table 1);
- an implementation of Density Functional Perturbation Theory (Table 4), see Chapter 19 of RM_ES;
- an implementation of GWA (Table 5).

Capability	Applicability and entry point
Dynamical matrices, phonon frequencies and eigenvectors	doc/tutorial/lesson_rf1.html (also available in a finite electric field)
Dielectric tensor	doc/tutorial/lesson_rf1.html
Born effective charges	doc/tutorial/lesson_rf1.html and doc/tutorial/lesson_ffield.html
Infra-Red activity	doc/tutorial/lesson_rf1.html
Raman scattering intensity	(NSP only) Sec. 5.3 and doc/tutorial/lesson_nlo.html
Non-linear dielectric susceptibility	(NSP only) doc/tutorial/lesson_nlo.html
Electro-optic tensor	(NSP only) doc/tutorial/lesson_nlo.html
Phonon Density of states	doc/tutorial/lesson_rf2.html
Vibrational internal energy, free energy, entropy, as a function of temperature	doc/tutorial/lesson_rf2.html
Vibrational specific heat	doc/tutorial/lesson_rf2.html
Atomic form factors (Debye-Waller factors)	doc/tutorial/lesson_rf2.html
Electron-phonon coupling	(not SPINOR) Sec. 5.1
Phonon lifetime due to scattering with electrons	(not SPINOR) Sec. 5.1
Phonon-mediated superconductivity	(not SPINOR) Sec. 5.1
Phonon-limited electric conductivity	(not SPINOR) Sec. 5.1
Electron lifetime due to phonons	(not SPINOR) Sec. 5.2
Temperature dependence of the electronic eigenenergies (including the gap)	(not SPINOR) Sec. 5.2
Elastic tensor	(not SPINOR) doc/tutorial/lesson_elastic.html
Piezoelectric tensor	(not SPINOR) doc/tutorial/lesson_elastic.html
Thermal expansion	From minimization of the total free energy, see Ref. [7]
Photoelasticity	From finite-difference of dielectric tensor, see Ref. [8]

Table 4

List of basic capabilities of ABINIT, arising from Density-Functional Perturbation Theory (DFPT), in the PW+NCPP methodology. DFPT is not yet available in production within PAW in ABINITv5.7, but should be available in ABINITv6.0 . No implementation of DFPT is yet started in WVL. By default, the applicability is NSP, COLL and SPINOR.

In Table 2, we list different levels of approximation implemented within DFT, complementing Table 1. In Table 3, we list capabilities of ABINIT that are derived from the KS electronic structure, also complementing Table 1. As explained in Chapter 7 of RM_ES, the KS electronic structure and the experi-

Type of self-consistency	Type of self-energy, and type of frequency integration for dynamical correlations
G_0W_0	Σ_{HF}
GW_0 (update of eigenenergies)	Σ_{COHSEX}
G_0W	Σ_{GW} with 4 plasmon-pole models
SCqpGW	Σ_{GW} with contour deformation
	Σ_{GW} with analytical continuation
	$\Sigma_{GW\Gamma}$ with 4 plasmon-pole models

Table 5

List of available methodologies within Many-body Perturbation Theory. See text for the abbreviations. By default, the applicability is NSP and COLL (spinors are not implemented), for PW+NCPP and PAW (no WVL). The entry points are Sec. 6, and the tutorials `doc/tutorial/lesson_gw1.html` and `doc/tutorial/lesson_gw2.html`. SCqpGW stands for Self-Consistent quasi-particle GW approximation.

Framework	Approaches	Methodology and entry point
DFT	Parallelism over wavevectors (k points) and spin	PW+NCPP and PAW doc/tutorial/lesson_parallelism.html
DFT	Parallelism over bands	WVL, Sec. 4.2
DFT	Parallelism over FFT planes, bands, k points and spin	PW+NCPP and PAW Sec. 7.1
DFT	Extrapolar speed-up of self-consistency	Sec. 7.3
GWA	Parallelism over wavevectors	Sec. 7.2
GWA	Parallelism over empty bands	Sec. 7.2
GWA	Extrapolar decrease of the number of empty bands	Sec. 7.4
GWA	Wannier interpolation of the quasi-particle band structure	Sec. 6.5

Table 6

Specific approaches to speed-up the calculations. See text for the abbreviations.

mental quasi-particle electronic structure cannot directly linked to each other. However, scientists have often derived properties of systems from the KS structure, as this approach is rather simple and fast from a computational point of view, convenient, and often qualitatively meaningful. The direct computation of excitation energies from Time-Dependent Density Functional Theory, see Sec. 6.6, is also listed in Table 3. The specific techniques related to the speed of ABINIT will be presented in Table 5, be these related to parallelism, or to speed-up of convergence or interpolation issues.

For each capability, we give the appropriate entry point. For the most basic properties, already present in ABINIT in 2005, Web-formatted tutorials exist, in the `doc/tutorial/` directory. This will be the preferred entry point mechanism, from which the relevant input variables are introduced, with commented
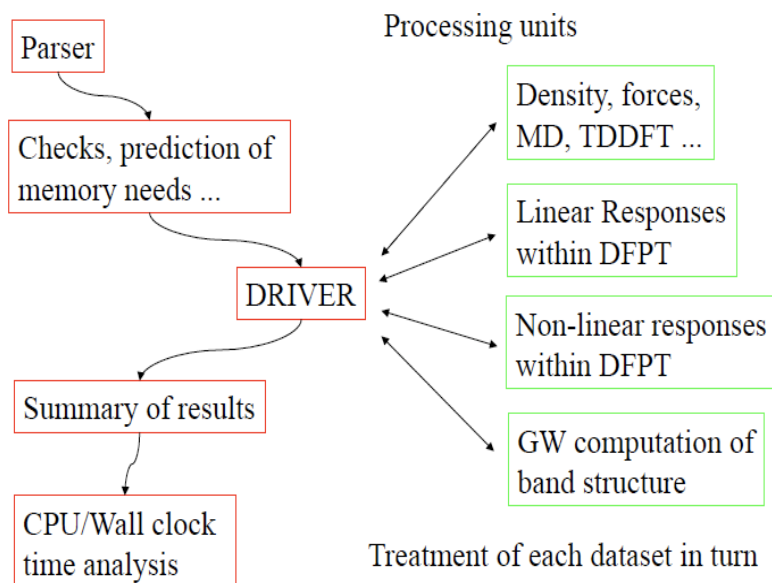
Fig. 1. Structure of the main ABINIT program, from the user's perspective.

examples. For the capabilities developed between 2005 and 2009, the Sections 3-7 of the present paper provide the best entry points, with appropriate reference to ABINIT input variables (entry point `doc/input_variables/keyhr.html`), and example test cases. Finally, for some minor capabilities of ABINIT, neither a tutorial nor a relevant description in the present paper exist. In these cases, some other entry points from the package will be listed in the table, together with possible reference to RM_ES.

In any case, for the ABINIT newcomer, it is not advised to jump immediately to the entry point that is closest to his/her research concerns. This person should first spend some time reading the new user's guide (`doc/users/new_user_guide.html`), as well as lessons 1 to 4 of the tutorial (in `doc/tutorial/`, the files `lesson_X.html` with X being 1, 2, 3 or 4).

2.2   The main ABINIT program, and other main programs

The sources of the different F90 programs of the ABINIT package are present in the `src/` directory. The corresponding top-level routines have been gathered in the `src/main/` subdirectory. One of these top-level routines, `src/main/abinit.F90` drives all the CPU-intensive computations needed for the capabilities mentioned in Sec. 2.1.

A detailed understanding of the structure of this main ABINIT program (referred to as `abinit`) is not required for the user to carry out calculations.

11

Moreover, the content of the output file of ABINIT is relatively self-contained. One aspect of the structure of the main ABINIT program is however apparent from the user's perspective, and has been sketched in Fig. 1, namely, the presence of "processing units", attacked with a "driver".

Indeed, in the input file, the user can define different "datasets", thanks to the input variable **ndtset**, that `abinit` will treat in turn, sequentially. The output of one dataset can be taken as output of the following datasets. The results of each dataset calculation are listed sequentially in the output file. The processing units correspond broadly to the different theoretical frameworks that have been presented in Sec. 2.1.

Let us inspect the top-level `abinit` source file, `src/main/abinit.F90`, and locate the elements on the left-hand side of Fig. 1 :
- The box "parser" refers to the call to `parsefile`, and its subsequent analysis thanks to the calls to `invars0`, `invars1m`, `invars2m` (the corresponding source files being `parsefile.F90`, `invars0.F90`, `invars1m.F90`, `invars2m.F90`, in the `src/13iovars` directory);
- The box "Checks, prediction of memory needs" refers to the calls to `chkinp`, and `chkvars` (source files in the `src/13iovars` directory);
- The box "DRIVER" refers to the call to `driver`
(source file `src/21drive/driver.F90`) ;
- The box "Summary of results" refers to the call to `outvars` (source file `src/13iovars/outvars.F90`) ;
- The box "CPU/Wall clock time analysis" refers to the call to `timana` (source file `src/21drive/timana.F90`).
Of course, `src/main/abinit.F90` accomplishes many more tasks, hidden to the user : initializations, file name handling, allocation of memory, transfer of data between variables or between processors.

In the `src/21driver/driver.F90` routine, one finds the elements present in the right-hand side of Fig. 1:
- The loop over the different datasets is driven by the line
"`do idtset=1,ndtset_alloc`" ;
- The box "Density, Forces, MD, TDDFT" refers to the call to `gstate` (source file `src/21drive/gstate.F90`);
- The box "Linear Responses within DFPT" refers to the call to `respfn` (source file `src/18seqpar/respfn.F90`);
- The box "Non-linear Responses within DFPT" refers to the call to `nonlinear` (source file `src/21drive/nonlinear.F90`);
- The box "GW Computation of band structure" refers to the calls to `screening` and `sigma` (source files `screening.F90` and `sigma.F90` in `src/18seqpar/`).
In `src/21driver/driver.F90`, in addition to the above-mentioned elements (as well as initializations, copy of data, etc), one finds also calls to processing units that are not associated to any capability explained in the present paper.

These units correspond to functionalities that are not yet operational.

We will not proceed further with the description of the structure of the main `abinit` program, the inspection of the enormous number of source routines being outside of the scope of this paper.

Other top-level routines of the `src/main/` directory will lead to either post-processors of the main ABINIT program, or utilities to manipulate the output files :
- `anaddb` is a code to perform the analysis of the DFPT output of `abinit`, whose use is explained in `doc/tutorial/lesson_rf2.html` and `doc/users/anaddb_help.html`;
- `optic` is a code to compute frequency-dependent linear and non-linear optics properties, based on KS eigenvalues and eigenfunctions from `abinit`, whose use is explained in `doc/tutorial/lesson_optic.html` and `doc/users/optic_help.html`;
- `cut3d` is a code to perform the analysis of the density and wavefunction files of `abinit`, whose use is explained in `doc/users/cut3d_help.html`;
- `aim` is a code to perform the Bader analysis of the charge density from `abinit`, whose use is explained in `doc/users/aim_help.html`;
- `macroave` is a code to perform the macroaverage of the charge density or potentials from `abinit`, whose use is exemplified in `tests/v4/t4X.in` where X stands for 0 to 5. See also `tests/v4/README`;
- the other top-level routines in `src/main/` correspond to utilities to manipulate different data files, whose usage is explained in the relevant tutorial or help files.

Having described the top-level routines of the `src/main/` directory, we now turn to the description of the overall structure of the package.

2.3   Structure of the package

ABINIT is made of 9 logical blocks, fully reflected in the structure of its source tree (see Fig. 2).

The core of ABINIT is composed of the `src/` subtree, which contains all the Fortran source files of the package. The subdirectories of `src/` are organized hierarchically. The numbers in front of the directory names indicate the presence of dependencies on other directories having a lower number.

The prerequisites are found in the `prereqs/` subtree. They are external components to ABINIT. Their presence is required to make sure that ABINIT can be built and used. At present this part contains the subset of BLAS and LAPACK routines ABINIT is calling in the course of its execution.
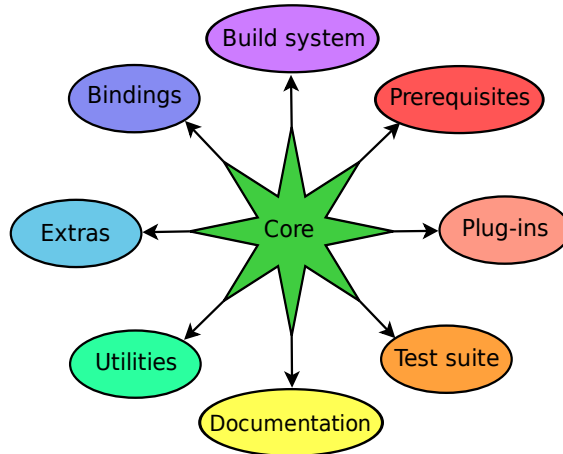
13

Fig. 2. Structure of the ABINIT package.

The so-called "plug-ins", found in the `plugins/` subtree, are external components as well, but they are optional. Their build system has been wrapped by the one of ABINIT, in order to facilitate their build and use within our package. They extend the capabilities of ABINIT in various ways:

- BigDFT[61] uses wavelets to deal with systems made of thousands of atoms;
- NetCDF[84] and ETSF_IO[85] allow for a platform-independent exchange of data between different software packages (currently supported by DP[86], EXC[87], and Yambo[82]);
- LibXC, from Octopus[78], provides a comprehensive, robust, and well-tested collection of exchange-correlation functionals;
- FoX[88] is able to read and write XML files from Fortran and will soon replace the currently supported XMLF90[89], which it supersedes;
- Wannier90[90] makes it possible to perform quantum transport calculations using Maximally-Localized Wannier Functions (MLWF).

The test suite is a very important component of ABINIT. It is now made of more than 600 tests, allowing for the checking of most of the package features, including the running in parallel and the plug-ins. The tests are organized in series. In addition to 5 built-in quick tests, there is one series for each major version of ABINIT (around 100 tests per series) and for each plug-in. Two other series cover the basic and response-function tutorials, providing the ability to verify that they can actually be run and give correct results, which is extremely useful when preparing hands-on sessions. The "*fast*" series is meant for quickly and roughly knowing whether the binaries behave correctly, but is limited when it comes to pinpointing problems. The "*cpu*" series is used for benchmarks. The "*physics*" series performs a production run and is not used much because of the resources it consumes. Last but not least, and contrary to all the other series, "*paral*" provides 70 tests to make sure that the parallel

version of ABINIT works fine.

The documentation is located in the `doc/` subtree. It contains a lot of information for users, developers, and maintainers. Here we will cover the first category only. The user documentation is threefold: generic information, help to build ABINIT, and help to run ABINIT.
The `theory/` subtree contains theoretical background on some of the main features of ABINIT. The ABINIT project and its philosophy are described in `presentation/`.
Most of the necessary information to build ABINIT can be found in `build/`, containing a manual, in `config/`, containing a configuration template and examples, as well as in `install_notes/`. It is also usually a good idea to consult the `KNOWN_PROBLEMS` file at the top of the source tree before starting to build ABINIT.
The `users/` subtree stores the manuals for all binaries and some of the advanced features of ABINIT, a FAQ and some methodological help. The capabilities of the programs are described in `features/`, while `release_notes/` makes a list of the latest important news in the history of the package. Of utmost importance is the `tutorial/` subtree, in particular for beginners, for all the useful information it contains.

Most of the ABINIT utilities, stored in the `util/` subtree, are meant for developers and maintainers. A few of them are however of interest to users. In particular, in `util/users/`, the `ABINITInputMaker.py` Python script can help beginners creating input files, while `ABINITBandStructureMaker.py` will prove useful to get and plot band structures from an ABINIT calculation. Another script, `chkinabi.pl`, looks for typos in input files. A few post-processing goodies can also be found in `extras/`.

Recently a project has been started to make parts of ABINIT directly usable by other software packages. This is what can be found in the `bindings/` subtree. A pilot test is currently being conducting with V_Sim, which is now able to parse ABINIT input files using the parser of ABINIT. In the future, other parts may become available, like the symmetry finder or the reader of ABINIT file headers. The feasibility of such projects is still being studied.

Last but not least, the build system of ABINIT is located in the `config/` subtree. It consists in a software layer which has been developed on top of the GNU Autotools (Autoconf, Automake, Libtool) and becomes invisible once a source tarball has been made, so that end-users do not have to care about anything related to it. See the Ψ_k Newsletter Highlight #90 (`http://www.psi-k.org/newsletters/News_90/Highlight_90.pdf`) for a detailed description of the ABINIT build system.

The build of ABINIT follows the well-established and ubiquitous procedure of Free Software packages:

```
mkdir <builddir>
cd <builddir>
../configure [options]
make
make check
make install
```

As one of the priorities of ABINIT is portability, the counterpart is that its configure script provides many options. The full list can be obtained by typing "`configure --help`". We will just cover here the most important ones.

All plug-ins, as well as a few other libraries (FFTW, PAPI, ScaLAPACK) can be customized through the use of 3 options:

- `--enable-<package>`: enable / disable support for the package;
- `--with-<package>-includes`: include flags to use at compile time;
- `--with-<package>-libs`: link flags to resolve symbols at link time.

For plug-ins, if the first option is used only, the version embedded in ABINIT will be built before the core sources. For the other ones, at least the first and third options will have to be used. A noticeable exception is the case of BLAS and LAPACK, that we have grouped and nicknamed "*linalg*", which cannot be disabled and which will be systematically built if the third option is not used.

The most delicate options are the MPI-related ones, and most of the difficulties in dealing with them arise from that the sequential (abinis) and parallel (abinip) binaries are built at once. The infinite possibilities in choosing the MPI implementation, its version, and where to install it make the configuration more complicated and more difficult too. However, in many cases, the use of the `--with-mpi-prefix` option solves all the problems, by giving the build system a location where to look for what it needs and letting it set the build parameters accordingly. The most frequent mistake is to forget to put the `<mpi_prefix>/lib/` directory in the `LD_LIBRARY_PATH` environment variable, which causes the configure script to abort.

Additionally, abinis should not in principle depend on MPI libraries, which means in most cases that it should be built with a different compiler than the mpif90 wrapper. This is unfortunately incompatible with having abinis and abinip built at the same time. The best compromise we have found so far is to

override the definitions of CC and FC with the compiler wrappers provided by MPI. This trick lets us make sure that both binaries are built with the same compiler, at the price of abinis depending on MPI libraries.

In order not to have to memorize the configure options between each build, it is possible to store them in a config file. The `abinit/doc/config/build-config.ac` file is a fully documented template, containing as well instructions on where to put it once it has been customized. When running configure, the build system looks for a config file in the following locations:

- system-wide (in `/etc/abinit/build/hostname.ac` on Unix-compliant systems);
- in the user's home directory;
- in the top source directory of ABINIT;
- in the top of the build directory;
- specified from the command line (using the `--with-config-file` option of configure).

The latest one found in this list wins, i.e. all other files are simply ignored. It is also possible to disable the use of a config file from the command line, when desirable (`--disable-config-file`).

Regarding customization, the build system of ABINIT distinguishes between 4 levels:

- default values;
- values stored in the config file;
- values specified on the command line;
- values stored in environment variables.

When there are conflicting values at different levels, the order of precedence is the following: the defaults are overridden by the config file, itself superseeded by the command-line options, which are in turn ignored if an environment variable of the same name is set.

This strategy (configure options + config files + precedence) allows for a very high flexibility. This lets ABINIT adapt to various platforms — ranging from personal computers to HPC environments to build farms — without having to hack into the build system, while bringing a fair amount of comfort to most users. It is nevertheless important to set the build environment carefully, some environment variables such as `PATH` or `LD_LIBRARY_PATH` having a critical influence on the build system behaviour. Most of the problems encountered at configure time actually come from improper settings of these variables.

While typing "`make`" will suffice in most cases to have the binary executables built, the build system of ABINIT also offers several options. One can build
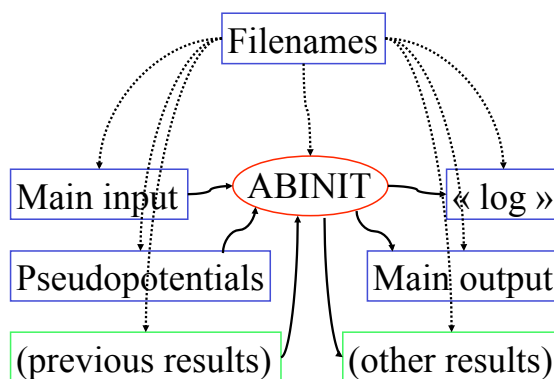
Fig. 3. Handling of external files in an ABINIT run.

just one binary executable by typing "`make <name_of_binary>`", or take advantage from a parallel platform by typing "`make mj4`". Please note that the usual "`make -j`*n*" will not work straightforwardly with ABINIT 5, due to some of the plug-ins not having support for it and too complicated interdependencies within the core sources. Hence the "*mj4*" target. This issue has already been solved and tested up to 16 processors for the upcoming ABINIT 6. Typing "`make help`" in various directories will display a help message.

After the binary executables have been successfully built, and before installing them, it is extremely important to check that they do what they are supposed to do, in particular if the results obtained from them are to be published in the mainstream scientific literature. The simplest way to do that is to run the whole test suite by typing "`make check`", which should be run at least once on every production machine before starting to use any new version of ABINIT.

Once it has been ensured that ABINIT is working on the platform it has just been built on, it is time to install it through "`make install`". What is installed includes all binary executables, the whole documentation and the whole test suite. By default, in ABINIT 5, everything is installed in `/opt/etsf/abinit/x.y/`, where x and y are respectively the first two digits in the version number. If a prefix has been specified at configure time, it replaces the "`/opt/etsf`" part. We are aware that this way of setting the install path is not a standard practice and can already say that ABINIT 6 will behave more conventionally in this respect.

2.5   How to run ABINIT

In the present section, we will suppose that the sequential version, `abinis`, is used. After the build, it is present in the package as `src/main/abinis`, or

```
opt/etsf/abinit/5.7/abinis .
```

As shown in Fig. 3, to run `abinis`, one needs at least four ingredients:
(1) Access to the executable (the center of Fig. 3);
(2) A list of file names, contained in a separate file ("Filenames" in Fig. 3);
(3) An input file ("Main input" in Fig. 3);
(4) A pseudopotential input file for each kind of element in the unit cell ("Pseudopotentials" in Fig. 3).

Supposing the file containing the list of file names is indeed named "Filenames", then, in order to launch abinis, issue the command (Unix):
```
abinis < Filenames
```
Alternatively, one can pipe the standard output in a file, e.g. "log", as follows:
```
abinis < Filenames > log
```
In addition to the standard output, the run will produce at least one file, called "Main output" in Fig. 3. While the standard output ("log") is rather verbose, and should not be examined except if problems happen, the "Main output" contains relevant information on how the run went, and important results such as total energy, forces, stresses, ..., in a self-descriptive format.

In order to learn how to construct an input file, please consult the tutorial, `doc/tutorial/lesson_1.html`, or the `doc/users/abinis_help.html` file. Example input files are presented in the various `tests/*/Input/` subdirectories.

Different sorts of pseudopotentials can be used with ABINIT. They are available from the Web site (http://www.abinit.org). Concerning norm-conserving pseudopotentials, there are two (nearly) complete sets available, as well as several dozen pseudopotentials of other types. PAW atomic datafiles exist, and a complete periodic table is in preparation. A subset of existing pseudopotentials is used for test cases, located in the `tests/Psps_for_tests` directory, that includes the complete set of Hartwigsen-Goedecker-Hutter LDA potentials [63]. Links to utilities to generate more pseudopotentials are mentioned on the Web site. Information on pseudopotential files can be found in the `doc/users/abinis_help.html` file and the `doc/psp_infos/` directory.

