Yambo:
a quick guided tour

version 1.0
the yambo team
www.yambo-code.org

This document introduces the main functionalities of the GPL version of yambo, a computer program based on Many-Body perturbation theory for calculations of electronic and optical properties in Solid State and Molecular Physics.

1 Before Starting:

What is yambo?

yambo is not simply a computer program. yambo aims to be a combination of several projects carried on by people working in different fields of Solid State and Molecular Physics. Each project is devoted to the device of new theoretical and/or numerical tools to study the excited state properties of electronic systems in an *ab-initio* framework.

In practice, the yambo project is a mixed Fortarn/C code composed of a core part and many extensions. These extensions are pieces of code embodied in the main yambo trunk, that can be isolated or joint together. The result is one source but many possible executables.

The core of the yambo code plus some basic excited state features are publicly released under the GPL license. With this version of yambo you can calculate:

⋆ quasiparticle energies within the GW approximation;

⋆ electron loss and optical absorption spectra of solids, and dynamical polarizability of molecules at different level of theory:

  ▷ Random Phase Approximation,
  ▷ Time Dependent Local Density Approximation,
  ▷ Bethe–Salpeter equation.

yambo relies on the Kohn-Sham wavefunctions generated by two Density Functional Theory public codes: abinit, and PWscf.

yambo is a suite of 4 different programs:

– a2y, the interface to abinit
– p2y, the interface to PWscf
– yambo, the main program
– ypp, the post-processing tool

What is needed to run yambo?

To run yambo you need:

• Having installed yambo on your computer **with the NETCDF support**. Note that no instruction regarding the installation are given in this document. Please

refer to the yambo web-page for more informations. For the calculations proposed here a desktop PC is enough.
- Knowing the basic shell commands on UNIX-like operative systems.
- Knowing the theories and approximations behind the program. Since yambo is not a black-box this is essential to get physical meaningful results.

How to use this document?

Through simple examples, this document explains the main functionalities of some of the programs in the yambo suite.
For clarity it is assumed that the yambo executables lies in your PATH.

2 Getting Started:

Preparing the input databases

If you try to run yambo in an arbitrary directory you get the following error message

```
~> yambo
yambo: cannot access CORE database (SAVE/*db1 and/or SAVE/*wf)
```

that tells you that yambo could not find the input databases. These files can be created using the abinit (http://www.abinit.org) and the PWscf (http://www.pwscf.org) codes. In this document we will use a ready-to-use calculation of abinit on a simple silicon bulk case as an example. To find more informations about how to generate the yambo input databases for your system refer to the yambo web-page (http://www.yambo-code.org/).

First you need to download the archive

```
http://www.yambo-code.org/doc/example.tgz
```

and to follow the instructions written in the README file to generate the yambo databases. Note that you need NETCDF binaries to read these database . If you do know what NETCDF is, please visit the http://www.unidata.ucar.edu/software/netcdf/ web-page. In a few words NetCDF (network Common Data Form) is a set of software libraries to support the creation, access, and sharing of array-oriented scientific data. The key property of the NETCDF databases is that they are platform-independent: you can create them on a Linux machine and then read them on an SGI/HP or whatever you want.
As a result you should have now the SAVE/ directory containing the databases:

- s.wf (or ns.wf when NETCDF is used) containing the wavefunctions,
- s.db1 (or ns.db1 when NETCDF is used), containing the lattice parameters, k-points, eigenvalues, symmetries, etc.

In general all the data's needed by the program are stored in `SAVE/` as `db.*` files. The command `yambo -D` reads the header of the databases in the `SAVE/` directory containing information on the parameters of the calculation that generated the databases.

Generating the input of yambo

`yambo` does not need precompiled input files, it works like a standard unix command and the possible action to be taken is selected using the command line options. To see its options you can invoke the long help:

```
> yambo -H
 -h              :Short Help
 -H              :Long Help
 -J <opt>        :Job string identifier
 -V <int>        :Input file verbosity
 -F <opt>        :Input file
 -I <opt>        :Core I/O directory
 -O <opt>        :Additional I/O directory
 -C <opt>        :Communications I/O directory
 -N              :Skip MPI initialization
 -D              :DataBases properties
 -S              :DataBases fragmentation
 -i              :Initialization
 -o <opt>        :Optics [opt=(c)hi/(b)se]
 -t <opt>        :The TDDFTs [opt=(a)LDA/(l)RC]
 -x              :Hartree-Fock Self-energy and Vxc
 -b              :Static Inverse Dielectric Matrix
 -p <opt>        :GW approximations [opt=(p)PA]
 -g <opt>        :Dyson Equation solver [opt=n(ewton)]
 -y <opt>        :BSE solver [opt=h/d]
```

As all programs in the `yambo` suite, information on the usage can be obtained with `-h` (short help) and `-H` (long help)
The options: `-i -o -t -x -b -p -g -y` are relative to the different runs you can perform with `yambo`. When you call `yambo` with one or more options the code

- ▷ Checks the existing databases to propose reasonable values for the parameters needed,
- ▷ creates and fills a text file (by default it is called `yambo.in` but you can specify a different name using the `-F` option)
- ▷ Call the editor you have specified during the compilation of the code (`vim` by default) to edit this input file.

Running yambo

The easiest thing you can do to run `yambo` is simply to launch it, without any input file

```
>yambo

 <---> [01] Job Setup
 <---> [02] Input variables setup
 <---> [02.01] K-grid lattice
 <---> [02.02] RL shells
 <---> Shells finder |####################| [100%] --(E) --(X)
 <---> [02.03] Input (E)nergies[ev] & Occupations
 <---> [03] Transferred momenta grid
 <---> X indexes |####################| [100%] --(E) --(X)
 <---> SE indexes |####################| [100%] --(E) --(X)
 <---> [04] Game Over & Game summary
```

In this case (and whenever there is no input file) `yambo` enters, by default, in the setup run, where the code performs some standard checks of the Brillouin zone sampling, Reciprocal space vectors, and so on. If we now list the contents of the `SAVE/` folder

```
> ls SAVE/
ndb.gops  ndb.kindx  ns.db1  ns.wf
```

we notice that this simple run has generated two new files: `ndb.gops, ndb.kindx`. These databases save the result of the setup run. In this way if you launch again `yambo` the code will skip all previously done calculations. This simple example shows the general philosophy of the code. Whenever it is possible `yambo` stores informations in a large number of files in order to save time in later calculations.

In this simple run we can also see that `yambo` writes a detailed list of informations about the run and the system in the `r_setup` file. Any possible run-level generates an appropriate report file.

3 Some physics ! The dielectric function of bulk Silicon

We are ready to do some physics, now. We want to calculate the zero and finite transferred momenta absorption of bulk silicon, within the Random Phase approximation. From the command line synopsis we see that we need to call

```
>yambo  -o c
```

Using this command `yambo` creates and edit the `yambo.in` input file

```
optics                          # [R OPT] Optics
chi                             # [R CHI] Dyson equation for Chi.
% QpntsRXd
  1 | 19 |                      # [Xd] Transferred momenta
%
% BndsRnXd
   1 |  50 |                    # [Xd] Polarization function bands
%
NGsBlkXd= 1              RL     # [Xd] Response block size
% EnRngeXd
  0