Given an input file and the required pseudopotential files, the user must create a "Filenames" file which lists names for the files the job will require, including the main input file, the main output file, root names for other input, output, or temporary files, and different pseudopotential file names.
Such a "Filenames" file (called for example ab.files) could look like:

```
ab_in
ab_out
abi
```

```
abo
tmp
14si.psp
```

In this example:
- The main input file is called "ab_in" ;
- The main output will be put into the file called "ab_out" ;
- The name of the files containing previous results (if any) will be built from the root "abi" (e.g. abi_WFK, see later) ;
- Output files will be build from the root "abo" (e.g. abo_WFK, see later) ;
- The temporary files created by ABINIT will have a name that use the root "tmp" (e.g. tmp_STATUS) ;
- The pseudopotential needed for this job is "14si.psp".

Finally, in Fig. 3, the last information to be mentioned concerns the "previous results" and "other results". These refer to binary or NetCDF files, that contain numerical data, like wavefunctions, density, potentials. The list of such files, their nomenclature, the settings needed to produce them, or to read them, are described in `doc/users/abinis_help.html`. In the context of the present paper, it suffices to know that the name of files containing wavefunctions usually ends with `_WFK` (for WaveFunctions at wavevector K), or `_KSS` (for Kohn-Sham Structure), and the name of files containing the density ends with `_DEN`.

3 Developments related to ground-state calculations with the Projector-Augmented Wave methodology

3.1 The projector augmented-wave method

General framework

The PAW (Projector Augmented-Wave) method has been introduced by Peter Blochl [9] in 1994. It is an extension of augmented wave methods and the pseudopotential approach, which combines their traditions into a unified electronic structure method. It is based on a linear and invertible transformation (the PAW transformation) that connects the "true" wavefunctions Ψ_{nk} (where nk refers to a band index n and a crystalline momentum $\mathbf{k}$) with the "auxiliary" (or "pseudo") soft wavefunctions $\tilde{\Psi}_{nk}$:

$$|\Psi_{nk}\rangle = |\tilde{\Psi}_{nk}\rangle + \sum_i (|\Phi_i\rangle - |\tilde{\Phi}_i\rangle)\langle \tilde{p}_i|\tilde{\Psi}_{nk}\rangle. \tag{1}$$

The global index i is used to identify the various partial waves located at the different atomic positions $\mathbf{R}$, of angular momentum (l, m) and possibly an additional label if more than one partial wave is required to accurately describe some atomic angular momentum channel.

This relation is based on the definition of atomic spheres (augmentation regions) of radius r_c around the atoms of the system in which the partial waves Φ_i form a basis of atomic wavefunctions; $\tilde{\Phi}_i$ are "pseudized" partial waves (obtained from Φ_i), and $\tilde{p}_i$ are dual functions of the $\tilde{\Phi}_i$, called projectors.

It is therefore possible to write every quantity depending on Ψ_{nk} (e.g. density, energy, Hamiltonian) as a function of $\tilde{\Psi}_{nk}$ and to find $\tilde{\Psi}_{nk}$ by solving self-consistent equations.

The PAW method has two main advantages:
- From $\tilde{\Psi}_{nk}(\mathbf{r})$, it is always possible to obtain the true "all electron" wavefunction $\Psi_{nk}(\mathbf{r})$.
- The expansion of the $\tilde{\Psi}_{nk}(\mathbf{r})$ ini term of plane waves is comparable to the ultrasoft pseudopotential scheme.
This effect of this last property can be seen, for instance, in *tutorial PAW1* provided in ABINIT. For a single diamond crystal structure, convergence of total energy at 1 mHa precision is achieved for a plane wave cut-off energy of 25 Ha when using a Troullier-Martins pseudopotential where as a plane wave cut-off energy of 12 Ha is needed using PAW.

From a practical point of view (user's point of view), a PAW calculation is rather similar to a norm-conserving pseudopotential. Most noticeably, one will have to use a special atomic data file (*PAW dataset*) that contains the $\Phi_i, \tilde{\Phi}_i$

and $\tilde{p}_i$ and that plays the same role as a pseudopotential file.

Available functionalities
The detailed implementation of the PAW method in ABINIT is discussed in Ref. [10]. In this paper are described the notations used here, the analytic formula to compute forces and stresses, numerical details concerning the calculation of real spherical harmonics, the non-local term and the exchange-correlation terms in the PAW atomic spheres.

Quite a large number of PAW functionalities have been implemented for the ground state: total energy, analytic computation of forces and stresses which allows geometry optimization and molecular dynamics, spin polarized and non-colinear spins calculations, electric and thermal conductivity, Berry phase calculation and calculation of total and partial Density of States.

When the electronic orbitals are localized into the PAW spheres, and if the partial wave basis is complete, we have for these orbitals:

$$|\tilde{\Psi}_{nk}\rangle \approx \sum_i |\tilde{\Phi}_i\rangle\langle\tilde{p}_i|\tilde{\Psi}_{nk}\rangle,$$

so that the relation (1) reduces to

$$|\Psi_{nk}\rangle \approx \sum_i |\Phi_i\rangle\langle\tilde{p}_i|\tilde{\Psi}_{nk}\rangle. \tag{2}$$

In this situation, it is therefore possible to develop functionalities that are fully localized in the PAW augmentation regions only. Examples of such functionalities, as described in later sections, are: LDA+U, local hybrid functionals (PBE0), spin-orbit coupling and core spectroscopy calculations.

Accuracy and efficiency
In order to ensure maximum accuracy of the results and to avoid issues due to numerical precision, each physical quantity is computed using the derived analytical formula. Use of numerical derivatives, splines or series developments has been avoided as far as possible and the refinement of the grid spacing is adjustable using keyword parameters in the input file (.in). As an example, we can focus on the computation of the stress tensor which full expression is given in Ref. [10] (see Appendix D). In addition to the plane wave part, development of stresses on the partial waves has been analytically expanded. It involves the use of spherical harmonics derivatives analytically computed at an arbitrary precision level.

To improve the speed performance of the code, the approach adopted was to compute once and store constant quantities as much as possible. This

is especially the case for "on-site" PAW quantities. For example, the non-local pseudopotential strengths (see Appendix E in Ref. [10]) is expressed as $D_{ij} = D_{ij}^0 + \sum_{kl} \rho_{kl} e_{ijkl} + D_{ij}^{XC} + \hat{D}_{ij}$ and the atomic quantities D_{ij}^0 and e_{ijkl} are precomputed before the self-consistency loop. Also, the moments of the compensation charge density $\hat{Q}_{ij}^{LM}(\mathbf{r}) = q_{ij}^{LM} g_L(r) S_{LM}(\hat{r})$ are computed for each new set of atomic positions and stored in memory at the beginning of the execution. As it is often the case, speed efficiency and memory requirement are in competition and the present PAW implementation is no exception as these constant PAW quantities may need quite a lot of memory. In order to help the user to choose the best compromise for his or her computational architecture, many parameters can be adjusted from the input file (see "*PAW variables*" in ABINIT documentation). It it thus possible to compute some quantities *on-the-fly* instead of storing them. By default all parameters are adjusted to "accuracy" (instead of "efficiency") and all constant quantities precomputed.

Self-consistent cycle and mixing
The robustness of ABINIT PAW calculations is largely influenced by the behaviour of the electronic density during the self-consistent cycle which is the core of the resolution of KS equations. The efficiency of the self-consistent cycle can be improved by the use of sophisticated mixing and preconditioning schemes (see for instance Section 7.3). This is particularly needed in the case of PAW because additional degrees of freedom are added to the electronic density written now as $n(\mathbf{r}) = \tilde{n}(\mathbf{r}) + \hat{n}(\mathbf{r})$, where $\tilde{n}(\mathbf{r})$ is the pseudo valence density and $\hat{n}(\mathbf{r})$ is the compensation charge density. The latter is directly related to the PAW occupation matrix ρ_{ij} which also has to be taken into account in the mixing scheme.

Since the first versions, ABINIT used the potential as the "mixed" variable; at the end of each self-consistent cycle step the potential residual is preconditioned and then mixed with previous potential in order to obtain the next potential. We added the same mixing scheme for the ρ_{ij} matrix. But this choice is not optimum as ρ_{ij} is not directly involved in the computation of the potential. Since this quantity is related to the density, it would be better to work solely the electronic density. For this reason, we modified the self-consistent cycle of ABINIT in order to mix the density instead of potential. "Potential mixing" or "density mixing" can be activated from a keyword parameter indicated in the input file (see **iscf** keyword). Their respective efficiencies are compared in Fig. 4 where the bcc iron is treated, taken from Ref. [10]. As it can be seen from this figure, final efficiencies are comparable for both schemes. However we have observed on several systems that the density mixing scheme was sometimes more efficient than the potential mixing. With regard to the behaviour of total energy during the self-consistent cycle, it is remarkably different: when potential mixing is activated, all parts of the total energy are computed at the same time; the total energy is thus variational with respect to the self-consistent cycle step. When density mixing is activated, parts of
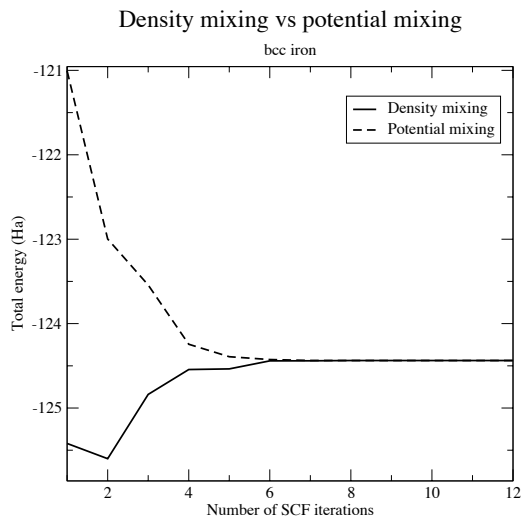
Fig. 4. Comparison between "potential mixing" and "density mixing" for PAW. Example of bcc iron [10]

total energy are computed at various stages of the cycle which results in a behaviour of total energy that is not variational.

All mixing and preconditioning schemes implemented in ABINIT — even most sophisticated ones — are compatible with the PAW formalism.

PAW datasets
In order to perform PAW calculations, ABINIT needs for each atomic species a so-called *PAW dataset* file containing all constant data for the atom: definition of the electronic configuration, partial waves basis and associated projectors, all "frozen" atomic data, etc. All the needed PAW dataset files, also improperly called *PAW pseudopotential* files, have to be set in *files* file exactly like norm-conserving pseudopotential files.

Use of PAW datasets gives the possibility to control not only approximations (frozen core approximation, completeness of partial waves basis, ...) but also

24

transferability and efficiency of atomic data (choosing radius of augmentation regions or pseudization schemes). For these reasons, even if pre-build PAW datasets are provided on the ABINIT web site, several tools to produce specific PAW datasets are included in the package. They are provided under the GNU-GPL license. A specific tutorial (*tutorial PAW2*) helps inexperienced users to become familiar with them.

From the ABINIT package, production of new PAW datasets is possible using two different utilities (PAW datasets generators):

- `Atompaw2Abinit` is a convertor generating PAW datasets for ABINIT from datasets produced by the ATOMPAW code [11,12]. Initially written by N. Holzwarth, ATOMPAW has been modified in order to produce data compatible with ABINIT requirements. Several improvements have also been added such as efficient schemes to build the partial wave basis, logarithmic radial grids and relativistic resolution of atomic problem.

- `Uspp2Abinit` is a plugin for the USPP code [13]. USPP, originally written by D. Vanderbilt to produce ultrasoft pseudopotentials [14], can be modified in order to produce PAW datasets directly usable by ABINIT. USPP provides a lot of interesting features and produces efficient partial wave basis that can be easily converted into PAW datasets using Uspp2Abinit.

Guide for use
In a norm-conserving pseudopotential calculation, the (plane wave) density grid is (at least) twice bigger than the wavefunctions grid, in each direction. In a PAW calculation, the (plane wave) density grid is tunable thanks to the input variable **pawecutdg** (PAW: ECUT for Double Grid). This is needed because of the mapping of quantities (e.g. densities, potentials) located in the augmentation regions (PAW spheres) onto the global FFT grid. The number of points of the Fourier grid located in the spheres must be high enough to preserve the precision. It is determined from the cut-off energy **pawecutdg**. One of the most sensitive quantity affected by this grid transfer is the compensation charge density; its integral over the augmentation regions (on spherical grids) must cancel with its integral over the whole simulation cell (on the FFT grid). Although convergence with respect to the parameter **pawecutdg** should always be checked, in practice, a common use it to put it at least twice larger than **ecut** and keep it constant during all calculations. Increasing **pawecutdg** slightly changes the CPU execution time, but above all it is highly memory-consuming. Note that, if **ecut** is already large, there is no need for a large **pawecutdg**. When testing **ecut** convergence, **pawecutdg** has to remain constant to obtain consistent results.

Comparing with a norm-conserving calculation, there are two differences to execute a PAW calculation:

- the keyword **pawecutdg** must be added in the input file. There is no default value for this variable as it governs the convergence of the FFT grid for densities.
- a PAW atomic data set must be used in place of the pseudopotential file. ABINIT automatically detects from the atomic file that the calculation is a PAW one.

There are other input keywords specific to PAW (see *"PAW variables"* in ABINIT documentation). They are to be used when tuning the computation, in order to gain accuracy or save CPU time. In a standard computation, these variables should not be modified.

Automatic tests and tutorials
PAW features in ABINIT are included in the set of automatic tests in the `tests/v4/`, `test/v5/` and `tests/paral/` directories.

In the `tests/v4/` directory, almost all the main ground-state functionalities — not specific to PAW — are tested. Energy, forces, stresses for several system types can be checked using `tests/v4/t04.in` to `tests/v4/t06.in` input files. Self-consistent cycle and different mixing schemes tests (either on potential or on density) are provided in `tests/v4/t07.in`. Other specific uses within PAW can be checked using `test/v4/t08.in`, `tests/v4/t09.in` and `tests/v4/t94.in` input files.

In the `tests/v5/` directory, all PAW specific functionalities are tested. They are described in the following sections. It can nevertheless be useful to cite that `tests/v5/t16.in` input file provides run examples for magnetic systems (ferromagnetic, antiferromagnetic and non-collinear), while `tests/t5/v17.in` tests the spin-orbit coupling implementation. Finally, the `tests/paral/` directory contains several input files referring to PAW capabilities in the framework of various parallelism schemes.

In addition, new users may be helped in using ABINIT PAW features by two tutorials. Tutorial PAW1 is related to the general use of ABINIT in the framework of the PAW method, while tutorial PAW2 is specifically devoted to the generation of PAW datasets.

On-going developments
The development of the PAW functionalities in ABINIT is now evolving in two directions, a coupling of PAW+LDA framework with the Dynamical Mean-Field Theory [15], and the implementation of the Density-Functional Perturbation Theory within the PAW formalism [16,17]

Density-functional theory, in the usual LDA or GGA approximations, does not describe correctly the interaction between electrons, and contains moreover a spurious self-interaction. As a consequence, it is not able to describe strongly correlated systems which contains partially filled d or f shells. A way to correct this failure is to take explicitly into account the interaction between d or f electrons through a Hubbard-like term in the Hamiltonian. In the static mean field approximation, this correction leads to the LDA+U [18–20] method, which depends on two parameters: the coulomb interaction U and the exchange J.

In this framework, the contribution to energy is the sum of the LDA energy for a given density, the electron-electron interaction term E_{ee} from the Hubbard term and a double counting term $-E_{dc}$: $E_{LDA+U} = E_{LDA} + E_{ee} - E_{dc}$.

- In the implementation, E_{ee} is used in the rotationally invariant form of Ref. [20]. As a consequence, the LDA+U energy is a function of an occupation matrix $n_{m,m'}^{\sigma}$ expressed in the correlated orbitals basis.
- Two implementations of the double counting correction are provided. In the "Full Localized Limit" (FLL) [18,20,21], activated if **usepawu**=1, E_{dc} corresponds to the value of E_{ee} in a reference system in which the occupation matrix is diagonal and diagonal elements are whole numbers. The "Around Mean Field" (AMF) version, in contrast, is designed to correct a system in which electrons are equally shared among correlated orbitals [21]. It is used if **usepawu**=2.

Other input variables are mandatory to use LDA+U: For each species, **lpawu** defines the angular momentum on which the LDA+U correction is applied (if **lpawu**=-1,no correction is applied). **upawu** and **jpawu** give the corresponding values for U and J. The implementation is described in some details and tests on well-known systems are given in Ref. [22].

Following the work of Bengone *et al* [23], the occupation matrix is defined only inside the PAW atomic sphere. Special care must be devoted to the choice of atomic data, which must have sufficiently large PAW radii. Two different expressions of the occupation matrix can be used according to the keyword **dmatpuopt** (see Ref. [22] for the explicit formulation).

The most used version of LDA+U, namely the Full localized limit (**usepawu**=1) minimizes the interaction energy if the correlated orbitals are either empty or full, whereas in LDA, electrons are equally shared among correlated d or f orbitals. This explains why LDA+U opens a gap inside correlated orbitals and is thus able to describe Mott insulators. However, as a consequence, it is possible to have a local minimum of the energy for different filling of the correlated

orbitals. In this case, it can be hard to find the absolute minimum. A way to solve this problem is to impose the starting occupation matrix, during a given number of steps (indicated by the input variable **usedmatpu**), in order to study different metastable states [22,24]. The value of the starting occupation matrix (for d-elements, a 5×5 matrix for each spin), is given by the input variable **dmatpawu**. In order to have more insight on the occupations of orbitals, one can diagonalise the occupation matrix thanks to **dmatudiag**=1.

4 automatic tests are given with the code (for NiO: see the example input files `tests/v5/t08.in,tests/v5/t11.in` and `tests/v5/t20.in` and Gd:`tests/v5/t19.in`). The calculations using an imposed density matrix are in particular tested for d and f elements. The collinear and non-collinear implementations are tested. A tutorial, `doc/tutorial/lesson_ldau.html` is available and explains how to do a LDA+U calculation for nickel oxide.

3.3   The local PBE0 approximation

Recently, Novák *et al.* proposed to apply the exact exchange functional to a restricted subspace formed by the correlated electrons of a correlated system and called this method "exact exchange for correlated electrons" (EECE) [25]. This approach is used in the Projector Augmented-Waves (PAW) framework: the PBE0 [26] exchange-correlation hybrid form functional is implemented only inside the PAW atomic spheres for correlated orbitals.
In this frame the exchange-correlation energy is:

$$E_{\mathrm{xc}}^{\mathrm{PBE0}}[\rho] = E_{\mathrm{xc}}^{\mathrm{PBE}}[\rho] + \frac{1}{4}(E_{\mathrm{x}}^{\mathrm{HF}}[\Psi_{sel}] - E_{\mathrm{x}}^{\mathrm{PBE}}[\rho_{sel}]) \tag{3}$$

where PBE refers to the Perdew-Burke-Ernzerhof exchange-correlation functional [27]. Ψ_{sel} and ρ_{sel} represent the wavefunction and the corresponding electron density of the selected electrons respectively [28]. The selected electrons are the correlated electrons, for instance the d electrons of Ni in NiO. The exchange Hartree-Fock (HF) term is:

$$E_{\mathrm{x}}^{\mathrm{HF}}[\Psi_{sel}] = -\frac{1}{2}\sum_{nn'}^{occ} \delta_{\sigma_n,\sigma_{n'}} \int d\mathbf{r} d\mathbf{r}' \frac{\Psi_n(\mathbf{r})\Psi_{n'}^*(\mathbf{r})\Psi_n^*(\mathbf{r}')\Psi_{n'}(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} \tag{4}$$

where n and n' range over occupied correlated states and σ_n and $\sigma_{n'}$ are the associated spins. As we are interested here only in localized correlated states, we make the assumption that the correlated orbitals are zero outside the PAW sphere. Putting Eq. 2 into Eq. 4 leads, after some calculations, to:

$$E_{\mathrm{x}}^{\mathrm{HF}}[\Psi_{sel}] = -\frac{1}{2} \sum_{LM} \frac{4\pi}{2L+1} \sum_{d_i d_j d_k d_l} F^L_{d_i d_j d_k d_l}$$

$$\times \sum_{\mathrm{m_i m_j m_k m_l}} \langle m_i | LM | m_j \rangle \langle m_k | LM | m_l \rangle \sum_\sigma \rho^\sigma_{ik} \rho^\sigma_{jl}, \tag{5}$$

where $\langle m_i | LM | m_j \rangle$ are real Gaunt coefficients calculated for the selected l momentum, $F^L_{d_i d_j d_k d_l}$ the Slater integrals, and $\rho^\sigma_{ij} = \sum_{nk} f_{nk} \langle \tilde{\Psi}^\sigma_{nk} | \tilde{p}_i \rangle \langle \tilde{p}_{\mathrm{j}} | \tilde{\Psi}^\sigma_{\mathrm{nk}} \rangle$; index i (resp. j) in ρ^σ_{ij} stand for (d_i, m_i) (resp. (d_j, m_j)), where $d_i = (l_i, n_i)$ for a selected (d) orbital.

This result is equivalent to the formulation already established in Ref. [29] for the all electron one-center part of the exchange energy. The Slater integrals and the real Gaunt coefficients are calculated once and for all. From the knowledge of the occupation matrix ρ^σ_{ij}, the energy is calculated directly from Eq.(5) and from the quantity $E_{\mathrm{x}}^{\mathrm{PBE}}[\rho_{sel}]$. The corresponding KS potential is deduced by derivation of this energy, which leads to a supplementary non-local term in the hamiltonian.

To use the local PBE0 approximation in ABINIT, two keywords must be activated:

- **useexexch** must be put to 1 to allow the use of the exact exchange in the spheres (default value is 0).

- **lexexch** controls the angular momentum selected for the correlated orbitals (2 for d orbitals; 3 for f orbitals).

It is also possible to change the mixing factor of exact exchange (1/4 by default) thanks to the keyword **exchmix**. A test is provided in the ABINIT package (see the example input file `tests/v5/t18.in`).

3.4   Berry phase calculation of polarization within PAW

The polarization of a finite piece of matter of volume V can be computed as the dipole moment per unit volume:

$$\mathbf{P} = \frac{e}{V} \left[\sum_\kappa Z_\kappa \mathbf{R}_\kappa - \int_V d\mathbf{r}\ \mathbf{r}\ n(\mathbf{r}) \right], \tag{6}$$

where Z_κ and $\mathbf{R}_\kappa$ are the atomic number and position of the κth atom, e is the absolute value of the electronic charge, and $n(\mathbf{r})$ is the electronic charge density. Although such a dipole moment is in principle well defined, $\mathbf{P}$ is not a bulk property, being dependent upon truncation and shape of the sample. However, the variations of $\mathbf{P}$ are indeed measured as bulk material properties in several circumstances. Thus, the calculation of the polarization under periodic boundary conditions is a nontrivial task, and this problem was solved by Resta, Vanderbilt, and King-Smith in Refs. [30–32] building the modern theory of polarization. The theory in the framework of the norm-conserving

and ultrasoft pseudopotentials formalism is respectively given in Refs. [32] and [33]. In addition, the formalisms of the Berry phase and maximally localized Wannier functions (MLWFs) are closely linked. The case of the generation of Wannier functions for ultrasoft pseudopotentials is detailed by Ferretti *et al* [34]. In this subsection, we only detail peculiarities of our PAW implementation.

The macroscopic polarization is the sum of an ionic part, $\mathbf{P}_{ion}$, and an electronic part, $\mathbf{P}_{elec}$. Within the PAW formalism, the last one is decomposed into two terms: the Berry Phase term (BP) and the expectation value (EV) term [33]:

$$\mathbf{P} = \mathbf{P}_{ion} + \mathbf{P}_{elec} = \mathbf{P}_{ion} + \mathbf{P}_{\text{BP}} + \mathbf{P}_{\text{EV}}. \tag{7}$$

The EV term, described in Ref. [33], can be written as:

$$\mathbf{P}_{\text{EV}} = -\frac{2e}{V} \sum_n \sum_{\mathbf{k}} \sum_{i,j} \mathbf{d}_{ij} \langle \tilde{\Psi}_{n\mathbf{k}} | \tilde{p}_i \rangle \langle \tilde{p}_j | \tilde{\Psi}_{n\mathbf{k}} \rangle, \tag{8}$$

where $\mathbf{d}_{ij} = \int \mathbf{r} Q_{ij}(\mathbf{r}) d\mathbf{r}$ is the dipole moment of the augmentation charge $Q_{ij}(\mathbf{r}) = (\Phi_i(\mathbf{r})\Phi_j(\mathbf{r}) - \tilde{\Phi}_i(\mathbf{r})\tilde{\Phi}_j(\mathbf{r}))$.

The BP term can be expressed as a function of the overlap matrix, $M_{mn}^{\mathbf{k},\mathbf{b}}$, used in the computation of MLWFs [33]. We can generalize the equation for $M_{mn}^{\mathbf{k},\mathbf{b}}$ in Ref. [34] to the PAW case:

$$M_{mn}^{\mathbf{k},\mathbf{b}} = \langle \tilde{\Psi}_{m\mathbf{k}} | e^{-i\mathbf{b}\mathbf{r}} | \tilde{\Psi}_{n\mathbf{k}+\mathbf{b}} \rangle + \sum_{i,j} Q_{ij}(\mathbf{b}) \langle \tilde{\Psi}_{m\mathbf{k}} | \tilde{p}_i \rangle \langle \tilde{p}_j | \tilde{\Psi}_{n\mathbf{k}+\mathbf{b}} \rangle, \tag{9}$$

where

$$Q_{ij}(\mathbf{b}) = \int d\mathbf{r} \; Q_{ij}(\mathbf{r}) \; e^{-i\mathbf{b}\cdot\mathbf{r}} = 4\pi \; e^{-i\mathbf{b}\cdot\mathbf{R}_\kappa} \sum_{L,M} \mathcal{G}_{l_i m_i l_j m_j}^{LM} \; (-i)^L S_{LM}(\hat{b})$$

$$\times \int dr \; r^2 \; j_L(br) \; (\phi_{l_i n_i}(r)\phi_{l_j n_j}(r) - \tilde{\phi}_{l_i n_i}(r)\tilde{\phi}_{l_j n_j}(r)). \tag{10}$$

In this equation, S_{LM} are real spherical harmonics, $\mathcal{G}$ are Gaunt coefficients, $\phi_{l_i,n_i}(r)$ is the radial part of $\Phi_i(\mathbf{r})$, and the $\mathbf{b}$-vectors connect nearest-neighbour $\mathbf{k}$-points. The equation for $M_{mn}^{\mathbf{k},\mathbf{b}}$ differs from the one obtained by Vanderbilt *et al* [33], but they have the same limit as the number of $\mathbf{k}$-points increases [34].

In practice, Eqs. (8) and (9) have been implemented inside ABINIT in the berryphase_new.F90 subroutine. The keyword **berryopt**=-1 activates the Berry phase calculation and the keyword **rfdir** must be used to specify the primitive vector along which the projection of the polarization will be computed. The computed polarization is then given both in reciprocal and cartesian coordinates. Presently, the Berry phase calculation is compatible with spin polar-

ization (**nsppol**=2) but it is not compatible with spinors (**nspinor**=2). Two automatic tests, `tests/v5/t12.in` and `tests/v5/t13.in`, are available.

3.5 Electronic transport properties in PAW

The transport properties such as electrical and thermal conductivities, index of refraction, or absorption can be obtained using norm-conserving or PAW potentials. In both cases, this requires a ground state calculation performed using the ABINIT package and the use of the `conducti` utility as explained in `doc/users/conducti_manual` and `doc/users/conducti_manual_paw`.

Neglecting the phonon contribution, the transport properties of condensed matter systems such as solids or dense plasmas are calculated for a fixed ionic configuration obtained from crystallographic structure information or molecular dynamics trajectories. For a simulation cell containing N atoms, the electronic wavefunction for the N_e active electrons obtained within the KS framework leads to n_b orbitals $\psi_{i,\mathbf{k}}$ with corresponding energies $\epsilon_{i,\mathbf{k}}$ for a given $\mathbf{k}$-point. The electrical and optical properties are obtained using the Kubo-Greenwood formulation which leads to the real part of the electrical conductivity [35–37], $\sigma_1(\mathbf{k}, \omega)$:

$$\sigma_1(\mathbf{k}, \omega) = \frac{2\pi}{3\omega\Omega} \sum_{j=1}^{n_b} \sum_{i=1}^{n_b} \sum_{\alpha=1}^{3} (F(\epsilon_{i,\mathbf{k}}) - F(\epsilon_{j,\mathbf{k}})) \tag{11}$$
$$\times |\langle\psi_{j,\mathbf{k}}|\nabla_\alpha|\psi_{i,\mathbf{k}}\rangle|^2 \delta(\epsilon_{j,\mathbf{k}} - \epsilon_{i,\mathbf{k}} - \omega).$$

We employ atomic units, with the electron charge e, Planck's constant $\hbar$, and the electron mass m_e all set to unity. The i and j summations range over the n_b discrete bands (orbitals) included in the triply periodic calculation for the cubic super-cell volume element Ω. The α sum is over the three spatial directions. $F(\epsilon_{i,\mathbf{k}})$ is the Fermi weight corresponding to the energy $\epsilon_{i,\mathbf{k}}$ for the ith band for the $\mathbf{k}$ point $\mathbf{k}$. The total optical conductivity is obtained by direct summation over all necessary $\mathbf{k}$-points. Using the Kramers-Kronig relations, the calculation of real part of the optical conductivity leads to the dielectric functions, $\epsilon(\omega) = \epsilon_1(\omega) + i\epsilon_2(\omega)$, complex index, $\mathbf{n}(\omega) = n(\omega) + ik(\omega)$, the absorption coefficient, $\alpha(\omega)$ and the reflectivity, $r(\omega)$ [38].

Within the PAW formalism, the matrix elements become

$$\langle\psi_{m,\mathbf{k}}|\vec{\nabla}|\psi_{n,\mathbf{k}}\rangle = \langle\tilde{\psi}_{m,\mathbf{k}}|\vec{\nabla}|\tilde{\psi}_{n,\mathbf{k}}\rangle$$
$$+ \sum_{i,j}\langle\tilde{\psi}_{m,\mathbf{k}}|\tilde{p}_i\rangle\langle\tilde{p}_j|\tilde{\psi}_{n,\mathbf{k}}\rangle \left(\langle\phi_i|\vec{\nabla}|\phi_j\rangle - \langle\tilde{\phi}_i|\vec{\nabla}|\tilde{\phi}_j\rangle\right) \tag{12}$$

where the first term is the plane wave contribution and the last two terms are the PAW corrections obtained in spherical coordinates using the usual separation $\phi(\mathbf{r}) = \frac{u_{n,l}(r)}{r}S_{l,m}(\hat{r})$ where $S_{l,m}(\hat{r})$ are real spherical harmonics. Details of the implementation can be found in Ref. [39]. Activation of the option **prtnabla**=1 during a ground state calculation produces these matrix elements in the file *filename.OPT*. Sample, input files for a complete calculations of the transport properties can be found in `tests/v5/t10.in` and `tests/v5/t11.in`.

In the X-ray regime, the absorption is directly proportional to the optical conductivity given by Eq.(12) but with the summation on the indices n and m involving core-electron states. To extend the calculation of the optical conductivity to include core electrons, we further evaluate the following matrix elements

$$\langle\psi_{m,\mathbf{k}}|\vec{\nabla}|\phi_c\rangle = \langle\tilde{\psi}_{m,\mathbf{k}}|\vec{\nabla}|\phi_c\rangle + \sum_i \langle\tilde{\psi}_{m,\mathbf{k}}|\tilde{p}_i\rangle\left(\langle\phi_i|\vec{\nabla}|\phi_c\rangle - \langle\tilde{\phi}_i|\vec{\nabla}|\phi_c\rangle\right). \quad (13)$$

Following M. Taillefumier *et al.* [40], we evaluate matrix elements for core electrons whose wavefunction vanishes outside the PAW sphere of the absorbing atom. In this case, and assuming that the pseudo partial wave $|\tilde{\phi}_{\mathbf{R},n}\rangle$ form a complete basis for the pseudo-wavefunction $|\tilde{\psi}_{i,\mathbf{k}}\rangle$ within the PAW sphere, the matrix elements reduce to

$$\langle\psi_{m,\mathbf{k}}|\vec{\nabla}|\phi_c\rangle = \sum_i \langle\tilde{\psi}_{m,\mathbf{k}}|\tilde{p}_i\rangle\langle\phi_i|\vec{\nabla}|\phi_c\rangle. \quad (14)$$

These matrix elements used in Eq.(12) allows to calculates the Xray absorption cross section, $\sigma^X(\omega)$, in a single particle approach. This directly allows to obtain the XANES spectra near an edge where the corresponding wavefunction vanishes outside the PAW sphere in the impurity model [40,41]. Activation of the option **prtnabla**=2 during a ground state calculation produces the file *filename.OPT2* containing the matrix elements involving core electrons. This calculation requires the core wavefunctions and energies saved during the pseudo-potential generation using a modified version of the *atompaw* generator. During the ABINIT run, the orbitals and their energies are read from the file *corewf.dat*.

3.6 Properties at the Nuclei: Electric Field Gradients and Mössbauer Isomer Shifts in the PAW Formalism

Because the PAW formalism provides a robust way to reconstruct the all-electron wavefunctions in the valence space, it is suitable for computing expectation values of observables localized even very close to the nuclei. Obtaining equivalent accuracy within the norm-conserving pseudopotential frame-

work would require very small atomic radii for the pseudization procedure, and concomitantly high planewave cutoff energies and lengthy calculations. There remains the question of whether even all-electron accuracy in the valence space is sufficient for accurate representation of observables close to the nuclei, where conventional wisdom would suggest that deep core polarizations might be quite significant for properties such as the electric field gradient or Fermi contact interaction. Such concerns turn out to be unwarranted, however, as our experience and others have shown that the PAW formalism together with a typical chemical valence/core separation are sufficient for accurate description of nuclear point properties such as the electric field gradient [42–44], Fermi contact interaction [45] and magnetic chemical shielding [46].

Both the electric field gradient and Fermi contact interaction are ground-state observables, and their computation adds negligible time to a normal self-consistent ground state calculation. The electric field gradient is computed from the electrostatic potential as

$$V_{\alpha\beta} = \left[\frac{\partial^2}{\partial x_\alpha \partial x_\beta} \int \frac{n(\mathbf{r})}{\mathbf{r}} d\mathbf{r} \right]\bigg|_{\mathbf{R}}, \tag{15}$$

where $n(\mathbf{r})$ is the total charge density and $\mathbf{R}$ is the location of the nucleus of interest. The total charge density in the PAW formalism contains the pseudovalence density, the nuclear ionic charges, and the all-electron and pseudo charge densities within the PAW spheres. As done in earlier work, the electric field gradient due to the pseudovalence density is computed in reciprocal space, and the gradient due to the (fixed) ionic charges is computed with an Ewald sum approach. The PAW sphere charge densities contribute matrix elements of $(3x_\alpha x_\beta - r^2 \delta_{\alpha\beta})/r^5$, weighted by the charge densities in each channel determined by the self-consistent minimization procedure. This treatment [44] is more flexible than for example assuming all bands are doubly occupied, and permits the current approach to be used with more complex electronic and magnetic states than just insulators.

Within ABINIT, the electric field gradient computation is invoked with the key word **prtefg** (for Print EFG), together with the key word **quadmom**. The **prtefg** key word takes the values 1–3. For value 1, the electric field gradient coupling in MHz is reported, where the conversion is made by atom by combining the gradient with the nuclear quadrupole moments supplied by **quadmom**. When **prtefg** is input as 2, the additional breakdown of the field gradient in terms of valence and on-site PAW terms is reported, along with the eigenvectors of the EFG matrix, so that principal directions may be determined with respect to the crystal axes. Finally, **prtefg**=3 allows additional computation of a point-charge model of the gradient, for comparison purposes. The point charges by atom are supplied through the additional variable **ptcharge**. Detailed examples of the use of ABINIT to compute EFG's

can be found in Refs. [44,47] and tests `tests/v5/35` and `tests/v5/51-53`.

The Fermi contact interaction, which is just the electron density evaluated precisely at the nuclear location, is an important component of the isomer shift measured in Mössbauer spectroscopy. The isomer shift is directly proportional to $n_{\mathrm{abs}}(\mathrm{R}) - n_{\mathrm{src}}(\mathrm{R})$, the difference in electron density at the absorber (sample) and the source. Evaluating the density at a nuclear position can be accomplished by treating $\delta(\mathbf{r} - \mathbf{R})$ as the observable, that is, the three-dimensional Dirac delta function centered on the nuclear position $\mathbf{R}$. Because of the short-range nature of the delta function, in the PAW-transformed version of the observable only matrix elements of the on-site all-electron valence functions are required [45], and these are evaluated from a linear fit to the first few points of the PAW radial functions.

Within ABINIT the Fermi contact interaction is invoked with the key word **prtfc** (for Print Fermi Contact, see also `tests/v5/t34,36`, which only takes the value 1. When called, the electron density at each nuclear position is reported, in atomic units (electrons per cubic Bohr). The isomer shift as measured in Mössbauer spectroscopy is typically reported in velocity units and is obtained from the formula

$$\delta = \frac{c}{E_\gamma} \frac{2\pi Z e^2}{3} \left[n_{\mathrm{abs}}(\mathrm{R}) - n_{\mathrm{src}}(\mathrm{R}) \right] \Delta \langle r^2 \rangle, \tag{16}$$

where c is the speed of light, E_γ the γ-ray energy, Z the atomic number, e the electron charge, and $\Delta \langle r^2 \rangle$ the change in the mean square nuclear radius for the transition. The electronic densities n_{abs} and n_{src} refer to the absorber and source respectively. Because of the linearity of this formula in the density at the absorber (sample) nucleus, the only unknown ($\Delta \langle r^2 \rangle$) can be obtained by comparing the calculated values in several standards to experiment and then the computations can be used to interpret the measurements of new materials. In Ref. [45] we showed how to perform such studies on a variety of compounds, including a number of ^{67}Zn cases where it is also necessary to correct for the second-order Doppler shift, and also the case of ZnO under pressure.

4 Other developments in ground-state calculations

4.1 Positron lifetime

In this section we describe the implementation of the positron lifetime calculation within the two-component density-functional theory [48–50] (TCDFT). This theory allows for the determination of the ground state electron charge and positron charge densities (respectively n_- and n_+) associated to a system of interacting electrons and positrons moving in an external potential V_{ext}. The TCDFT states that the total energy of such a system is given by:

$$E[n_+, n_-] = F[n_+] + F[n_-] \tag{17}$$

$$+ \int d\mathbf{r} V_{ext}(\mathbf{r})[n_-(\mathbf{r}) - n_+(\mathbf{r})] \tag{18}$$

$$+ \int d\mathbf{r} \int d\mathbf{r}' \frac{n_-(\mathbf{r})n_+(\mathbf{r})}{|\mathbf{r} - \mathbf{r}'|} + E_c^{e-p}[n_+, n_-] \tag{19}$$

where $F[n]$ denotes the one-component DFT functional for electrons (or positrons) and $E_c^{e-p}[n_+, n_-]$ is the electron-positron correlation-energy functional. In the following we will focus on the case of a single positron in a solid. That implies that no positron self-interaction is considered. Moreover, among the various parametrizations of the electron-positron correlation energy available in the literature, we have chosen to apply the so-called conventional scheme in which one considers that (i) the positron density does not affect the DFT electron density (the positron and its screening electron cloud form a neutral quasiparticle entering the system) and (ii) the positron state and annihilation characteristics are calculated in the LDA (Local Density Approximation) and the zero-positron-density limit of the electron-positron correlation functional. Although this approach is only correct in the case of a delocalized positron in a solid, it has been shown that due to cancellation effects it works also pretty well for a trapped positron at a vacancy-type defect [49,51,52]. Within the conventional scheme, a rather simple procedure may be used to compute the positron lifetime. First, we compute the ground state electronic density as well as the associated Hartree-Coulomb potential $(v_H[n_-(\mathbf{r})] = \int d\mathbf{r}' \frac{n_-(\mathbf{r})n_+(\mathbf{r})}{|\mathbf{r}-\mathbf{r}'|})$ within the common DFT framework. Then we construct the potential felt by the positron as $v^+(\mathbf{r}) = -v_H[n_-(\mathbf{r})] + v_c^{e-p}[n_-(\mathbf{r})] + v_n[\{Z_I, \mathbf{R}_I\}]$, and we solve the following KS equation [1]:

$$-\frac{1}{2}\nabla^2 \psi_i^+(\mathbf{r}) + v^+(\mathbf{r})\psi_i^+(\mathbf{r}) = \epsilon_i^+ \psi_i^+(\mathbf{r}), \tag{20}$$

---

[1] Note that the thermalized positron in a lattice is in the $\mathbf{k}$=0 state. When studying ideal bulk systems we thus calculate the positron wavefunction at the Γ point only.

where $v_n[\{Z_I, \mathbf{R}_I\}]$ stands for the interaction between the nuclei (charge Z_I and positions $\mathbf{R}_I$) and the positron, ψ_i^+ and ϵ_i^+ are respectively the positron wavefunctions and eigenvalues. The wavefunction $\psi^+(\mathbf{r})$ that corresponds to the lowest energy of ϵ^+ yields the positron charge density $n_+(\mathbf{r}) = |\psi^+(\mathbf{r})|^2$. Note that the Coulomb potential arising from the electrons is the same as the one sensed by the electrons except for a change of sign. At this stage, we know both the ground state electrons charge and positron charge densities. The positron lifetime τ if then defined as the inverse of the annihilation rate λ :

$$\tau = 1/\lambda = (\pi r_e^2 c \int d\mathbf{r}\, n_+(\mathbf{r}) n_-(\mathbf{r}) \Gamma[n_-(\mathbf{r})])^{-1} \qquad (21)$$

where r_e is the classical electron radius and c is the speed of light. The term Γ is called the enhancement factor and is introduced in order to take into account the increase in the annihilation due to the screening cloud of electrons around the positron. We adopted the correlation potential and enhancement factor parametrized by Boroński and Nieminen [49] from the results of Arponen and Pajanne [53].

In order to compute a positron lifetime with ABINIT, one has to chain two calculations. The first is the ground state calculation from which we output the Hartree electrons potential by setting **prtvha** to 1. In the second calculation, the positron ground-state as well as the annihilation rate are computed by introducing the two keywords **positron**=1 and **ixcpositron**=1. The first one implements the KS equation of the positron (Eq. 20) and the second one chooses the parametrization of the electrons-positron correlation potential. Note that the first dataset can be performed either within the norm-conserving pseudopotential framework or within the PAW one assuming that $\tilde{n}_- + \hat{n}_-$ is a sufficiently good approximation of the all-electron (AE) density n_- (see Makkonen *et al.* [54]). Obviously this latter assumption does not allow to benefit from the AE accuracy of the PAW method. For both of the two cases presented above, we have to construct a pseudopotential describing the interaction between the ions (the nuclei plus the core electrons) and the positron. In this work we choose to use a purely local form for this potential $v_H[n_c + Z_I]$ which is computed from an electronic core-density obtained with the AE atomic code FHI.

Within the ABINIT package, we provide an automatic test on bulk aluminum (`tests/v4/t95.in` followed by `tests/v4/t96.in`)at the experimental volume ($V_0 = 16.61\text{Å}^3$). The calculated positron lifetime is 163 ps which is in good agreement with all the theoretical values available in the literature as well as with experiments (the commonly accepted experimental value is 163 ps [55]). Indeed, LMTO-ASA calculations are in the range 163-165 ps [56–58], while the more accurate FPLAPW method leads to a value of 166 ps [59].

The ABINIT code integrates also a DFT method based on Daubechies wavelets [60] basis set. They form a systematic orthogonal and smooth basis that is localized both in real and Fourier space and that allows for adaptivity. Such method was developed as an electronic structure code *per se*, in the framework of a three years european STREP project, named BigDFT. Via the use of the input variable **usewvl**, the KS SCF treatment in ABINIT can be done with the routines of the BigDFT code. The features of this code were described in its reference paper [61], and we will briefly summarize here the principal points.

There are two fundamental functions in Daubechies family, the scaling function and the wavelet. The full basis set can be obtained from all translations by a certain grid spacing h of the scaling and wavelet functions centered at the origin. Both functions are localized, with compact support. Every scaling function and wavelet on a coarse grid of spacing h can be expressed as a linear combination of scaling functions at the finer grid level $h/2$. For this reason, wavelet functions complete the information which is lacking for refining the resolution level. The implementation of the BigDFT code uses only two resolution levels, namely one level of adaptivity. We can thus classify the coarse and the fine degrees of freedom by the information expanded by the scaling and the wavelet functions respectively.

For a three-dimensional description, the simplest basis set is obtained by a tensor product of one dimensional basis functions. In each grid point the coarse degrees of freedom are expanded by a single three dimensional function $\phi^0_{i_1,i_2,i_3}(\mathbf{r})$, which corresponds to the tensor product of three scaling functions, while the fine degrees of freedom can be expressed by adding other seven basis functions, $\phi^\nu_{j_1,j_2,j_3}(\mathbf{r})$, which include tensor products with one-dimensional wavelet functions. A wavefunction $\Psi(\mathbf{r})$ can thus be expanded in this basis:

$$\Psi(\mathbf{r}) = \sum_{i_1,i_2,i_3} c^0_{i_1,i_2,i_3} \phi^0_{i_1,i_2,i_3}(\mathbf{r}) + \sum_{j_1,j_2,j_3} \sum_{\nu=1}^{7} c^\nu_{j_1,j_2,j_3} \phi^\nu_{j_1,j_2,j_3}(\mathbf{r}) \qquad (22)$$

The sum over i_1, i_2, i_3 runs over all the grid points contained in the low resolution region and the sum over j_1, j_2, j_3 over all the points contained in the (generally smaller) high resolution region. Such points belong to a uniform mesh of grid spacing h. Each wavefunction is then associated to a set of coefficients $\{c^\mu_{j_1,j_2,j_3}\}$, $\mu = 0, \cdots, 7$. The fundamental input parameters, which define the mesh spacing of the simulation box and the extension of the low and high resolution regions are **wvl_hgrid**, **wvl_crmult** and **wvl_frmult** respectively. The wavefunctions are stored in a compressed form where only the nonzero scaling function and wavelets coefficients are stored. The basis set

being orthogonal, several operations such as scalar products among different orbitals and between orbitals and the projectors of the non-local pseudopotential can directly be done in this compressed form.

Wavelet properties allow to express analytically (within machine precision) the matrix elements of the kinetic operator as well as the nonlocal part of the pseudopotentials. The BigDFT code supports the GTH-HGH [62–64] norm-conserving pseudopotentials which, thanks to their analytic expression, allow an accurate expansion in Daubechies basis.

The real space expression of the KS wavefunctions, needed for expressing the charge density and for applying the local potential, is given in the Deslauriers-Dubuc interpolating scaling functions [65]. The latter is a basis set which is intimately connected with the Daubechies scaling functions, and which is optimal for expressing functions in real space. Indeed, the expansion coefficients of a given function in the interpolating scaling functions basis coincide exactly with the values of that function on the mesh points. Working in this basis eases the treatment of the local potential, including the XC term and the Hartree potential. The passage between Daubechies and Deslauriers-Dubuc basis can be expressed efficiently without loss of precision. This can be done via the so-called "magic filter" transformation [66], which allows to express with better accuracy the point values of the wavefunctions.

Thanks to these properties, the absolute precision of the results is only driven by the basis completeness, and we achieve a convergence rate of $\mathcal{O}(h^{14})$. All computational operations of the BigDFT code, except for the linear algebra routines and the Poisson solver, can be written in terms of convolutions with *short*, *separable* filters. Such kind of operation can be efficiently optimised, which results in a code with very good performances.

After the application of the hamiltonian, the KS wavefunctions are updated via a direct minimization scheme [67], which in its actual implementation is fast and reliable only for insulators. Tough at present the iterative diagonalization scheme used in the SCF cycle of ABINIT cannot be used with a wavelet run, we see no reason why such formalism cannot be extended to metallic systems.

Also the parallelization scheme used in a wavelet run of ABINIT is rather different than the one which is used conventionally. Two data distribution schemes are used. In the orbital distribution scheme, used for the application of the hamiltonian and the preconditioning, each processor is working on one or a few orbitals for which it holds all its scaling function and wavelet coefficients. The wavefunction orthogonalization as well as the calculation of the Lagrange multipliers is done in the coefficient distribution scheme, where each processor holds a certain subset of the coefficients of all the orbitals. Switch back and forth between the orbital distribution scheme and the coefficient distribution

scheme is done by the MPI global transposition routine MPI_ALLTOALL. The parallelization scheme of the code has an overall efficiency of about 90%, also for large systems with a big number of processors.

A wavelet run of ABINIT is optimal for simulating isolated systems like molecules or clusters. The localization of the basis set allows us to reduce considerably the number of degrees of freedom (*i.e.* the number of basis function which must be used) to attain a given absolute precision with respect to a plane-wave based formalism. This fact reduces the memory requirements and the number of floating point operations and, together with the optimal parallelization scheme, conceived for massive parallel computations, it allows to treat also systems with few hundred atoms.

4.3   Wavelet-based computation of Coulomb energy and potential with finite boundary conditions

As already discussed, the Deslauriers-Dubuc interpolating scaling function basis described in Sec.4.2 provide an interpretation of the point values of a function on a uniform mesh. In other terms, for a given function, its values on a real space grid can be interpreted as the expansion coefficients of that function in the interpolating scaling function basis. This basis is particularly well suited for solving electrostatic problems. Indeed, for a function expressed in this basis, the first m discrete and continuous moments are identical for a mth order interpolating wavelet family. For isolated boundary conditions, namely finite systems, the main features of an electrostatic potential are determined by the multipoles of the charge distribution.

For these reasons, within the BigDFT project, a Poisson solver based on interpolating scaling functions was developed [68]. This solver can also be used within other DFT methods, and is not restricted to wavelet basis sets. It is already integrated in different DFT codes, like CP2K and OCTOPUS. It is also integrated in the conventional plane-waves treatment of the ABINIT code. By putting the input variable **icoultrtmt**=1 this real space Poisson solver is used instead of the conventional reciprocal space treatment.

Let us summarize the main features of this solver. For a given charge distribution ρ, the Hartree potential V_H, solution of the Poisson's equation $\nabla^2 V_H = -4\pi\rho$, can be obtained from a three-dimensional integral

$$V_H(\mathbf{r}) = \int d\mathbf{r}' G(|\mathbf{r} - \mathbf{r}'|)\rho(\mathbf{r}') , \qquad (23)$$

where $G(r) = 1/r$ is the Green's function of the Laplacian operator for free boundary conditions. Within the interpolating scaling function description

39

the integral equation becomes a sum over a real space mesh with optimal quadrature coefficients, represented by a convolution with a kernel K:

$$V_{H_{j_1,j_2,j_3}} = \sum_{i_1,i_2,i_3} K(i_1 - j_1, i_2 - j_2, i_3 - j_3)\rho_{i_1,i_2,i_3} \ , \tag{24}$$

where the indices i_p and j_p run over the mesh points in the pth direction. The kernel function is expressed as the integral between the Green's function and the interpolating scaling function basis:

$$K(i_1, i_2, i_3) = \int \mathrm{dxdydz} \frac{\varphi(\mathrm{x} - \mathrm{i}_1)\varphi(\mathrm{y} - \mathrm{i}_2)\varphi(\mathrm{z} - \mathrm{i}_3)}{\sqrt{\mathrm{x}^2 + \mathrm{y}^2 + \mathrm{z}^2}} \tag{25}$$

where the interpolating scaling function is indicated by $\varphi(x)$. The quadrature rule defined by the kernel is optimal in the sense that it allows to express the point values of the Hartree potential by preserving the first multipoles of the continuous function defined in (23). This formalism is thus optimal for electrostatic problems in isolated boundary conditions. It is based on real-space point values and, given the Green's function approach to the problem, it allows the treatment of charged systems explicitly. Moreover, for a grid of N points, the convolution operation defined in (24) can be done with $\mathcal{O}(N \log N)$ scaling by using zero-padded FFT.

The kernel function can be calculated by performing a tensor product decomposition of the Green's function $1/r$ as a linear combination of Gaussians. This allows the kernel calculation to be separable in the three directions and reduces considerably the computational overhead. The kernel functions can thus be calculated once and for all at the beginning of the DFT calculation and stored in memory.

The Poisson solver obtained in this way exhibit excellent computational performances, also in parallel computation, with optimal accuracy. Moreover, a version of this solver based on interpolating scaling functions for slab-like boundary conditions has been developed [69]. This solver is integrated in the BigDFT standalone code and it will be integrated in the future versions of ABINIT, either with wavelets or plane waves calculations.

4.4 High-temperature DFT based on a recursion methodology

Increasing the temperature, KS DFT loses efficiency because the number of wavefunctions grows rapidly and actually DFT calculations are limited at $\sim 4\,eV$. To extend the DFT calculations to higher temperature ranges we introduce into ABINIT a method based on a recursion approach. This recursion method (RM), presented in this context in [70], avoiding Hamiltonian diagonalization, enables high temperature calculations and is an order-N method.

Like usual DFT, the RM is based on the Hohenberg-Kohn-Sham framework where the ground-state energy of a system is a functional of the electronic density $\rho(\mathbf{r})$ only. But in the RM, $\rho(\mathbf{r})$ is calculated directly from the Hamiltonian $H = -\frac{\triangle}{2} + V_{eff}$, without to solve the KS equation.

If $f(\epsilon_n)$ is the Fermi-Dirac distribution of the electrons in the state ψ_n of energy ϵ_n, then the electronic density can be written as:

$$\rho(\mathbf{r}) = \sum_n \psi_n^*(\mathbf{r}) f(\epsilon_n) \psi_n(\mathbf{r}) = 1 - \left\langle \mathbf{r} \left| \frac{1}{1 + e^{-\beta(H-\mu)}} \right| \mathbf{r} \right\rangle , \qquad (26)$$

where $|\mathbf{r}\rangle = \delta_{\mathbf{r}}$ is the Dirac-delta function centered in $\mathbf{r}$, β is the inverse of the temperature and μ the chemical potential. We can apply to ρ a partial fraction decomposition (PFD) which, for a reason that will be clear later, permits to boost the temperature $(1/\beta \rightarrow 2p/\beta)$. So we write the density as:

$$\rho(\mathbf{r}) = 1 - \frac{1}{2p} \sum_{j=0}^{2p-1} \left\langle \mathbf{r} \left| \frac{z_j}{z_j - e^{-\beta(H-\mu)/2p}} \right| \mathbf{r} \right\rangle . \qquad (27)$$

On the other hand, if a matrix A is tridiagonal in some basis $\{|\mathbf{u}_0\rangle, |\mathbf{u}_1\rangle, \dots\}$:

$$A|\mathbf{u}_n\rangle = b_n|\mathbf{u}_{n-1}\rangle + a_n|\mathbf{u}_n\rangle + b_{n+1}|\mathbf{u}_{n+1}\rangle, \qquad (28)$$

then $G(\mathbf{u}_0, z) = \langle \mathbf{u}_0 | 1/(z - A) | \mathbf{u}_0 \rangle$, the mean value of the Green operator associated to A on the first basis vector, can be expanded as a continuous fraction of the elements (a_n, b_n) of A [71]. So, if (a_n, b_n) are the tridiagonal elements of EH $\equiv e^{-\beta(H-\mu)/2p}$ in $\mathbf{r}$, then Eq. (27) is given by:

$$\rho(\mathbf{r}) = 1 - \frac{1}{2p} \sum_{j=o}^{2p-1} z_j G(\mathbf{r}, z_j) = 1 - \frac{1}{2p} \sum_{j=o}^{2p-1} \frac{z_j}{z_j - a_0 - \cfrac{b_1^2}{z_j - a_1 - \cfrac{b_2^2}{\dots}}}. \qquad (29)$$

In ABINIT, $\rho(\mathbf{r})$ is calculated on the discretized real-space which grid is defined by the variable **ngfft**. The fact that the local density on each grid point can be computed independently implies that the RM is highly scalable.

The tridiagonalization of the operator EH is implemented by a Lanczos algorithm [72]. For any $\mathbf{r}_i$ on the grid, posing $|\mathbf{u}_0\rangle = |\mathbf{r}_i\rangle$ and $b_0^i = 1$, this algorithm permits to find recursively the coefficients (a_n^i, b_n^i) and the basis vectors $\mathbf{u}_n^i$. If the numerical local density is converged under the desired precision or the recursion rank n exceeds the value fixed by **recnrec**, then the algorithm is stopped. The convergence properties of this algorithm are considered in [73].

During the Lanczos algorithm, the crucial step is the application of the EH on generic vector $|\mathbf{u}\rangle$. This is implemented by the second order Trotter formula

which permits to factorize the exponential of the sum of operators:

$$\text{EH}|\mathbf{u}\rangle = e^{\mu\beta/2p}e^{-V_l\beta/4p}\mathcal{F}^{-1}\left[\mathcal{F}[T]\mathcal{F}\left[e^{-V_l\beta/4p}|u\rangle\right]\right] + \text{ErrorTerm}, \qquad (30)$$

where $\mathcal{F}$ represents the FFT and $T(\mathbf{r}) = (p/\pi\beta)^{3/2}e^{-p\|\mathbf{r}\|^2/\beta}$ is the kernel if the convolution operator $e^{\beta\triangle/4p}$ [70]. The Error Term in Eq. (30) has been estimated in [73] and shown to decrease with the temperature. Therefore, when the density is expressed as in Eq. (27), we tune the parameter p in order to be able to neglect the Error Term in Eq. (30). The value of p can be varied by **recptrott**. Let us note that, at present, only the local part of the potential is implemented into Eq. (30).

The RM is activated by the keyword **tfkinfunc**=2. In any self-consistent loop, the density ρ is calculated on all the points of the grid and then μ is calculated by imposing that $\sum_i \rho(\mathbf{r}_i) = N$, the total number of electrons. When ρ has converged, the self-consistent loop is stopped. Finally the entropy S and the Grand Potential Ω are calculated by path integral methods using the (a_n^i, b_n^i) calculated during the RM:

$$S = \sum_i \frac{1}{2i\pi}\oint(h(z) + h(z^{-1}))G(\mathbf{r}_i, z)\mathrm{d}z, \qquad (31)$$

$$\Omega = -\frac{1}{2i\pi\beta}\sum_i \oint g(z)G(\mathbf{r}_i, z)\mathrm{d}z, \qquad (32)$$

where $G(\mathbf{r}_i, z)$ is defined by Eq. (29), $h(z) = -\frac{\ln(1+z^{2p})}{1+z^{2p}}$ and $g(z) = \ln(1 + z^{2p})$. From this, the pressure can computed in the usual way. In our implementation the path integral are discretized on a number of points given by **recnpath**. Using the previous relations, the internal energy is given by: $E = \Omega + S/\beta + \mu N$.

The size of the vector $\mathbf{u}$ in Eq. (30), which represents the interaction range and its locality, can be controlled by the definition of a radius of truncation **recrcut**. Its default value is zero, and it corresponds to take $\mathbf{u}$ of size equal to the size of the ngfft grid.

The discretization of the operator T in Eq. (30) requires, specially at high temperatures, a small grid step [73]. We introduces a double-grid system which permits to compute the local density $\rho(\mathbf{r}_i)$ on a coarse grid and the tridiagonalization, Eq. (30), on a fine grid. Successively the density and the recursion coefficients are interpolated on the fine grid by FFT interpolation. The keyword associated to the double grid system is **recgratio** representing the ratio between the two grid steps.

In ABINIT, when using Norm-Conserving Pseudopotentials, the electronic ground state of a periodic solid is represented with Bloch functions $\psi_{\mathbf{k},n}$ at each band n and $\mathbf{k}$-point. An alternative to using Bloch functions is to use Wannier functions $\mathcal{W}_n(\mathbf{r} - \mathbf{R})$, where $\mathbf{R}$ is the lattice vector of a unit cell [74].

Wannier functions are the solid-state equivalent of "localized molecular orbitals" [75], and thus provide a chemical picture of the system. It was N. Marzari and D. Vanderbilt who exploited the gauge of freedom of Wannier functions to construct Maximally-localized Wannier functions (MLWFs) [76]. MLWFs have found many useful applications: they are a powerful tool to interpolate generic operators and they are a minimal localized basis set for large-scale calculations, among others.

A set of N generalized Wannier functions $\mathcal{W}_{\mathbf{R},n}(\mathbf{r})$ can be constructed as

$$\mathcal{W}_{\mathbf{R},n}(\mathbf{r}) = \frac{V}{2\pi}^3 \int\limits_{BZ} d^3\mathbf{k} e^{-i\mathbf{k}\cdot\mathbf{R}} \sum_{m=1}^{N_{\mathbf{k}}} U_{mn}^{(\mathbf{k})} \psi_{\mathbf{k}m}(\mathbf{r}) \qquad (33)$$

from Bloch functions $\psi_{\mathbf{k}m}$, where V is the volume of the unit cell and the $N_{\mathbf{k}} \times N$ matrices are to be determined. In the case of an isolated manifold of bands $N_{\mathbf{k}}$ is equal to N and if $N_{\mathbf{k}} > N$ then an additional disentanglement procedure is needed [77].

To construct MLWFs we choose the $U_{mn}^{(\mathbf{k})}$ matrices that minimize the spread functional Ω,

$$\Omega = \sum_{n=1}^{N} \left[\langle \mathcal{W}_{\mathbf{0}n} | r^2 | \mathcal{W}_{\mathbf{0}n} \rangle - |\langle \mathcal{W}_{\mathbf{0}n} | \mathbf{r} | \mathcal{W}_{\mathbf{0}n} \rangle|^2 \right], \qquad (34)$$

where we refer to the unit cell at the origin, since all sets of Wannier functions are equivalent within a lattice-vector translation, $\mathcal{W}_{\mathbf{R}n}(\mathbf{r}) = \mathcal{W}_{\mathbf{0}n}(\mathbf{r} - \mathbf{R})$.

As it is shown in Ref. [76] the spread functional Ω can be expressed in terms of $U_{mn}^{(\mathbf{k})}$ and the overlap matrix:

$$M_{mn}^{(\mathbf{k},b)} = \langle \psi_{\mathbf{k}m} | e^{i\mathbf{b}\cdot\mathbf{r}} | \psi_{\mathbf{k}+\mathbf{b}n} \rangle, \qquad (35)$$

where $\mathbf{b}$ are vectors connecting each $\mathbf{k}$-point to its near neighbors. Having Ω in terms of $U_{mn}^{(\mathbf{k})}$ and $M_{mn}^{(\mathbf{k},b)}$, the gradient of the spread functional can be also obtained from these quantities. Therefore, $M_{mn}^{(\mathbf{k},b)}$ becomes the principal ingredient used in the algorithm to minimize the spread.

The minimization algorithm is started from an initial approximation to $U_{mn}^{(\mathbf{k})}$. A set of N localized trial orbitals $g_n(\mathbf{r})$ is chosen corresponding to some rough

initial guess for the MLWFs. This initial guess is given to the algorithm in the form of a $N_{\mathbf{k}} \times N$ matrix:

$$A_{mn}^{(\mathbf{k})} = \langle \psi_{\mathbf{k}m} | g_n \rangle \tag{36}$$

In short, Wannier90[90] minimizes the spread Ω with respect to the set of $U_{mn}^{(\mathbf{k})}$ to obtain MLWFs. The only ingredients needed are the matrices $M_{mn}^{(\mathbf{k},b)}$ and $A_{mn}^{(\mathbf{k})}$ which are provided by ABINIT in a transparent way to the user.

ABINIT builds the $M_{mn}^{(\mathbf{k},b)}$ matrix after a ground state calculation. In the case of norm-conserving Pseudopotentials it is obtained directly from the Eq. 35 and within the PAW formalism we require some extra terms (See section 3.4).

We have implemented a set of guiding functions g_n to build the $A_{mn}^{(\mathbf{k})}$ matrices in Eq. 36. We included the full set of features specified in Tables 3.1, 3.2 and 3.3 of the Wannier90 User Guide, allowing atomic-like orbitals and hybrids to be centered at arbitrary sites and oriented along arbitrary axes. For the radial part of g_n we added Gaussian functions which are faster to compute than the hydrogenic radial functions listed in Table 3.3. Since the $A_{mn}^{(\mathbf{k})}$ are just a rough initial guess, we have $A_{mn}^{(\mathbf{k})} = \langle \tilde{\psi}_{\mathbf{k}m} | g_n \rangle$ within the PAW approximation (so far, we have found that this is sufficient and there is no need to compute the inside-sphere contributions).

The input variables related to the Wannier90 interface are just three: prtwant, w90iniproj and w90prtunk. "prtwant" is a flag to indicate the interface will be used, "w90iniproj" controls the form of the guiding functions g_n mentioned above and "w90prtunk" is used to plot the Wannier functions. There is a secondary input file which is used to set all variables related to the Wannier90 algorithms as documented in the Wannier90 User Guide. For instance, we can indicate the number of steps in the minimization and the form of the initial guiding functions, among others.

Guided by the Web site, a newcomer will be able to learn how to use the Wannier90 interface without any external help. We chose four examples to introduce the user progressively to this interface. In addition, we also provide internal tests with more advanced capabilities such as the inclusion of GW corrections as explained in section 6.5.

4.6 Interfacing with the Libxc library

In order to solve the KS equations, an approximation to E_{xc} and v_{xc} is required. Many different approximations exist (we estimate them to be on the order of 200), but there are usually very few implemented in most computer codes. This means it may be difficult to reproduce calculations made with other codes, with older functionals, or with state-of-the-art functionals. The fact

that each code implements its own version of a given functional is also a waste of resources and developers time. The Libxc library was born within the Octopus project [78] as an attempt to solve some of these problems. It does so by providing implementations of as many functionals as possible in a well tested, robust library. Libxc was written in C from scratch, it includes bindings both in C and in Fortran, and is released under the GNU Lesser General Public License (version 3.0). Exchange-correlation functionals are usually divided into families according to the ingredients that are used to calculate E_{xc} [79]. The simplest approximation is the local density approximation (LDA), where E_{xc} depends only on the electronic density. Generalized gradient approximations (GGA) also depend on the gradient of the electronic density ∇n, while for the meta-GGAs further dependencies on $\nabla^2 n$ and/or on the kinetic energy density τ are added. Hybrid functionals are constructed by mixing a fraction of the exact exchange with GGA exchange and correlation. Libxc includes by now 19 LDA, 55 GGA, 24 hybrid, and 7 meta-GGA functionals. It is able to return E_{xc} and the necessary quantities to compute v_{xc}, as well as higher order derivatives of E_{xc}, although the later are not yet available for all the functionals.

In ABINIT it is possible to use the LDA and GGA functionals from Libxc to obtain E_{xc} and v_{xc}. To do so, ABINIT should be compiled with the Libxc plug-in enabled (configure keyword `--enable-libxc` from ABINIT 5.8, `--enable-etsf-xc` in older versions). These functionals are accessed by negative values of the **ixc** keyword. The functionals from Libxc can be either exchange-only functionals, correlation-only functionals, or combined exchange-correlation functionals. Each of them is uniquely identified by an integer ranging from 0 to 999. The absolute value of **ixc** to be used is obtained by concatenating the identifiers of the two functionals when one uses an exchange-only with a correlation-only functional, or by just taking the corresponding identifier when one uses a combined exchange-correlation functional. As an example, let us consider the Perdew-Burke-Ernzerhof GGA. This functional has an exchange part and a correlation part, whose identifiers are 101 and 130 respectively. Thus, to use this functional, **ixc** should be set to -101130.

The Libxc interface in ABINIT was tested by comparing results obtained using the native ABINIT functionals with results obtained with the corresponding Libxc functionals. The spin-unpolarized version of the functionals was tested by running a ground-state calculation of the helium atom, while the spin-polarized version was tested by running a ground-state calculation of the bismuth atom. The results were excellent, with an agreement better than 10^{-6} a.u. in the total energy for most cases. Automatic tests for all the LDA and GGA functionals included in Libxc, both in the spin-unpolarized and spin-polarized cases, are provided in the directory `tests/etsf_xc` for portability checking purposes.

The European Theoretical Spectroscopy Facility (ETSF) has recently normalized a file format to improve the inter-operability of *ab initio* codes within its scope [80]. In its 2.2 version, this normalization is focused on crystallographic data, density files and wavefunctions restart information. Indeed, ABINIT is often used as a starting point for post-ground state calculations. In these cases, the converged wavefunctions and the description of the basis-set are saved in a _KSS file and can be used as input for *GW* calculations in GWST [81] for instance or can be used to process excited states with e.g. Yambo [82]. The interoperability goal of the ETSF file format should also make possible the exchange of restart files between codes doing the same kind of calculations to help cross-checking the results. As a final point, the technology suggested by the ETSF to implement this file format, namely NetCDF [83,84], can bring the guarantee of platform independence to ABINIT, as well as backward-compatibility and extendability for its own output files.

To help the adoption of this normalized file format, a Fortran90 library has been developed. This library, called ETSF_IO [85], is code-independent and gives routines to easily read and write NetCDF files in accordance to the specifications. This library is used by ABINIT for crystallographic information, the density (namely the _DEN file) and wavefunctions (the _WFK file) data, and the _KSS file, which is used as a starting point for excited-state calculations in time-dependent DFT or the *GW* approximation.

ETSF_IO is added as a plug-in and enabled at configure time with the configure keyword `--enable-etsf-io`. It requires a valid Fortran implementation of NetCDF. Output files are generated using this format when the ABINIT input keyword **accesswff** is set to 3 (as a reminder, 0 corresponds to the ABINIT Fortran binary format, 1 switches to the use of MPI/IO in a parallel run and 2 is used to generate ABINIT binary files with NetCDF). All generated files have names appended with '-etsf.nc'.

In addition to the implementation of the ETSF file format through the ETSF_IO library in the main executable of the ABINIT package, one option has been added to the executable `cut3d` to be able to read the density files written in this new file format. This is the option 2 when `cut3d` asks for the file format.

To ensure the stability of this IO feature, ten automatic tests have been written in the `tests/etsf_io/` subdirectory of the source tree, checking density output, `cut3d` reading of densities, KSS generation and wavefunctions write and restart (in the wavelet formalism).

5 Developments in linear and non-linear responses

5.1 Electron-phonon coupling

Many everyday phenomena demonstrate the coupling of electrons to vibrations in crystals. Finite temperature effects, such as Raman spectral shapes, thermal and electrical resistivity or the band gap (see Sect. 5.2), and superconductivity, depend intricately on the coupling between electrons and phonons (EPC). The subject of electron-phonon coupling is well reviewed in Grimvall [91], and Allen and Mitrović [92]. The formalism in `ABINIT` follows broadly Savrasov and Savrasov [93].

Here we describe the implementation in `ABINIT`, the associated variables and workflow, and one application. The calculation of EPC is the subject of the `telphon` Tutorial in `ABINIT`. For a phonon of wavevector q and mode j, let $\mathbf{d}_{qj}$ be its displacement vector, and ω_{qj} its frequency. The matrix elements of the coupling between electrons and phonons are $\mathbf{m}_{ki,k+qi'} = \langle k+q\,i' \mid \nabla V \mid k\,i \rangle$, with $\mid k\,i \rangle$ an electronic state in band i, and ∇V the potential perturbed by atomic displacements.

In an `ABINIT` phonon calculation, the input keyword **prtgkk** should be set to 1 in order to output the matrix elements, in files suffixed _GKKx, where x is the index of the perturbation. To ensure all the needed perturbations are calculated, **prepgkk** should be set to 1. The symmetry operations may be used to complete the q-points in the irreducible Brillouin Zone. It is required that the grid of k contain the Γ point, and that the grid of q be a sub-grid of the k (e.g. 2×2×2 q and 4×4×4 k). Once the matrix elements have been calculated, the _GKK files are merged by the `mrggkk` utility. All perturbations at a given q-point must be grouped together.

Analysing electron-phonon quantities
The spectral representation of the phonon self-energy is given by the Eliashberg function $\alpha^2 F(\Omega) = N_F \sum_{jkik'i'} |\mathbf{d}_{qj} \cdot \mathbf{m}_{k\,i\,k'\,i'}|^2 \delta(\Omega - \omega_{qj})\delta(\epsilon_{ki})\delta(\epsilon_{k'i'})$ where $q = k - k'$ and N_F is the density of states at the Fermi level E_F. The integrand contains $\delta(\epsilon_{ki})\delta(\epsilon_{k'i'})$, which requires a very dense grid of k-points. The interaction with the electrons gives an expression for the phonon linewidth $\gamma_q^j = 1/2\omega_{qj} \sum_{kii'} |\mathbf{d}_{qj} \cdot \mathbf{m}_{k\,i\,k+q\,i'}|^2$.

`anaddb` interpolates $\mathbf{m}_{k\,k+q}$ onto the ground state k-point grid by Fourier transform. The interpolated object here is a product of $\mathbf{m}$ at a given q:

$$\bar{\gamma}_{k\,i\,k+q\,i'}^{\kappa\,\alpha\,\kappa'\,\beta} = (\mathbf{m}_{k\,i\,k+q\,i'}^{\kappa\,\alpha})^\dagger \mathbf{m}_{k\,i\,k+q\,i'}^{\kappa'\,\beta} \tag{37}$$

where the κ label atoms in the unit cell and α, β Cartesian directions. As

can be seen, $\bar{\gamma}$ eliminates all phase factors due to wavefunctions, by construction. $\bar{\gamma}(\kappa\alpha\kappa'\beta)$ can be symmetrized and interpolated in the same way as the dynamical matrix, although formally the $\mathbf{m}$ are not proper objects for interpolation[94], in particular due to crossings of electronic bands.

The main analysis is performed in `anaddb`, setting **elphflag** variable to 1. Most of the procedure is automatic, but can be lengthy if a large number of k-points is being used. The **nqpath** and **qpath** variables must be set, specifying a path in reciprocal space. `anaddb` generates files containing the phonon linewidths (suffixed _LWD) and frequencies ω_{qj} (suffixed _BST) along **qpath**. One can calculate the nesting function $n(q) = \sum_{kii'} \delta(\epsilon_{ki})\delta(\epsilon_{k+qi'})$ by setting **prtnest** to 1 (output to _NEST).

After interpolation, the quantities must be integrated over the Fermi surface. Several quadrature methods are available. The default (**telphint**=1) is to use Gaussian weighting, with a width **telphsmear**. Another option is the improved tetrahedron method[95] (**telphint**=0). Finally (**telphint**=2), one can integrate a given set of electron bands, between **ep_b_max** and **ep_b_min**. The resulting integrated quantities are the Eliashberg function (in a file suffixed _A2F), and the EPC strength λ which is printed in the main output file.

The calculation of bulk transport quantities (electrical and thermal conductivities) is possible using `anaddb`. Analogous quantities are obtained from the `conducti` post-processor, but due to electron-electron scattering, instead of electron-phonon. A preliminary calculation of the derivatives of the wavefunctions with respect to k-vector must be carried out. The transport calculation is turned on by setting **ifltransport** to 1 in `anaddb`. The transport quantities depend on the Fermi velocity for each band, and the electronic band-dependence of the matrix elements must be preserved before integration, by setting **tkeepbands** to 1. This increases the memory used, by the square of the number of bands crossing E_F. The results are the transport Eliashberg function (in file _A2F_TR), the electrical resistivity (in file _RHO), and the thermal conductivity (in file _WTH).

Tests inputs in files tests/v5/Input/t85.in to t99.in concern electron-phonon calculations and transport.

A number of applications have been published calculating electron-phonon coupling using `ABINIT`. The coupling in one dimensional systems (nanowires) has been studied in Ref. [96] and [97]. In the former we demonstrate changes in the coupling due to nanostructural effects. Fig. 5 shows the Eliashberg functions for atomic wires of Al, and huge variations of the EPC for wires which have a zigzag shape.
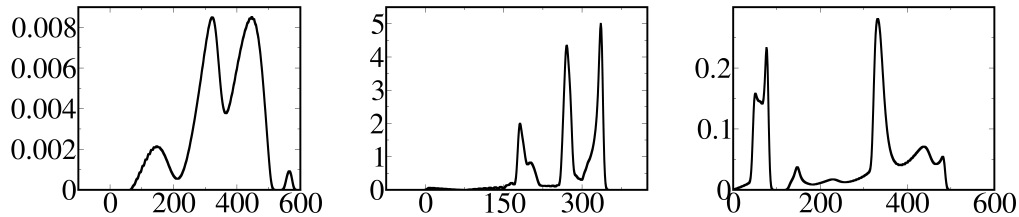
Fig. 5. Eliashberg spectral functions (dimensionless) of nanowires of Al, as a function of frequency (in cm^{-1}). The left function is that of a straight Al wire, the middle of a zigzag wire with bond angles of $\sim 60°$. The right hand panel is for a stretched zigzag wire, where all phonon modes have been stabilized. Huge size and structural effects appear in the electron-phonon coupling, covering several orders of magnitude. Adapted from Ref. [96]

5.2 Temperature-dependence of the band gap

Consider a crystal with atoms labelled by κ which occupy sites $\mathbf{R}(\mathbf{l}\kappa)$ and have displacements $\mathbf{u}(\mathbf{l}\kappa)$ from the equilibrium, where $\mathbf{l}$ labels the unit cells. The displacement vectors $\mathbf{u}$ are time dependent, they can be written as a Fourier series using the phonon frequencies. However, in the adiabatic approximation, one neglects this temporal dependence and treats $\mathbf{u}_{\kappa,\alpha}$ as a parameter. The systems is now reduced to a statically perturbed lattice, within an error of the order of $(m/M)^{1/2}$ (electron to atom mass ratio). To second order in $\mathbf{u}$, the energy change of the eigenvalue $\epsilon_{\mathbf{k}n}$ is given by $\Delta\epsilon_{\mathbf{k}n} = \sum_{\alpha\mathbf{l}\kappa,\beta\mathbf{l}'\kappa'} \epsilon^{(2)}_{\mathbf{k}n} u_\alpha(\mathbf{l}\kappa)u_\beta(\mathbf{l}'\kappa')$, where the second-order eigenvalue, $\epsilon^{(2)}_{\mathbf{k}n}$ is obtained by:

$$\epsilon^{(2)}_{\mathbf{k}n} = \frac{1}{2} < \mathbf{k}n| \frac{\partial^2 V}{\partial R_\alpha(\mathbf{l}\kappa)\partial R_\beta(\mathbf{l}'\kappa')}|\mathbf{k}n > + \sum_{\mathbf{k}'n'} \frac{< \mathbf{k}n|\frac{\partial V}{\partial R_\alpha(\mathbf{l}\kappa)}|\mathbf{k}'n' >< \mathbf{k}'n'|\frac{\partial V}{\partial R_\beta(\mathbf{l}'\kappa')}|\mathbf{k}n >}{\epsilon_{\mathbf{k}n} - \epsilon_{\mathbf{k}'n'}}.$$

(38)

The first term in Eq. (38) corresponds to the first order expansion of the Debye-Waller factor[98], representing the lowest order phonon screening of the ionic potential. The full Debye-Waller theory is equivalent to doing infinite even-order perturbation theory. The second term in Eq. (38) is the adiabatic approximation to the Fan theory [99], i.e. where the phonon energies have been neglected in the denominators. This contribution corresponds to a self-energy in the quasiparticle framework and is frequently referred as the self-energy contribution.

Allen and Heine [100] where the firsts to discuss the properties of these corrections in the case of the long wavelength acoustic modes. The condition of translational invariance requires that the eigenvalues are unaffected if every atom in the crystal is displaced by a uniform displacement vector $\mathbf{u}$. This implies a relation between the elements of the Debye-Waller contribution and the self-energy matrix elements. Using the rigid-ion approximation, in which the

49

non-diagonal Debye-Waller elements are neglected, this has enabled Allen and Heine to entirely eliminate the use of second-order Hamiltonians, thus considerably simplifying the formalism for semi-empirical calculations. Finally, performing a Fourier transform to the phonon representation, using Bloch's theorem and the orthogonality relations between the phonon displacement vectors, we obtain that $\Delta\epsilon_{\mathbf{k}n}(T) = \sum_{\mathbf{Q}j} \frac{\partial\epsilon_{\mathbf{k}n}}{\partial n_{\mathbf{Q}j}}[n_{\mathbf{Q}j}(T) + 1/2]$, where $\mathbf{Q}$ is a phonon wavevector, j labels a phonon branch and $n_{\mathbf{Q}j}(T)$ is the Bose-Einstein distribution. The real part of the coefficient $\frac{\partial\epsilon_{\mathbf{k}n}}{\partial n_{\mathbf{Q}j}}$ contributing to the energy shift is the sum of the Debye-Waller and Fan parts,

$$\frac{\partial\epsilon_{\mathbf{k}n}}{\partial n_{\mathbf{Q}j}} = \left[\frac{\partial\epsilon_{\mathbf{k}n}}{\partial n_{\mathbf{Q}j}}\right]_{DW} + \left[\frac{\partial\epsilon_{\mathbf{k}n}}{\partial n_{\mathbf{Q}j}}\right]_{Fan}, \qquad (39)$$

and their explicit expression is given in equations (3) and (4) of reference [101].

The above results have been derived using normal perturbation theory and have been used in many semi-empirical calculations [100–103] to model the temperature dependence of tetrahedral semiconductors. In the context of modern density-functional theory (DFT) calculations, these eigenvalue shifts must be revisited from the point of view of adiabatic density-functional perturbation theory [104]. It results in a reformulation of the second-order eigenvalue:

$$\epsilon_{\mathbf{k}n}^{(2)} = \frac{1}{2} < \mathbf{k}n| \frac{\partial^2 V}{\partial R_\alpha(\mathbf{l}\kappa)\partial R_\beta(\mathbf{l'}\kappa')}|\mathbf{k}n > + < \psi_{\mathbf{k}n,\alpha\mathbf{l}\kappa}^{(1)}|\frac{\partial V}{\partial R_\beta(\mathbf{l'}\kappa')}|\mathbf{k}n > \qquad (40)$$

$$+ < \mathbf{k}n|\frac{\partial V}{\partial R_\alpha(\mathbf{l}\kappa)}|\psi_{\mathbf{k}n,\beta\mathbf{l'}\kappa'}^{(1)} > + < \psi_{\mathbf{k}n,\alpha\mathbf{l}\kappa}^{(1)}|H^{(0)} - \epsilon_{\mathbf{k}n}|\psi_{\mathbf{k}n,\beta\mathbf{l'}\kappa'}^{(1)} >$$

$$- \sum_{\mathbf{k'}n'}^{N} \frac{< \mathbf{k}n|\frac{\partial V}{\partial R_\alpha(\mathbf{l}\kappa)}|\mathbf{k'}n' >< \mathbf{k'}n'|\frac{\partial V}{\partial R_\beta(\mathbf{l'}\kappa')}|\mathbf{k}n >}{\epsilon_{\mathbf{k}n} - \epsilon_{\mathbf{k'}n'}}.$$

The main effect of this new formalism is the splitting of the Fan contribution into multiple terms. The last of which as the same form as before, but now the sum is limited to the N occupied bands. The new terms, which include first-order wavefunctions that are orthonormal to the occupied states, are the conduction band contributions to the energy shifts. It is this equation which is used in the current version of the ABINIT package to calculate the second-order eigenvalues. The first order wavefunctions are calculated self-consistently using a variational procedure.

Calculation of the second-order electronic eigenvalues (`abinis` : `ieig2rf` 1)

The above mentioned DFPT formalism was coded in the ABINIT package. It is activated by setting the variable **ieig2rf** (Integer for second-order EIGvalues from Response Function) to 1. This will trigger the production of a file, named

with the extension _EIGR2D, containing the second-order electronic eigenvalues for the perturbation.

Post-processing using `anaddb` *(anaddb : **thmflag** 3,**telphint** 0 or 1)*

An actual calculation will usually involve many perturbations, i.e. many phonon wavelengths, and thus involves the creation of many _EIGR2D files. A simple post-processing code (`mrgddb`) can validate the consistency of these files, verifying that they correspond to calculations using the same parameters, and handle their merging into one single file.

The thermal corrections to the electronic eigenvalues can be computed using the `anaddb` post-processing code, by setting the variable **thmflag** to 3. In that case, the code will read the second-order electronic eigenvalues, specified at input as the electron-phonon matrix, and calculate all the quantities in Eq. (39). By setting the variable **telphint** to 1, the user can choose to use the improved tetrahedron method interpolation scheme.

Automatic Tests

The implementation of this formalism is tested with four automatic tests, one for each of the steps required to output the thermal corrections. These tests are located in the `tests/v5` directory, tests `t26.in`, `t27.in`, `t29.in` and `t30.in`. They focus on the thermal properties of the direct gap of silicon. They reproduce the results found in Ref. [102] if converged parameters are used. These files can be consulted as reference to produce input files for other systems.

5.3   Raman intensities and electro-optic properties

In Raman experiments, the incident light, usually a polarized or unpolarized laser, is scattered by the sample, and the energy as well as polarization of the outgoing light is measured. A Raman spectrum, presenting the energy of the outgoing photons, will consist of rather well-defined peaks, around an elastic peak.

At the lowest order of the theory, the dominant mechanism is the absorption or emission of a phonon by a photon. The energy of the absorbed or emitted phonon corresponds to the energy difference between the outgoing and incident photons. Thus, even more straightforwardly than the IR spectrum, a Raman spectrum is directly related to the energy of phonons at the Brillouin-zone center: when the zero of the energy scale is set at the incident light energy, the absolute value of the energy of the peaks corresponds to the energy of the phonons.

The Raman intensity of a specific mode is determined by the specific Raman scattering efficiency (S) via a prefactor that includes the frequency of the incoming photon, ω_0, the frequency ω_m of the phonon m and the temperature, T. Its full expression is [105,106]:

$$\frac{dS}{d\Omega} = \frac{(\omega_0 - \omega_m)^4}{c^4} |\mathbf{e_s}.\alpha^m.\mathbf{e_0}|^2 \frac{h}{2\omega_m} (n_m + 1) \tag{41}$$

where the dependency on temperature is given by the boson factor:

$$n_m = \frac{1}{e^{\hbar\omega_m/k_B T} - 1} \tag{42}$$

and the Raman tensor:

$$\alpha_{ij}^m = \sqrt{\Omega_0} \sum_{\kappa\beta} \frac{\partial \chi_{ij}^{\infty(1)}}{\partial \tau_{\kappa\beta}} \eta_m(\kappa\beta) \tag{43}$$

is related to the derivative of the linear electronic dielectric susceptibility tensor $\chi_{ij}^{\infty(1)}$ with respect to the set of atomic displacements $\tau_{\kappa\beta}$ associated to the phonon eigendisplacement vector $\eta_m(\kappa\beta)$. The $|\mathbf{e_s}.\alpha^m.\mathbf{e_0}|$ term in Eq. 41 gives the coupling between the incoming phonon with polarization $\mathbf{e_0}$, the crystal, characterized by a Raman tensor α^m and the scattered phonon with polarization $\mathbf{e_s}$.

For non-centrosymmetric crystals, some modes are active in both Raman and infrared. In this case, the same formalism as above holds for the TO modes (E=0), while a supplementary correction, arising from the macroscopic electric field generated by the polar lattice vibration, is needed for the modes in LO geometry (D=0). In this case, the Raman tensor becomes:

$$\frac{\partial \chi_{ij}^{\infty(1)}}{\partial \tau_{\kappa\beta}}\Big|_{D=0} = \frac{\partial \chi_{ij}^{\infty(1)}}{\partial \tau_{\kappa\beta}}\Big|_{\varepsilon=0} - \frac{8\pi}{\Omega_0} \frac{\sum_l Z_{\kappa,\beta l}^* q_l}{\sum_{ll'} q_l \epsilon_{ll'}^\infty q_l'} \sum_l \chi_{ijl}^{\infty(2)} q_l \tag{44}$$

where Z^* is the Born effective charge tensor, ϵ^∞ is the optical dielectric constant and $\chi^{\infty(2)}$ is the non-linear optical dielectric susceptibility. Quite often, experimental data are recorded using polarized or unpolarized lasers on powder samples. If we neglect surface effects on the Raman tensor, an approximation that is valid for meso- and macro-crystals, then the resulting Raman spectra can be obtained by performing averages over all possible orientations of the crystal and over all possible laser polarizations. The latter part is equivalent to summing up over the parallel and perpendicular laser polarizations. The averages over all possible crystal orientations for TO modes are given by [107–111] :

$$G_0 = \frac{(\alpha_{xx} + \alpha_{yy} + \alpha_{zz})^2}{3}$$

$$G_1 = \frac{(\alpha_{xy} - \alpha_{yz})^2 + (\alpha_{yz} - \alpha_{zx})^2 + (\alpha_{zx} - \alpha_{xy})^2}{2}$$

$$G_2 = \frac{(\alpha_{xy} + \alpha_{yz})^2 + (\alpha_{yz} + \alpha_{zx})^2 + (\alpha_{zx} + \alpha_{xy})^2}{2} +$$
$$\frac{(\alpha_{xy} - \alpha_{yy})^2 + (\alpha_{yy} - \alpha_{zz})^2 + (\alpha_{zz} - \alpha_{xx})^2}{3}$$

The intensities are then:

$$I_{\parallel}^{powder} = C(10G_0 + 4G_2)$$

$$I_{\perp}^{powder} = C(5G_1 + 3G_2)$$

$$I_{total}^{powder} = I_{\parallel}^{powder} + I_{\perp}^{powder}$$

Both the derivatives of the linear electronic dielectric susceptibilities with respect to atomic displacements and the non-linear electronic dielectric susceptibilities required to evaluate the Raman intensities are non-linear responses which can be formulated as third derivatives of the energy. In the ABINIT implementation, they are computed within the density functional perturbation theory, as described in Ref. [105]. Thanks to the $2n + 1$ theorem, their formulation only requires the knowledge of the ground-state and first-order changes in the wave-functions.

This non-linear response formalism has been successfully applied to a large variety of systems. We have so far studied the Raman spectra of ferroelectric oxides (LiNbO$_3$ [112], BaTiO$_3$ and PbTiO$_3$ [113]), different minerals under pressure conditions characteristic to the interior of the Earth [111,114,115], organic crystals (oligothiophenes [116]) and molecular solids under extreme conditions [117,118]. The ABINIT implementation also serves to build the WURM database of computed Raman spectra for minerals [119]. Although the present implementation is restricted to non-spin-polarized cases, the Raman intensities of magnetic systems is nevertheless accessible using a finite difference technique, as recently illustrated on the prototypical multiferroic BiFeO$_3$ [120,121]. The computation of the non-linear optical susceptibilities has also been applied to several polar dielectrics [122].

As a by-product of the calculation of the Raman tensor and non-linear optical coefficients, it is also possible to determine directly within ABINIT the electro-optic (EO) coefficients $r_{ij\gamma}$ (Pockels effect) which describe the change of optical dielectric tensor in a (quasi-)static electric field through the following

expression [105]:

$$\Delta \left(\varepsilon^{-1} \right)_{ij} = \sum_{\gamma=1}^{3} r_{ij\gamma} \mathcal{E}_{\gamma}. \qquad (45)$$

The clamped (zero strain) EO coefficients include an electronic and an ionic contribution directly accessible within ABINIT. The unclamped EO coefficients include an additional piezoelectric contribution which must be computed separately from the knowledge of the elasto-optic and piezoelectric strain coefficients. This formalism was for instance applied to different ferroelectric ABO_3 compounds [123,124].

6 Developments in excited state calculations

6.1 Hedin's equations and the GW approximation

DFT performs reasonably well for the determination of structural properties, but fails to predict accurate band gaps. A more rigorous framework for the description of excited states is provided by many-body perturbation theory (MBPT) [125,126], based on the Green's functions formalism and the concept of quasi-particles [127]. Within MBPT, one can calculate the quasi-particle (QP) energies, E, and amplitudes, Ψ, by solving the nonlinear equation

$$\left[T + v_H[n](\mathbf{r}_1) + v_{\text{ext}}(\mathbf{r}_1)\right]\Psi(\mathbf{r}_1) + \int \Sigma(\mathbf{r}_1, \mathbf{r}_2; E)\Psi(\mathbf{r}_2)\,\mathrm{d}\mathbf{r}_2 = E\Psi(\mathbf{r}_1), \quad (46)$$

involving the non-Hermitian, nonlocal and frequency dependent self-energy operator Σ. Eq. (46) goes beyond the mean-field approximation of independent KS particles as it accounts for the dynamic many-body effects in the electron-electron interaction. In principle, the exact self-energy can be obtained by solving self-consistently the set of coupled integro-differential equations proposed by Hedin [3]. The fundamental building blocks of Hedin's equations [2] are, besides $\Sigma(1,2)$, the Green's function of the interacting many-body system, $G(1,2)$, the Green's function of an appropriate noninteracting system, $G_0(1,2)$, and the irreducible polarizability, $\tilde{\chi}(1,2)$, which, through the inverse dielectric matrix $\varepsilon^{-1}(1,2)$, re-normalizes the static Coulomb potential, $v(1,2) = \frac{\delta(t_1-t_2)}{|\mathbf{r}_1-\mathbf{r}_2|}$, resulting in the dynamical screened interaction $W(1,2) = \varepsilon^{-1}(1,3)v(3,2)$. Finally, the vertex function $\Gamma(1,2,3)$ describes the interactions between virtual holes and electrons. The left panel of Fig. 6 shows how the various physical quantities are interrelated.

The practical solution of Hedin's equations is extremely complicated : they contain a functional derivative in Γ, whose direct evaluation is challenging. A tractable approach is provided by the so-called GW method [3], which consists in approximating Γ as a local and instantaneous function, *i.e.* $\Gamma^{\text{GW}}(1,2,3) = \delta(1,2)\delta(1,3)$, giving the relatively simpler set of four self-consistent equations depicted in the right panel of Fig. 6. In the first iteration, one starts with $\Sigma = 0$, and G is approximated with the Green's function, G_0, of an appropriate non-interacting system (in ABINIT, the KS system). The polarizability is hence given by the expression $\tilde{\chi}(1,2) = -i\,G_0(1,2)G_0(2,1^+)$, and the screened

---

[2] Henceforth, the following conventions are used. The natural number (1) is a contracted notation for the space-time coordinates $(\mathbf{r}_1, t_1)$ with $1^+ \equiv (\mathbf{r}_1, t_1 + \eta)_{\eta \to 0^+}$. Repeated indices are integrated over, unless they appear on both sides of the equation.
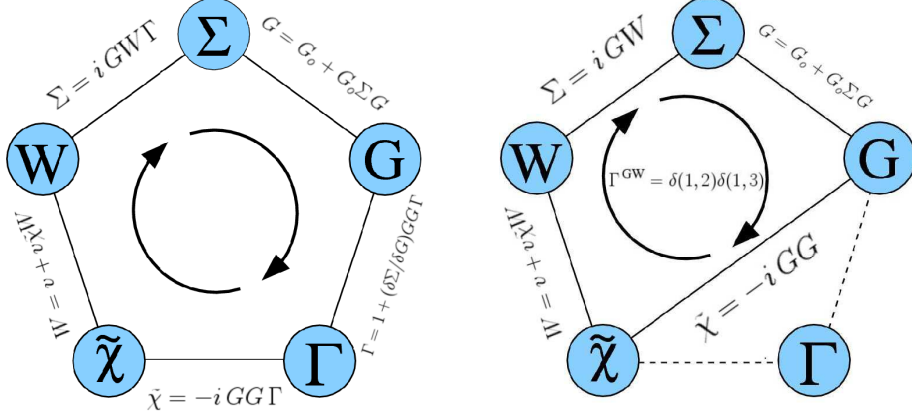
Fig. 6. Left panel: Graphical representation of Hedin's equations. Right panel: The four coupled integro-differential equations involved in the GW approximation. G_0W_0 corresponds to a single iteration of the "loop".

interaction $W(1,2) = \varepsilon^{-1}(1,3)v(3,2)$ is obtained using the dielectric matrix of the auxiliary KS system *i.e.* $\varepsilon(1,2) = \delta(1,2) - v(1,3)\tilde{\chi}(3,2)$. The new self-energy, $\Sigma(1,2) = iG(1,2)W(1^+,2)$ is constructed, and Eq.(46) is solved obtaining a new set of QP energies and amplitudes. Ideally, the GW loop should be iterated to self-consistency, but most applications use the so-called G_0W_0 approach in which only a single iteration is performed. Indeed, numerically costly self-consistent GW calculations were reported to worsen the results when compared to the relatively cheap G_0W_0 method [128,129].

Solving Eq.(46) for a given self-energy is not an easy task since it requires the knowledge of the frequency dependence of Σ. A simple, albeit accurate, solution for the QP energies can be obtained using first-order perturbation theory, considering v_{xc} as a static, local and Hermitian approximation to the true nonlocal and energy dependent Σ [130]. Following the perturbative approach, the QP energies are obtained as

$$
\begin{cases}
\varepsilon^{\mathrm{QP}} = \varepsilon^{\mathrm{KS}} + Z\langle\Psi^{\mathrm{KS}}|\Sigma(\varepsilon^{\mathrm{KS}}) - v_{\mathrm{xc}}|\Psi^{\mathrm{KS}}\rangle, \\[2mm]
Z = \left[1 - \langle\Psi^{\mathrm{KS}}|\frac{\partial\Sigma(\varepsilon)}{\partial\varepsilon^{\mathrm{KS}}}|\Psi^{\mathrm{KS}}\rangle\right]^{-1}.
\end{cases} \tag{47}
$$

A typical GW calculation consists of two different steps (following a DFT calculation): first ε^{-1} is calculated and stored on disk (`optdriver=3`), then the KS band structure and W are used to evaluate the matrix elements of Σ, finally obtaining the QP corrections (`optdriver=4`). We refer the reader to the two GW tutorials, `lesson_gw1.html` and `lesson_gw2.html`, in `doc/tutorial`, where a complete description of the entire procedure to be followed to obtain the QP band gap of silicon and the QP band structure of aluminum is extensively discussed.

In the frequency domain, the GW self-energy is given by the convolution

$$\Sigma(\omega) = \lim_{\delta \to 0^+} \frac{i}{2\pi} \int d\omega' e^{i\omega'\delta} G(\omega + \omega') W(\omega'), \qquad (48)$$

whose evaluation, in principle, requires the knowledge of the frequency dependence of $W(\omega')$. Due to the ragged behaviour of $G(\omega)$ and $W(\omega)$ along the real axis, a fine grid of real frequencies is in principle required to converge the results. ABINIT provides two different, more effective, techniques to evaluate Eq.(48): (i) integration with a plasmon-pole model (PPM) and (ii) integration with contour deformation (CD). In the former case, the frequency dependence of $\varepsilon^{-1}(\omega)$ is modeled with a simple analytic form, and the frequency convolution is carried out analytically. In the latter approach, the integral is evaluated numerically extending the functions in the complex plane in order have a smoother integrand.

Four different plasmon pole models (PPMs) are available in ABINIT. The choice of the particular PPM to be used is controlled by the variable **pp-model**. For examples of use of the PPM technique, see `tests/v3/t30.in` and `tests/v3/t31.in`. The first two options (**ppmodel**=1,2) refer to approximations employed in the pioneering implementations of the GW formalism: the plasmon-pole models of Godby-Needs [133] (GN) and Hybertsen and Louie [130] (HL). In both cases the imaginary part of the $\varepsilon^{-1}(\omega)$ is approximated in terms of a delta function centered at the plasmon frequency $\tilde{\omega}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})$ with amplitude $A_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})$, *i.e.*

$$\Im \varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega) = A_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}) \left[\delta(\omega - \tilde{\omega}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})) - \delta(\omega + \tilde{\omega}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})) \right]. \qquad (49)$$

The real part is then obtained by means of the Kramers-Kronig relation

$$\Re \varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega) = \delta_{\mathbf{G}_1\mathbf{G}_2} + \frac{\Omega^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})}{\omega^2 - \tilde{\omega}^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})}. \qquad (50)$$

where $\Omega^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}) = -A_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})\tilde{\omega}^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})$. Eq. (49) is a reasonable approximation since experiments and first-principles analysis reveal that $\Im W(\omega)$ is generally characterized by a sharp peak in correspondence to a plasmon excitation at the plasmon frequency. At this point, one defines a set of physical constraints to find the parameters entering Eq.(49) and (50). In the GN approach (**ppmodel**=1), the parameters of the model are derived such that $\varepsilon^{-1}(\omega)$ is correctly reproduced at two different frequencies: the static limit and an additional imaginary point located at the Sommerfeld plasma frequency $i\omega_{\mathrm{P}}$, where

$\omega_P = \sqrt{4\pi\rho}$ with ρ the number of electrons per volume [134]. After some algebra, the following set of equations defining the plasmon-pole coefficients can be derived:

$$
\begin{cases}
A_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}) = \varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega = 0) - \delta_{\mathbf{G}_1\mathbf{G}_2} = -\dfrac{\Omega^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})}{\tilde{\omega}^2_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})} \\[2em]
\tilde{\omega}^2_{\mathbf{G}_1\mathbf{G}_2} = \omega_P{}^2 \left[\dfrac{A_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})}{\varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega = 0) - \varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, i\omega_p)} - 1 \right]
\end{cases}
\tag{51}
$$

In the HL model (**ppmodel**=2), the PPM parameters are calculated so as to reproduce the static limit exactly and to fulfill a generalized f-sum rule relating the imaginary part of the many-body ε^{-1} to the plasma frequency and the charge density [135,136]. The final expression for the PPM parameters can be found in [130].

Models based on Eq.(49) and (50) have a number of undesirable features, despite their success. The total number of poles is large since it is equal to the square of the number of plane waves used to represent $\varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega)$, so expensive tables of $N_{\mathbf{G}}^2$ elements have to be stored in memory during the entire calculation. Moreover, for some elements with $\mathbf{G}_1 \neq \mathbf{G}_2$, the plasmon poles $\tilde{\omega}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q})$ can become very small or even imaginary which is somewhat unphysical [130]. ABINIT provides two more recent PPM approaches due to Von der Linden and Horsch [137] (vLH) and Engel and Farid [138] (EF), which are computationally more efficient and expected to be more accurate. The vLH plasmon-pole model with $N_{\mathbf{G}}$ parameters is derived starting from the spectral decomposition of the symmetrized inverse dielectric matrix $\tilde{\epsilon}^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega) \equiv \frac{|\mathbf{q}+\mathbf{G}_1|}{|\mathbf{q}+\mathbf{G}_2|}^{-1} \varepsilon^{-1}_{\mathbf{G}_1\mathbf{G}_2}(\mathbf{q}, \omega)$ [137] by assuming that the frequency dependence is solely contained in the eigenvalues (see Ref. [137]). The disadvantage of the vLH approach is that it satisfies the f-sum rule only for the diagonal elements. In the EF plasmon-pole model, also with $N_{\mathbf{G}}$ parameters, eigenvalues and eigenvectors are frequency dependent, and derived from an approximation to the reducible polarizability, $\chi(1, 2) \equiv \frac{\delta n(1)}{\delta v_{\text{ext}}(2)}$ [138], which is exact both in the static- and high-frequency limit. For further details on this plasmon-pole technique, see Ref. [138].

Since the frequency convolution in Eq.(48) can be carried out analytically once the plasmon-pole parameters are known, the PPM technique is the ideal tool for initial convergence studies. Besides it usually proves to be accurate to within 0.1-0.2 eV for states close to the Fermi level, when compared to results obtained with a costly numerical integration of Σ [139]. On the other hand, the accuracy worsens for states far from the gap, especially for low-lying states To analyze these physical properties, it is necessary to avoid PPM methods, and calculate explicitly the frequency dependence of $W(\omega)$ as described now.

A straightforward numerical evaluation of Eq.(48) is problematic due to the fact that G and W both have poles infinitesimally above and below the real axis.Therefore a naive integration algorithm along the real axis would need to evaluate the integrand precisely in the region where it is ill-behaved. ABINIT proposes an alternative route to evaluate Eq.(48), which traces back to the earliest GW calculations for the homogeneous electron gas [140]. The Green's function G and the screened Coulomb interaction W are analytic functions (except along the real axis) and can consequently be analytically continued to the full complex plane. The strategy is to use a deformation of the contour of integration in order to avoid as much as possible having to deal with quantities close to the real axis. Instead of evaluating the integral along the real axis, one evaluates the integral along the imaginary axis, and then adds the residues arising from the poles enclosed in the contour.

The contour deformation technique is activated by setting the input variable **gwcalctyp** to 2. The integration along the imaginary axis requires the calculation of $\varepsilon^{-1}(\omega)$ for purely imaginary frequencies. The frequency mesh for the quadrature is governed by the input variable **nfreqim**, and can be very coarse since the integrands is very smooth in this region. The evaluation of the residue of the poles requires the calculation of $\varepsilon^{-1}(\omega)$ on a fine mesh along the real axis. This regular mesh, sampling the interval $[0, +\infty]$, is defined by the two input variables **nfreqre** and **freqremax**. For examples of the contour deformation technique, see `tests/v4/t84.in`, `tests/v4/t85.in`, and `tests/paral/tU.in`. The CD approach requires many evaluations of $\varepsilon^{-1}(\omega)$ and can therefore be computationally highly demanding. On the other hand, it is the preferred approach for calculating the QP correction of low-lying states. Moreover, it is the only technique available in ABINIT to compute the imaginary part of $\Sigma(\omega)$ and the spectral function $A(\omega) = |\frac{1}{\pi} \Im G(\omega)|$.

6.3 Self-consistent Hartree-Fock, Screened Exchange, COHSEX, or GW

Following the original recipe of Hybertsen and Louie [130], the GW self-energy is usually constructed starting from a KS band-structure calculation, and the QP corrections are calculated using the perturbative approach described in Sec.(6.1). This procedure has been shown to be very efficient [139], but several questions arise. How much does the G_0W_0 result depend of the starting point? What happens if the starting KS band structure is qualitatively wrong? A self-consistent GW self-energy calculation should be free of such criticisms. However, the self-consistency in GW is everything but straightforward, since Σ, being non-Hermitian and energy-dependent, should have left and right energy-dependent eigenvectors. In practice, this is not tractable without approximation. Furthermore, fully self-consistent GW calculations worsen the results obtained with the standard one-shot G_0W_0 method.

A different approach to the self-consistency issue is the self-consistent quasi-particle GW approximation (SCqpGW) proposed by Faleev and co-workers [141–143]. SCqpGW employs a perturbation theory constructed around some initial single-particle Hamiltonian (for instance the Hartree Hamiltonian), and uses a self-consistent procedure to determine an optimal non-interacting Hamiltonian given by

$$h_{SCqpGW}(\mathbf{r}_1, \mathbf{r}_2) = -\tfrac{1}{2}\nabla^2 + v_H[n](\mathbf{r}_1) + v_{\text{ext}}(\mathbf{r}_1) + \tilde{\Sigma}(\mathbf{r}_1, \mathbf{r}_2), \qquad (52)$$

with the true self-energy approximated via a static and Hermitian operator $\tilde{\Sigma}(\mathbf{r}_1, \mathbf{r}_2)$ constructed according to

$$\tilde{\Sigma}_{ij}^{\mathbf{k}} \equiv \tfrac{1}{2}\,\mathrm{Herm}\Big[\Sigma_{ij}^{\mathbf{k}}(\varepsilon_{i\mathbf{k}}) + \Sigma_{ij}^{\mathbf{k}}(\varepsilon_{j\mathbf{k}}).\Big], \qquad (53)$$

In the above equation, Herm means that only the Hermitian part of the matrix is considered. Then the approximated self-energy, $\tilde{\Sigma}$, is diagonalized obtaining a new set of orthogonal QP amplitudes and real-valued QP energies. From this new set of orbitals, a new density and the corresponding $v_H[n]$ is generated, a new $\tilde{\Sigma}$ is constructed and the procedure is iterated to self-consistency. The SCqpGW approach improves the G_0W_0 results, giving band gaps very close to experiments with errors that are small and highly systematic [142]. Following the same spirit, Bruneval and coworkers [132] proposed to use an alternative Hermitian and static approximation to the GW self-energy: the COHSEX approximation already derived by Hedin in 1965 [3]. The COHSEX is a simple approximation that consists of two terms: the COulomb Hole and the Screened EXchange. These terms do not involve any summation over empty states. Performing self-consistency within the COHSEX approximation is hence more tractable than within the GW self-energy of Faleev and coworkers, although the COHSEX self-energy may be a cruder approximation than $\tilde{\Sigma}$.

These two methods are implemented in ABINIT. The corresponding input variable **gwcalctyp** has two digits. The first digit states the self-consistency type: 0 means no self-consistency, 1 self-consistency on eigenvalues only, 2 full self-consistency on wavefunctions and eigenvalues. The second digit governs the type of calculation: 5 stands for Hartree-Fock, 6 for screened exchange, 7 for COHSEX, 8 for Faleev's method with plasmon-pole model, 9 for Faleev's method with contour deformation technique.

Diagonalizing the operator in Eq.(52) in the reciprocal space representation would not be practical as the calculation of $\Sigma_{\mathbf{G}_1\mathbf{G}_2}^{\mathbf{k}}$ scales with the fourth power of the number of atoms [130]. The computational effort can be reduced significantly if a finite number of KS eigenstates is employed to describe the QP amplitudes. In most cases, the KS wavefunctions are not so different from the QP amplitudes, hence, KS states are expected to constitute an accurate basis set for the solution of the problem. As a consequence, after the diagonalization

of Eq.(52), the QP amplitudes, $|\Psi_{m\mathbf{k}}^{\mathrm{QP}}\rangle$, are expressed as linear combination of the KS states, $|\Psi_{n\mathbf{k}}^{\mathrm{KS}}\rangle$, used to initialize the first iteration of the cycle:

$$|\Phi_{m\mathbf{k}}^{\mathrm{QP}}\rangle = \sum_n \mathcal{U}_{mn}^{\mathbf{k}}|\Psi_{n\mathbf{k}}^{\mathrm{KS}}\rangle, \tag{54}$$

where $\mathcal{U}^{\mathbf{k}}$ is a unitary matrix. The size of the KS basis set used to represent the new wave functions is governed by the input variable **bdgw**, which consists of two integers defining the range of KS states to be included at each **k**-point. Several sample test files demonstrate how to use the different self-consistent approaches: see the example input files `t86.in`, `t87.in`, `t88.in` in directory `tests/v4/`, and the parallel tests `paral/tO.in`, `paral/tV.in` in `tests/paral/`.

6.4   The GW approximation with PAW

For several decades, thanks to the excellent agreement obtained with respect to experiments, pseudopotential-based methods have represented a *de facto* standard for MBPT calculations. In recent years, however, results obtained with all-electron (AE) approaches [144,143] have revealed that a fully consistent treatment of the electronic degrees of freedom produces *GW* band gaps that are systematically smaller than pseudopotential (PP) results, thus worsening the agreement between G_0W_0 and experiments. These findings have led to quite an intense debate in the scientific literature concerning the reliability of the PP approach for MBPT calculations, see Refs. [145–147].

The PAW formalism, discussed in Section 3.1, represents a flexible and efficient alternative to the use of pseudopotentials for the solution of the *GW* equations. It constitutes a full potential method which gives direct access to the true single-particle wavefunction with the correct nodal shape in the atomic regions. This allows for an accurate description of physical quantities or matrix elements sensitive to the nuclear region. In what follows, we only discuss how the implementation of the *GW* equation has to be generalized to include the PAW method.

Within the PAW formalism, the oscillator strengths –the basic ingredients required to evaluate $\tilde{\chi}^0(\omega)$, and the matrix elements of $\Sigma(\omega)$– can be obtained by means of the following equation [149]:

$$\langle \Psi_{b_1 \mathbf{k}-\mathbf{q}} | e^{-i(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} | \Psi_{b_2 \mathbf{k}} \rangle = \langle \tilde{\Psi}_{b_1 \mathbf{k}-\mathbf{q}} | e^{-i(\mathbf{q}+\mathbf{G})\cdot\mathbf{r}} | \tilde{\Psi}_{b_2 \mathbf{k}} \rangle + \sum_{ij} \langle \tilde{\Psi}_{b_1 \mathbf{k}-\mathbf{q}} | \tilde{p}_i \rangle \langle \tilde{p}_j | \tilde{\Psi}_{b_2 \mathbf{k}} \rangle$$

$$e^{-i(\mathbf{q}+\mathbf{G})\cdot\mathbf{R}_i} \underbrace{\left[\langle \phi_i | e^{-i(\mathbf{q}+\mathbf{G})\cdot(\mathbf{r}-\mathbf{R}_i)} | \phi_j \rangle - \langle \tilde{\phi}_i | e^{-i(\mathbf{q}+\mathbf{G})\cdot(\mathbf{r}-\mathbf{R}_i)} | \tilde{\phi}_j \rangle \right]}_{4\pi \sum_{lm} (-i)^l \mathcal{Y}_m^l(\widehat{\mathbf{q}+\mathbf{G}}) \, \mathcal{G}_{l_i m_i l_j m_j}^{lm} \int j_l(|\mathbf{q}+\mathbf{G}|r) \left(\phi_{n_i l_i} \phi_{n_j l_j} - \tilde{\phi}_{n_i l_i} \tilde{\phi}_{n_j l_j} \right) \mathrm{d}r}, \qquad (55)$$

where the plane wave has been expressed terms of Bessel functions $j_l(x)$ and real spherical harmonics $\mathcal{Y}_m^l(\widehat{\mathbf{G}})$ via the Rayleigh expansion. The symbol $\mathcal{G}_{l_i m_i l_j m_j}^{lm}$ is used to denote the Gaunt coefficient [150]. The last point worth discussing concerns the core-valence decoupling in a PAW formalism based on the frozen core approximation. When a perturbative approach is used to calculate QP corrections, the appropriate exchange-correlation (XC) contribution due to the valence has to be subtracted out from the Σ constructed from the valence electrons. The simplest approach to the core-valence decoupling consists in assuming that the core-valence exchange-correlation (XC) and core-polarization contribution of a valence state is given by [130]:

$$\langle \Psi | v_{\mathrm{xc}}{}^{\mathrm{core\text{-}val}} | \Psi \rangle \approx \langle \Psi | v_{\mathrm{xc}}[n_v + n_c] | \Psi \rangle - \langle \Psi | v_{\mathrm{xc}}[n_v] | \Psi \rangle. \qquad (56)$$

In this case, the valence-electron XC contribution is removed from the valence self-energy according to $\langle \Psi | \Sigma - v_{\mathrm{xc}}[n_v] | \Psi \rangle$ where $v_{\mathrm{xc}}[n_c + n_v] = v_{\mathrm{xc}}[n_c] + v_{\mathrm{xc}}[n_v]$ has been implicitly assumed. A typical input file showing how to run GW calculations with PAW can be found in `tests/v5/t66.in`.

6.5 Interpolation of GW band structure

The exact evaluation of the GW band structure along different symmetry lines in the Brillouin zone can be an extremely demanding task. At the end of a GW calculation we usually end up with the GW eigenvalues at a uniform mesh of $\mathbf{k}$-points, which is exactly what we need to construct Maximally-Localized Wannier Functions (MLWFs). Thus, it becomes natural to employ MLWFs to interpolate GW band structures.

Given a set of LDA or GW eigenvalues, $\epsilon_{n\mathbf{k}}$, calculated on a homogeneous grid of $\mathbf{k}$-points, we express the Hamiltonian $H_{mn}^{(\mathbf{k})} = \epsilon_{n\mathbf{k}}$ in the Wannier gauge representation,

$$H_{mn\mathbf{k}}^{\mathcal{W}} = \sum_{i=1}^{N} \left(U_{im}^{(\mathbf{k})} \right)^\dagger U_{in}^{(\mathbf{k})} \epsilon_{n\mathbf{k}}, \qquad (57)$$

where $U_{mn}^{(\mathbf{k})}$ are the matrices that minimize the spread of the Wannier Functions (See Section 4.5).

We then obtain the Hamiltonian in real space using a discrete Fourier transform,

$$H_{mn\mathbf{R}}^{\mathcal{W}} = \frac{1}{N_{\mathbf{k}}} \sum_{\mathbf{k}} e^{-i\mathbf{k}\cdot\mathbf{R}} H_{mn\mathbf{k}}^{\mathcal{W}}, \tag{58}$$

where $N_{\mathbf{k}}$ is the number of $\mathbf{k}$-points in the mesh.

Assuming that $|H_{mn\mathbf{k}}^{\mathcal{W}}|$ to decay fast with $|\mathbf{R}|$, we apply an inverse Fourier transform to calculate the Hamiltonian at any arbitrary $\mathbf{k}'$ point,

$$H_{mn\mathbf{k}'}^{\mathcal{W}} = \sum_{\mathbf{R}} e^{i\mathbf{k}'\cdot\mathbf{R}} H_{mn\mathbf{k}}^{\mathcal{W}}. \tag{59}$$

The final step is to diagonalize $H_{mn\mathbf{k}'}^{\mathcal{W}}$ to find the interpolated energies $\epsilon_{n\mathbf{k}'}$.

GW or LDA band structure interpolation based on MLWFs is a highly accurate method because it preserves valuable information about band connectivity. Hence, band crossings, avoided crossings and degeneracies are treated correctly at a low computational cost [154,155].

In the LDA case, ABINIT constructs MLWFs passing to Wannier90 the LDA wavefunctions $\psi_{n\mathbf{k}}^{LDA}$ and the LDA eigenvalues $\epsilon_{n\mathbf{k}}^{LDA}$ in the full irreducible Brillouin zone as explained in Section 4.5. Then, one uses the post-processing tools of Wannier90 to interpolate the band structure. In the GW case, the procedure is equivalent; one has to pass to Wannier90 the GW quantities instead of the LDA ones, which is done automatically by ABINIT setting **prtwant** to 3. In case we just calculated the GW corrections in a reduced set of $\mathbf{k}$-points of the irreducible Brillouin zone, ABINIT uses symmetry operations to obtain the corrections corresponding to the full irreducible Brillouin zone.

This feature is intended to be easy to use since we just have to employ one variable. In addition, we provide an internal test as an example of how to do GW band structure interpolation with self explanatory notes (`tests/wannier90/t03.in`).

6.6 Excitation energies from Time-dependent Density Functional Theory

Time-Dependent Density Functional Theory (TDDFT) is the extension of DFT to the time domain when the external potential is a function of time, and so it is the density. TDDFT relies on the Runge-Gross (RG) Theorem (1984) [156]. One of its successful applications is in the computation of the absorption spectrum of molecules within the linear response formalism (TDDFRT).

The dynamical polarizability $\alpha(\omega)$, describing optical absorption spectra of isolated systems, can be obtained from the linear-response function; the latter is calculated, within TDDFRT, by solving a Dyson-like equation [157]. This can be done in practice by solving an eigenvalue equation in the configuration

space, i.e. in the space of the products of two KS wavefunctions (one particle and one hole):

$$
\begin{pmatrix} A & B \\ B^* & A^* \end{pmatrix} \begin{pmatrix} X_I \\ Y_I \end{pmatrix} = \omega_I \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} X_I \\ Y_I \end{pmatrix}
\tag{60}
$$

where (the indexes ij and hk are particle-hole indexes in the KS basis, the Greek indexes refer to spin)

$$
A_{ij\sigma,hk\tau}(\omega) = \delta_{\sigma,\tau}\delta_{i,h}\delta_{j,k}(\epsilon_{h\tau} - \epsilon_{k\tau}) - \sqrt{|g_{i\sigma} - g_{j\sigma}|} f^{Hxc}_{ij\sigma,hk\tau}(\omega) \sqrt{|g_{h\tau} - g_{k\tau}|}
$$
$$
B_{ij\sigma,hk\tau}(\omega) = - \sqrt{|g_{i\sigma} - g_{j\sigma}|} f^{Hxc}_{ij\sigma,kh\tau}(\omega) \sqrt{|g_{k\tau} - g_{h\tau}|}.
$$

This procedure is usually referred as "TDDFT a la Casida" [158]. Here ϵ are the KS eigenvalues, g the KS occupation numbers, and $f_{Hxc} = \delta^2 A_{Hxc}[\rho]/\delta\rho^2$ is the exchange and correlation kernel of the TDDFRT (A_{Hxc} is the "Hartree+exchange and correlation" part of action of the system). Finally (X_I, Y_I) are the eigenvectors end ω_I the eigenvalues solution of the problem; in particular the eigenvectors are divided into the "particle-hole" and the "hole-particle" part X_I and Y_I respectively.

The matrix which has to be diagonalized in Eq.(60) is in general not Hermitian. However, for collinear and isolated systems (when only the Gamma point is involved) it is always possible to work with real KS wavefunctions and the problem can be rewritten in the form of an Hermitian matrix

$$
\Omega(\omega_I)F_I = \omega_I F_I
\tag{61}
$$

where

$$
\Omega = (A - B)^{-1/2}(A + B)(A - B)^{-1/2}
\tag{62}
$$

The resulting eigenvalues correspond to the excitations energies of the system, and the spectral intensities can be constructed from the corresponding eigenvectors. The ABINIT package solves Eq. (61) within the adiabatic-LDA (ALDA), i.e. by using the kernel: $f_{xc}(\mathbf{r}t, \mathbf{r}'t') \simeq \delta(t-t')\delta^2 E^{LDA}_{Hxc}(\rho)/\delta\rho(\mathbf{r})\delta\rho(\mathbf{r}'))$.

Both closed shell and open shell systems can be treated. In the first case, the matrix Ω (and so the eigenvalue problem (61)) can be blocked in two, one for the spin singlet channel and one for the spin triplet channel. For open shell systems, since the spin symmetry is not automatically respected, both spin channels are described by a single matrix. In the case of open shell systems TDDFT-ALDA is known to suffer from the *spin contamination* problem [159], which could only be solved by inclusion of double excitations. The latter, which are missing in any frequency-independent kernel, would be required for a qualitatively correct description of the spin configuration of some of the excited states [160]. In most systems, anyway, this problem affects only dipole-forbidden transitions.

To run "TDDFT a la Casida" within ABINIT one has to start from a previously computed ground state DFT calculation done with `prtden=1` and `prtwf=1`, so that the density and the KS electronic structure are written out into a file. By setting `iscf=-1`, `getden=1` and `getwf=1` ABINIT reads the needed information from the output of the previous run, and solves Eq.(61). All parameters have to be consistent with the ground state calculation, except the input variables **ixc** and **nband** which can be freely chosen. The former specifies the chosen approximation for the exchange and correlation kernel. This choice should generally be the same as the one used for the potential in the ground state calculation, but it can also be varied, being aware of the fact that in this case possible violations of the sum rule can occur [161]). The other input variable, **nband**, specifies the number of states used to construct the matrix Ω, and must be varied in order to achieve convergence of the resulting spectra. A convergence parameter for the non-self-consistent calculation of the required KS states (`tolwfr`) must also be provided. See `doc/tutorial/lesson_tddft.html` for a tutorial on "TDDFT a la Casida" for closed shell systems.

7   Developments concerning speed-up and convergence issues

7.1   The three levels of parallelization of ground-state calculations

For systems involving a large number of atoms (hundreds), each electronic and/or molecular dynamics step can be very time consuming on single-processor computers. Therefore, parallelization of electronic structure code has become one of the main software development tasks in the past decade [162–166]. Meanwhile, the number of processors of supercomputers keeps on increasing, with nowadays tens of thousands processors available on massively parallel supercomputers (see URL: http://www.top500.org).
This issue is strongly linked to the algorithms used (i) to perform the three-dimensional (3D) fast Fourier transformations (FFTs) and (ii) to solve the KS equations. In order to address (i), we choose the 3D FFT implemented by Goedecker *et al.* [167] in ABINIT due to its efficiency on massively parallel supercomputers. For (ii), since the band-by-band conjugate gradient (CG) proposed by Teter *et al.* [168] does not fit in the requirements of parallelism, we have implemented the multi-band eigensolver LOBPCG proposed by Knyazev [169,170]. This latter is well suited to address the issue of large-scale and complex systems. On the one hand, regarding its performances, this algorithm can be parallelized and scales linearly over hundreds processors. On the other hand, about its robustness, LOBPCG can overtake CG, especially when we increase the size of the block used during the iterative minimization(See Ref. [170]).
The scheme of parallelization, as implemented in ABINIT, is based on three levels. First, all the quantities expanded over the PW basis set (wavefunctions, densities, potentials, etc.) are distributed over a two-dimensional virtual grid of processors. As showed previously (see Ref. [170]), by distributing the workload over two levels of parallelization rather than one (the FFT or band level) we improve the efficiency of the code. In Fig.7, we show the scaling obtained by using the two levels together. In this case, norm-conserving and PAW calculations scales lineary up to 200 and 100 processors, respectively. Finally, a third level of parallelization is added, the **k**-point one, already introduced in ABINIT a long time ago [5]. The scaling is perfect along this dimension, as expected. By demonstrating the feasibility of such a coupling, we are now able to cover all the possible situations met in the framework of *ab initio* simulations: large numbers of **k**-points, $\{\mathbf{G}\}$ vectors and/or bands.

As concerns the compilation, the user have to fill various fields along the configure process. The compilation of the parallel version is activated by setting **enable-mpi="yes"**. If the MPI-IO and/or the SCALAPACK libraries are available on the platform, set also **enable-mpi-io** and/or **enable-scalapack="yes"**,
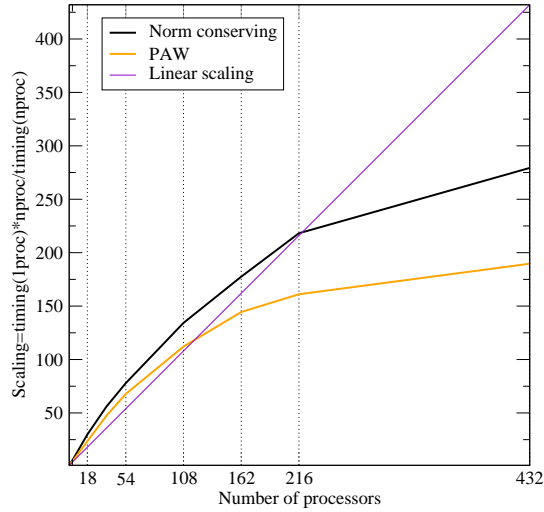
Fig. 7. Two-level parallelization: Speedup of the abinit code with respect to the number of the processors.

respectively. We also strongly advise the user to link the parallel version to the libraries (LAPACK, BLAS and SCALAPACK) distributed for the supercomputer; in order to benefit of all the platform-dependent optimizations. Concerning the input variables in the input file of ABINIT, the user have to indicate:

- **paral_kgb**=1: activate the triple "kgb" parallelization.
- **npband**=m, **npfft**=p and **npkpt**=k: numbers of band, FFT and kpt processors.
- **wfoptalg**=14,4: LOBPCG with or without optimizations.
- **fft_opt_lob**=2: generalization of the transposition principle.
- **fftalg**=401: the 3dim parallel FFT of Goedecker *et al.* [167].
- **bandpp**=2: advised for **istwfk**=2 (or to improve the convergence).

For real applications using LOBPCG with the triple "kgb parallelization", the user can refer to recent studies (see Ref. [171,172]) and also follow the examples included in the `tests/paral/` test suite (`tP.in`, `tQ.in`, `tR.in`, `tS.in`, `tW.in`, `tX.in` and `tY.in`).

7.2 Parallelism of the GW calculations

Many-body calculations have CPU and memory requirements much larger than those needed for standard ground-state applications. Therefore an optimized implementation able to take advantage of symmetries as well as of a

parallel environment supporting the MPI protocol is of fundamental importance.

Symmetry properties, including Shubnikov anti-ferromagnetic symmetries of type IV [173], are extensively employed in each phase of the GW calculation in order to reduce the computing effort and the memory consumption. The number of independent matrix elements needed to calculate the irreducible polarizability (`optdriver=3`) can be significantly reduced by taking advantage of time-reversal invariance (**awtr**=1), as well as space group symmetries (**symchi**=1). An analogous computing saving can be achieved during the calculation of the QP corrections (`optdriver=4`) by setting **symsigma** to 1. Input files exemplifying these features can be found in the tests `t63.in`, `t64.in`, `t68.in`, and `t69.in` located in the directory `tests/v5/`.

For what concerns the MPI parallelization, two different parallel algorithms are available. The choice of the particular algorithm to use is governed by the input variable **gwpara**. The value **gwpara**=1 refers to the standard parallelism over **k**-points commonly employed in many *ab initio* codes. The **k**-point parallelism is characterized by a very small overload due to MPI communications, is very efficient in the case of metallic compounds, but it is of no help in the case of systems sampled only at the Γ point. More importantly, **gwpara**=1 does not lead to any reduction of the memory requirement since each node has to store in memory the entire set of bands (both empty and occupied) used to evaluate the Green's functions.

To tackle large systems, ABINIT provides a different parallel algorithm in which the calculation of $\tilde{\chi}$ and Σ is parallelized over bands (**gwpara**=2). This particular distribution –which strongly differs from the standard approach used in the other parts of ABINIT– allows both CPU and memory saving since each processor has to calculate only the partial contribution to $\tilde{\chi}$ or Σ arising from a subset of energy states. The GW band parallelism presents good scalability both concerning CPU time and memory requirements, since an increasing number of empty states can be compensated for by just using more processors. The only limiting factor is represented by the number of bands used to evaluate the Greens' function (**nband**) which is usually of the order of hundreds or even more. Exemplifying automatic tests are provided in several input files: the **k**-point parallelism is tested in `paral/tM.in`, and `paral/tO.in` while the input files `paral/tN.in`, `paral/tU.in`, and `paral/tV.in` show how to run a GW calculations parallelized over bands.

Despite the intensive coarse grained MPI parallelization of the GW part, QP calculations are still challenging, especially in the case of systems with a large unit cell. The most important memory bottlenecks are represented by the storage of the wavefunctions in real space and of the two-point function $\varepsilon_{\mathbf{G}1,\mathbf{G}_2}^{-1}(\mathbf{q},\omega)$ required to construct W. It is worth stressing that, in the present

implementation, the memory consumption due to $\varepsilon^{-1}_{\mathbf{G}1,\mathbf{G}_2}(\mathbf{q},\omega)$ does not scale with respect to the number of processing units. This might render GW calculations on memory-distributed clusters with a relatively small amount of RAM per node unfeasible. The memory consumption can, however, be significantly reduced (at the price of worsening performance) through the input variable **gwmem** which consists of two separated digits. The first digit relates to the memory strategy to be used for ε^{-1}, while the second digit defines how the storage of the wavefunctions should be managed. The storage in memory of the full ε^{-1} can be partly avoided by opting for an out-of-core solution in which a single **q**-point is stored in the central memory while the entire two-point function is stored on disk (**gwmem**=0x). Also the memory needed to store the wavefunctions can be significantly decreased by using **gwmem**=x0. In this case, only the wavefunctions in **G**-space are stored in memory, and an FFT from **G** to **r** is performed whenever a particular wavefunction is required. Example files showing how to use these features of the implementation are provided in `tests/v5/t65.in`, and `tests/v5/t70.in`.

For systems requiring a dense sampling of the Brillouin-zone and, therefore, a large number of **q**-point in $\varepsilon^{-1}_{\mathbf{G}1,\mathbf{G}_2}(\mathbf{q},\omega)$, ABINIT provides the possibility of splitting the calculation of the full dielectric matrices into smaller jobs that can be run independently. The partial matrices generated in the different calculations can then be gathered in a unique binary file by means of the `mrgscr` utility. The two variables **nqptdm** and **qptdm** are used to define the number and the reduced coordinates of the subset of **q**-points, respectively. The example input files `t87.in`, `t88.in`, and `t89.in` in the directory `tests/v3/` show how to split a screening calculation then merging the final results.

7.3 The extrapolar method for self-consistent cycles

Background: The most common and stable algorithm for solving DFT problems consists in a two step eigenvalue search. For a given Hamiltonian, one seeks the firsts eigenvectors (bands), then the resulting bands are used to redefine a new Hamiltonian; this process being performed until the bands (or the resulting Hartree potential, or the associated energy, ...) stop changing. A critical step in this algorithm is the construction of the new Hamiltonian. It can be non convergent and then prevent reaching the system ground state. This problem being all the more stringent as system size increases and band gap decreases.

In systems with homogeneous dielectric behaviour, the Kerker's [174] preconditioner can solve efficiently this problem. For the less trivial inhomogeneous case, an universal method [175], derived from Newton-Raphson algorithm, was derived, long ago, to change a diverging system into a converging one. Yet it

yields an unfavorable scaling with system size together with a large prefactor that makes it virtually unusable on a plane-wave basis for systems with more than about ten atoms. The extrapolar [176] method is an approximate of this method that allows a dramatic reduction of the prefactor while almost preserving its perfect preconditioner behaviour.

Description and variables: Applying a preconditioner to the DFT problem can be resumed in the following steps: evaluation of the susceptibility matrix χ_0, from which the dielectric matrix and its inverse is deduced; and then multiplying the potentials residuals with the inverse dielectric matrix. The evaluation of χ_0 is the computationally heavy part of this process since

$$\chi_0(\mathbf{r}, \mathbf{r}') = \sum_k \sum_{n=1}^{n_{valence}} \sum_{n'=n_{valence}+1}^{\infty} 2 \frac{\psi_{kn}^*(\mathbf{r})\psi_{kn'}(\mathbf{r})\psi_{kn'}^*(\mathbf{r}')\psi_{kn}(\mathbf{r}')}{E_{kn'} - E_{kn}}. \tag{63}$$

Where $\mathbf{r}$ stands for spatial coordinates; k are sampling points of reciprocal space; ψ_{kn} are bands; n is the number attributed to bands ordered by eigenvalues starting from the lowest; and E_{kn} is the eigenvalue associated to a band.

The extrapolar method consist in approximating this calculation with the following formula:

$$\tilde{\chi_0}(\mathbf{r}, \mathbf{r}') = \sum_k \sum_{n=1}^{n_{valence}} \sum_{n'=n_{valence}+1}^{n_{band}} 2 \frac{\psi_{kn}^*(\mathbf{r})\psi_{kn'}(\mathbf{r})\psi_{kn'}^*(\mathbf{r}')\psi_{kn}(\mathbf{r}')}{E_{kn'} - E_{kn}}$$
$$+ 2 \sum_k \sum_{n=1}^{n_{valence}} \frac{\Lambda}{\bar{E} - E_{kn}} \times \left[\rho_n(r)\delta(\mathbf{r} - \mathbf{r}') - \psi_{kn}^*(\mathbf{r})\psi_{kn}(\mathbf{r}') \sum_{n'=1}^{n_{band}} \psi_{kn'}(\mathbf{r})\psi_{kn'}^*(\mathbf{r}') \right]. \tag{64}$$

Where $n_{valence}$ is the number of occupied bands; n_{band} is the total number of calculated bands; ρ_n is the density associated to a band; and Λ and $\bar{E}$ are the parameters used to tune the approximation. Within Abinit those parameters are associated with the input variables **dielam** and **diegap**.

Besides, since usually only the largest wavelengths are associated with divergence of the SCF-cycle, the computation of the preconditioner is limited to a small part of the plane-wave basis set. The cutoff energy used to defined this sub-basis is called **diecut**.

The default values for **dielam**, **diegap** and **diecut** are respectively: 0.5, 0.1 Hartree and 2.2 Hartree. Our tests on various materials and configuration shows that those values are quite safe to use in any situation. However for
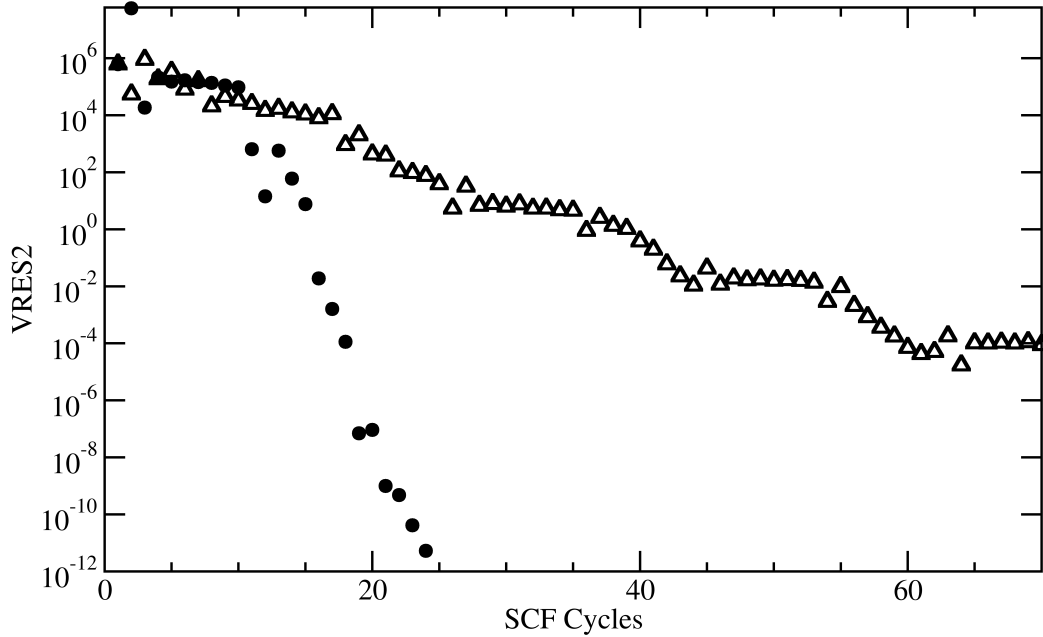
Fig. 8. Square norm of the potential residuals (vres2) as a function of the SCF step on an infinite slab of α-silver, cut along [110], wide of 18 atoms separated by about 32 bohrs. The triangles and the plain circles depict evolution of vres2 using respectively Kerker's method or extrapolar for preconditioner.

the largest systems it can be useful and almost not detrimental to reduce the value of **diecut**.

Since the dielectric behaviour of a system from the very first random Hamiltonian to the converged one hardly changes, it is not necessary to evaluate the preconditioner at each cycle. The variable used to choose the preconditioner in Abinit is **iprcel**. Choosing a value from 41 to 49 will make abinit evaluate the extrapolar preconditioner at the first step and then at $(\mathbf{iprcel}-40)^{th}$ step. Choosing a value between 141 and 149 will make abinit compute the preconditioner every $(\mathbf{iprcel}-140)$ step, starting at the first one. This last option is better reserved for the most difficult convergence problem. For most system **iprcel** 45+ is very convenient.

Fig. 8 shows the effect of the extrapolar method on a typical ill-conditioned DFT problem. Both calculations use default values for the mixing algorithm and preconditioning. For extrapolar the inverse dielectric matrix is evaluated at step one and five while Kerker's method is by default set up for highly conductive metals.

Automatic tests:

The extrapolar method is used within many of the Abinit self-tests. Namely in `tests/v1/Input/tXX.in` where XX stands for 51, 53 to 55, 57, 71 to 73; `tests/v2/Input/t05.in` and `tests/v2/Input/t08.in`; `tests/v4/Input/t46.in`;

71

in `/tests/tutorial/Input`, `t47.in` and `t48.in`; and in `/tests/paral/Input`, `tD.in` and `tW.in`. Among them some are specifically dedicated to test extrapolar. Those uses surfaces of molybdenum and perform a simple preconditioned ground state calculation. Otherwise extrapolar is used for improving convergence rate of ground state calculation required by tests that relies on converged ground state properties.

7.4 The extrapolar method for GW calculations

As explained in Sec. 6.1, the GW calculations are computationally demanding. This originates in particular from the many empty states required to achieve the convergence. Typically several hundred empty states are necessary, depending on the system. The extrapolar method exposed in the previous paragraph looks appealing in order to eliminate most of the empty states.

In the framework of the GW calculations, the extrapolar method can be used to decrease the number of empty states both in the independent-particle response function and in the GW self-energy, as proposed by Bruneval and Gonze [177]. The implementation of the extrapolar method within GW introduces only one single parameter: the average high-energy $\bar{E}$. The corresponding weight Λ has been simply set to one. The input parameter **gwcomp** governs the use of the extrapolar method when performing a screening calculation (`optdriver=3`) or a self-energy calculation (`optdriver=4`). The energy $\bar{E}$ is set by the input variable **gwencomp**. Its value is referenced with respect to the highest band energy included in the calculation. In all the cases studied so far, a carefully chosen value of $\bar{E}$ has always permitted for a dramatic reduction of the number of empty states.

The problem of choosing the right value for the parameter $\bar{E}$ can be addressed by finding a constraint. Indeed, we know from Ref. [136] that the independent-particle response function satisfies a sum-rule:

$$\int_0^{+\infty} d\omega \omega \frac{4\pi}{|\mathbf{q}+\mathbf{G}|^2} \Im \chi_{0\mathbf{G}\mathbf{G}}(\mathbf{q},\omega) = -\frac{\pi}{2}\omega_p^2. \tag{65}$$

When truncating the number of empty states, the resulting response function χ_0 violates this sum-rule. When adding the extrapolar correction, one can try to recover the fulfillment of the sum-rule, even with a limited number of empty states. This constitutes a guideline to determine an "optimal" value for $\bar{E}$. The value of the integral in the left-hand side of Eq. 65 is calculated by ABINIT during a screening run (`optdriver=3`), so it can be monitored by the user.

The extrapolar method in the GW context is exemplified in the automatic tests `tests/v5/t65.in`, `tests/v5/t69.in`, and `tests/paral/tV.in`.

8 Summary

The present paper had the aim to describe ABINIT 5.7.4, in view of its use by newcomers as well as more experienced users. The different capabilities of ABINIT have been reviewed, with, for each capability, an adequate entry point to more information, to be found either in the package itself (existing tutorials, for capabilities developed before 2005) or in the present paper (for capabilities developed after 2005). In all cases, example input files are present in the package, together with reference output files (more than 600 test cases are present). The present paper also give some information on the structure of the main ABINIT program, the structure of the whole package, and describes how to build and to run ABINIT.

9 Acknowledgments

References

[1] P. Hohenberg, W. Kohn, Phys. Rev. 136 (1964) 864.

[2] W. Kohn, L. Sham, Phys. Rev. A 140 (1965) 1133.

[3] L. Hedin, Phys. Rev. 139 (1965) A796.

[4] R.M. Martin, Electronic Structure. Basic Theory and Practical Methods, Cambridge University Press, Cambridge, U.K., 2004.

[5] X. Gonze, J.-M. Beuken, R. Caracas, F. Detraux, M. Fuchs, G.-M. Rignanese, L. Sindic, M. Verstraete, G. Zerah, F. Jollet, M. Torrent, A. Roy, M. Mikami, Ph. Ghosez, J.-Y. Raty, D. C. Allan, Comput. Mater. Sci. 25 (2002) 478.

[6] X. Gonze, G.-M. Rignanese, M. Verstraete, J.-M. Beuken, Y. Pouillon, R. Caracas, F. Jollet, M. Torrent, G. Zerah, M. Mikami, Ph. Ghosez, M. Veithen, J.-Y. Raty, V. Olevano, F. Bruneval, L. Reining, R. Godby, G. Onida, D.R. Hamann, D.C. Allan, Z. Kristallogr. 220 (2005) 558.

[7] G.-M. Rignanese, J.-P. Michenaud and X. Gonze, Phys. Rev. B. 53 (1996) 4488.

[8] F. Detraux and X. Gonze, Phys. Rev. B 63 (2001) 115118.

[9] P.E. Blöchl, Phys. Rev. B 50 (1994) 17953.

[10] M. Torrent, F. Jollet, F. Bottin, G. Zerah, X. Gonze, Comput. Mater. Sci. 42 (2008) 337.

[11] A. R. Tackett, N. A. W. Holzwarth, G. E. Matthews, Comput. Phys. Commun. 135 (2001) 329. http://pwpaw.wfu.edu.

[12] N. A. W. Holzwarth, G. E. Matthews, A. R. Tackett, R. B. Dunning, Phys. Rev. B 57 (1998) 11827.

[13] http://www.physics.rutgers.edu/ dhv/uspp.

[14] D. Vanderbilt, Phys. Rev. B 41 (Rapid Communications) (1990) 7892.

[15] B. Amadon, F. Lechermann, A. Georges, F. Jollet, Phys. Rev. B 77 (2008) 205112.

[16] C. Audouze, F. Jollet, M. Torrent, X. Gonze, Phys. Rev. B 73 (2006) 235101.

[17] C. Audouze, F. Jollet, M. Torrent, X. Gonze, Phys. Rev. B 78 (2008) 035105.

[18] V. I. Anisimov, J. Zaanen, O. K. Andersen, Phys. Rev. B 44 (1991) 943.

[19] V. I. Anisimov, F. Aryasetiawan, A. I. Lichtenstein, J. Phys.: Cond. Matt. 9 (1997) 767.

[20] A. I. Liechtenstein, V. I. Anisimov, J. Zaanen, Phys. Rev. B 52 (1995) R5467.

[21] M. T. Czyżyk, G. A. Sawatzky, Phys. Rev. B 49 (1994) 14211.

[22] B. Amadon, F. Jollet, M. Torrent, Phys. Rev. B 77 (2008) 155104.

[23] O.Bengone, M. Alouani, P. Blöchl, J. Hugel, Phys. Rev. B 62 (2000) 16392.

[24] G. Jomard, B. Amadon, F. Bottin, M. Torrent, Phys. Rev. B 78 (2008) 075125.

[25] P. Novák, J. Kunes, L. Chaput, W.E. Pickett, Phys. Stat. Sol. (b) (2006) 563.

[26] M. Ernzerhof, G.E. Scuseria, J. Chem. Phys. (1999) 5029.

[27] J.P. Perdew, K. Burke, M. Ernzerhof, Phys. Rev. Lett. (1996) 3865.

[28] F. Tran, P. Blaha, K. Schwarz, P. Novák, Phys. Rev. B,74(2006) 155108.

[29] J. Paier, R. Hirschl, M. Marsman, G. Kresse, J. Chem. Phys. (2005) 234102.

[30] R. Resta, Rev. Mod. Phys. 66 (1994) 899.

[31] D. Vanderbilt, R. D. King-Smith, Phys. Rev. B 48 (1993) 4442.

[32] R. D. King-Smith, D. Vanderbilt, Phys. Rev. B 47 (1993) 1651.

[33] D. Vanderbilt, R. D. King-Smith, available at http://xxx.lanl.gov/ps/cond-mat/9801177 (1998).

[34] A. Ferretti, A. Calzolari, B. Bonferroni, R. Di Felice, J. Phys.: Condens. Matter 19 (2007) 036215.

[35] W. A. Harrison, Solid State Theory, Mc Graw-Hill, 1970.

[36] G. D. Mahan, Many Particle Physics, Plenum Press, New York, 1981.

[37] J. Callaway, Quantum Theory of the solid state, Academic Press New York, 1974.

[38] L. Collins, S. Bickham, J. Kress, S. Mazevet, T. Lenosky, W. Windl, Phys. Rev. B 63 (2001) 184110.

[39] S. Mazevet, M. Torrent, V. Recoules, F. Jollet, High En. Dens. Phys. journal (2009) in press.

[40] M. Taillefumier, D. Cabaret, A. N. Flank, F. Mauri, Phys. Rev. B 66 (2002) 195107.

[41] S. Mazevet, G. Zerah, Phys. Rev. Lett. 101 (2008) 155001.

[42] H. M. Petrilli, P. E. Blöchl, P. Blaha, K. Schwarz, Phys. Rev. B 57 (1998) 14690.

[43] M. Profeta, F. Mauri, C. J. Pickard, J. Am. Chem. Soc. 125 (2003) 541.

[44] J. W. Zwanziger, M. Torrent, Appl. Magn. Reson. 33 (2008) 4477.

[45] J. W. Zwanziger, J. Phys. Conden. Matt. 21 (2009) 195501.

[46] C. J. Pickard, F. Mauri, Phys. Rev. B 63 (2001) 245101.

[47] J. W. Zwanziger, Phys. Rev. B 79 (2009) 033112.

[48] R.M. Nieminen, E. Boroński, L. Lantto, Phys. Rev. B 32 (1985) 1377.

[49] E. Boroński, R.M. Nieminen, Phys. Rev. B 34 (1986) 3820.

[50] M.J. Puska, R.M. Nieminen, Rev. Mod. Phys. 66 (1994) 841.

[51] L. Giglien, G. Galli, F. Gygi, R. Car, Phys. Rev. Lett. 72 (1994) 3214.

[52] M.J. Puska, A.P. Seitsonen, R.M. Nieminen, Phys. Rev. B 52 (1995) 10947.

[53] J. Arponen, E. Pajanne, Ann. Phys. (N.Y.) 121 (1979) 343.

[54] I. Makkonen, M. Hakala, M.J. Puska, Phys. Rev. B 73 (2006) 035103.

[55] A. Seeger, F. Banhart, phys. stat. sol. (a) 102 (1987) 171.

[56] P.A. Sterne, J.H. Kaiser, Phys. Rev. B 43 (1991) 13892.

[57] T. Korhonen, M.J. Puska, R.M. Nieminen, Phys. Rev. B 54 (1996) 15016.

[58] A. Rubaszek, Z. Szotek, W.M. Temmerman, Phys. Rev. B 58 (1998) 11285.

[59] H. Takenaka, D.J. Singh, Phys. Rev. B 77 (2008) 155132.

[60] I. Daubechies, Ten Lectures on Wavelets, SIAM, Philadelphia, 1992.

[61] L. Genovese, A. Neelov, S. Goedecker, T. Deutsch, S. A. Ghasemi, A. Willand, D. Caliste, O. Zilberberg, M. Rayson, A. Bergman, R. Schneider, J. Chem. Phys 129 (2008) 014109, http://inac.cea.fr/sp2m/L_Sim/BigDFT.

[62] S. Goedecker, M. Teter, J. Hutter, Phys. Rev. B 54 (1996) 1703.

[63] C. Hartwigsen, S. Goedecker, J. Hutter, Phys. Rev. B 58 (1998) 3641.

[64] M. Krack, Theor. Chem. Acc. 114 (2005) 145.

[65] G. Deslauriers, S. Dubuc, Constructive Approx. 5 (1989) 49.

[66] A. I. Neelov, S. Goedecker, J. Comp. Phys. 217 (2006) 312-339.

[67] M. Payne, M. Teter, D. Allan, T. Arias, J. Joannopoulos, Rev. of Mod. Phys. 64 (1992) 1045.

[68] L. Genovese, T. Deutsch, A. Neelov, S. Goedecker, G. Beylkin, J. Chem. Phys. 125 (2006) 074105.

[69] L. Genovese, T. Deutsch, S. Goedecker, J. Chem. Phys. 127 (2007) 054704.

[70] E. Lorin, G. Zerah, Comp. Phys. Comm. 158 (2004) 39.

[71] R. Haydock, Solid State Phys. 35 (1980) 215.

[72] C. Lanczos, J. Res. Natl. Bur. Stand 45 (1950) 255.

[73] S. Le Roux, G Zerah, J. Comp Phys. 226 (2007) 2063-2077.

[74] G. H. Wannier, Phys. Rev. 52 (1937) 191.

[75] S. F. Boys, Rev. Mod. Phys 32 (1960) 296; J. M. Foster, S. F. Boys, Rev. Mod. Phys 32 (1960) 300.

[76] N. Marzari, D. Vanderbilt, Phys. Rev. B 56 (1997) 12847.

[77] Ivo Souza, N. Marzari, D. Vanderbilt, Phys. Rev. B 65 (2001) 035109.

[78] A. Castro, H. Appel, M. Oliveira, C.A. Rozzi, X. Andrade, F. Lorenzen, M.A.L. Marques, E.K.U. Gross, and A. Rubio, Phys. Stat. Sol. B 243 (2006) 2465. M.A.L. Marques, A. Castro, G. F. Bertsch, and A. Rubio, Comput. Phys. Commun. 151 (2003) 60. http://www.tddft.org/programs/octopus

[79] J.P. Perdew, S. Kurth, in: A primer in Density Functional Theory, C. Fiolhais, F. Nogueira, and M.A.L. Marques (eds.) Springer, Berlin, 2003 p. 1.

[80] X. Gonze, C.-O. Almbladh, A. Cucca, D. Caliste, C. Freysoldt, M. Marques, V. Olevano, Y. Pouillon, M.J. Verstraete, Comp. Mat. Sc. 43 (2008) 1056.

[81] H. Rojas, R. Godby, R. Needs Phys. Rev. Lett. 74 (1995) 1827.

[82] A. Marini, C. Hogan, M. Grüning, D. Varsano Comp. Phys. Comm. (2009) in press.

[83] R. Rew, G. Davis, IEEE Computer Graphics and Applications 10 (1990) 76.

[84] URL http://www.unidata.ucar.edu/software/netcdf/.

[85] D. Caliste, Y. Pouillon, M.J. Verstraete, V. Olevano, X. Gonze, Comp. Phys. Com. 179 (2008) 748.

[86] URL http://theory.polytechnique.fr/codes/dp/.

[87] URL http://www.bethe-salpeter.org/.

[88] URL http://uszla.me.uk/space/software/FoX/.

[89] URL http://lcdx00.wm.lc.ehu.es/ag/xml/.

[90] A.A. Mostofi, J.R. Yates, Y.-S. Lee, I. Souza, D. Vanderbilt, N. Marzari, Comput. Phys. Commun. 178 (2008) 685.

[91] G. Grimvall, The electron phonon interaction in metals, North-Holland, Amsterdam, 1981.

[92] P. B. Allen, B. Mitrović, Theory of Superconducting T_c, in: Solid State Phys., Vol. 37, H. Ehrenreich, F. Seitz, D. Turnbull (eds.), Academic Press, New York, 1982.

[93] S. Y. Savrasov, D. Y. Savrasov, Phys. Rev. B 54 (1996) 16487.

[94] F. Giustino, J.R. Yates, I. Souza, M.L. Cohen, S.G. Louie, Phys. Rev. Lett. 98 (2007) 047005.

[95] P. E. Blöchl, O. Jepsen, O. K. Andersen, Phys. Rev. B 49 (1994) 16223.

[96] M. J. Verstraete, X. Gonze, Phys. Rev. B 74 (2006) 153408.

[97] P. Sen, Chem. Phys. Lett. 428 (2006) 430.

[98] E. Antoncik, Czechosl. Journ. Phys. 5 (1955) 4.

[99] H. Y. Fan, Physical Review 82 (1951) 900.

[100] P.B. Allen, V. Heine, J. Phys. C 9 (1976) 2305.

[101] S. Gopalan, P. Laustenschlager, M. Cardona, Physical Review B 35 (1987) 5577.

[102] P.B. Allen, M. Cardona, Physical Review B 27 (1983) 4760.

[103] M. Cardona, Solid State Communications 133 (2005) 3-18.

[104] X. Gonze, Physical Review A 52 (1995) 1096.

[105] M. Veithen, X. Gonze, Ph. Ghosez, Phys. Rev. B, 71 (2005) 107.

[106] M. Cardona, Light Scattering in Solids II: Basic Concepts and Instrumentation, vol. 50, in: Topics in Applied Physics, M. Cardona, G. Guüntherodt (eds.), Springer, New York, 1982, p. 19-168.

[107] S.A. Prosandeev, U. Waghmare, I. Levin, J. Maslar, Phys. Rev. B 71 (2005) 214307.

[108] G. Placzek, in: Handbuch der Radiologie Vol. 6, Akademische Verlagsgeselschaft, Leipzig, 1934, p. 208.

[109] W. Hayes, R. Loudon, Scattering of Light by Crystals, Wiley, New York, 1978, p. 113.

[110] H. Hamaguchi, Adv. Infrared Raman Spectrosc. 12 (1985) 273.

[111] R. Caracas, R.E. Cohen, Geophys. Res. Lett. 33 (2006) L12S05.

[112] P. Hermet, M. Veithen, Ph. Ghosez, J. Phys. : Condens. Matter 19 (2007) 456202.

[113] P. Hermet, M. Veithen, Ph. Ghosez, J. Phys. Condens. Matter (2009) in press.

[114] R. Caracas E.J. Banigan, Phys. Earth Planet. Inter., (2009) in press.

[115] R. Caracas, R.E. Cohen, Phys. Rev. B, 76 (2007) 184101.

[116] P. Hermet, N. Izard, A. Rahmani, Ph. Ghosez, J. Phys. Chem. B 110 (2006) 24869.

[117] R. Caracas, Phys. Rev. Lett., 101 (2008) 085502.

[118] R. Caracas, J. Chem. Phys. 127 (2007) 144510.

[119] A freely-accessible web-based database of computed Raman spectra for minerals, available at http://www.wurm.info.

[120] P. Hermet, M. Goffinet, J. Kreisel, Ph. Ghosez, Phys. Rev. B 75 (2007) R220102.

[121] P. Hermet, M. Goffinet, Ph. Ghosez, J. Appl. Phys. 105 (2009) 036108.

[122] R. Caracas, R.E. Cohen, Appl. Phys. Lett., 91 (2007) 092902.

[123] M. Veithen, X. Gonze, Ph. Ghosez, Phys. Rev. Lett. 93 (2004) 187401.

[124] M. Veithen, Ph. Ghosez, Phys. Rev. B 71 (2005) 132101.

[125] A.L. Fetter, J.D. Walecka, Quantum Theory of Many-Particle Systems, McGraw-Hill, New York, 1971.

[126] A.A. Abrikosov, L.P. Gorkov, E. Dzyaloshinskii, Methods of quantum field theory in statistical physics, Dover, New York, 1975.

[127] G. Onida, L. Reining, A. Rubio, Rev. Mod. Phys. 74 (2002) 601.

[128] B. Holm, U. von Barth, Phys. Rev. B, 57 (1998) 2108.

[129] W.-D. Schöne, A.G Eguiluz, Adolfo, Phys. Rev. Lett. 81 (1998) 1662.

[130] M. S. Hybertsen, S. G. Louie, Phys. Rev. B 34 (1986) 5390.

[131] J-L. Li, G.-M. Rignanese, S.G Louie, Phys. Rev. B, 71 (2005) 193102.

[132] F. Bruneval, N. Vast, L. Reining, Phys. Rev. B 74 (2006) 045102.

[133] R. W. Godby, R. J. Needs, Phys. Rev. Lett. 62 (1989) 1169.

[134] N.W. Ashcroft, N.D Mermin, Solid State Physics, Cornell University, 1976.

[135] D. L. Johnson, Phys. Rev. B 9 (1974) 4475.

[136] M. Taut, J. Phys. C 18 (1985) 2677.

[137] W. von der Linden, P. Horsch, Phys. Rev. B 37 (1988) 8351.

[138] G. E. Engel, B. Farid, Phys. Rev. B 47 (1993) 15931.

[139] W. G. Aulbur, L. Jonsson, J. W. Wilkins, Solid State Phys. 54 (2000) 1.

[140] B.I. Lundqvist, Phys. Kondens. Materie 7 (1968) 117.

[141] S. V. Faleev, M. van Schilfgaarde, T. Kotani, Phys. Rev. Lett. 93 (2004) 126406.

[142] M. van Schilfgaarde, T. Kotani, S.V. Faleev, Phys. Rev. Lett. 96 (2006) 226402.

[143] T. Kotani, M. van Schilfgaarde, S.V. Faleev, Phys. Rev. B 76 (2007) 165106.

[144] T. Kotani, M. van Schilfgaarde, Solid State Communications, 121 (2002) 461.

[145] R. Gómez-Abal, X. Li, M. Scheffler, C. Ambrosch-Draxl, Phys. Rev. Lett. 101 (2008) 106404.

[146] A. Marini, G. Onida, R. Del Sole, Phys. Rev. Lett. 88 (2001) 016403.

[147] M. L. Tiago, S. Ismail-Beigi, S. G. Louie, Phys. Rev. B 69 (2004) 125212.

[148] F. Mauri, B.G Pfrommer, S.G Louie, Phys. Rev. Lett. 77 (1996) 5300.

[149] B. Arnaud, M. Alouani, Phys. Rev. B 62 (2000) 4464.

[150] The 3-j and 6-j symbols, Technology Press, M. Rotenber, R. Bivins, N. Metropolis, J.K Wooten. Massachusetts Institute of Technology, Cambridge, 1959.

[151] M.R. Pick, H.C. Morrel, R.M Martin, Phys. Rev. B, 1 (1970) 910.

[152] S. Baroni, R. Resta, Phys. Rev. B, 33 (1986) 7017.

[153] M.S. Hybertsen, S.G Louie, Phys. Rev. B, 35 (1986) 5585.

[154] J. R. Yates, X. Wang, D. Vanderbilt, I. Souza. Phys. Rev. B 75 (2007) 195121.

[155] D. R. Hamann, D. Vanderbilt, Phys. Rev. B 79 (2009) 045109.

[156] E. Runge, E.K.U. Gross, Phys. Rev. Lett. 52 (1984) 997.

[157] G. Onida, L. Reining, A. Rubio, Rev. of Mod. Phys. 74 (2002) 601.

[158] M. E. Casida, in: Recent Development and Applications of Modern Density Functional Theory, J.M. Seminario (ed.) Elsevier, Amsterdam, 1996, p. 391.

[159] M.E. Casida, A. Ipatov, F. Cordova, in: Time-Dependent Density-Functional Theory, M.A.L. Marques, C. Ullrich, F. Nogueira, A. Rubio, E.K.U. Gross (eds.), Lecture Notes in Physics, Springer: Berlin, 2006, pp. 243-257.

[160] M.E. Casida, J.Chem.Phys. 122 (2005) 054111.

[161] B. Farid, Philos. Mag. Lett. 79 (1999) 581.

[162] R. V. Vadali, Y. Shi, S. Kumar, L. V. Kale, M. E. Tuckerman, G. J. Martyna, J. Comp. Chem. 25 (2004) 2006.

[163] J. Hutter, A. Curioni, Chem. Phys. Chem. 6 (2005) 1788.

[164] J. Hutter, A. Curioni, Parallel Computing 31 (2005) 1.

[165] C. K. Skylaris, P. Haynes, A. A. Mostofi, M. C. Payne, Phys. Stat. Sol. 243 (2006) 973.

[166] C. K. Skylaris, P. Haynes, A. A. Mostofi, M. C. Payne, J. Chem. Phys. 122 (2005) 084119.

[167] S. Goedecker, M. Boulet, T. Deutsch, Comput. Phys. Comm. 154 (2003) 105.

[168] M. P. Teter, M. C. Payne, D. C. Allan, Phys. Rev. B 40 (1989) 12255.

[169] A. Knyazev, SIAM J. Sci. Computing 23 (2001) 517.

[170] F. Bottin, S. Leroux, A. Knyazev, G. Zérah, Comput. Mater. Sci. 42 (2008) 329.

[171] S. Mazevet, F. Lambert, F. Bottin, G. Zérah, J. Clérouin, Phys. Rev. E 75 (2007) 056404.

[172] F. Bottin, G. Zérah, Phys. Rev. B 75 (2007) 174114.

[173] C.J. Bradley, A.P. Cracknell, The Mathematical Theory of Symmetry in Solids, Representation Theory for Point Groups and Space Groups, Clarendon Press, Oxford 1972.

[174] G.P. Kerker, Phys. Rev. B 23 (1981) 3082.

[175] K.-M. Ho, J. Ihm, J.D. Joannopoulos, Phys. Rev. B 25 (1982) 4260.

[176] P.-M. Anglade, X. Gonze, Phys. Rev. B 78 (2008) 045126.

[177] F. Bruneval, X. Gonze, Phys. Rev. B 78 (2008) 085125.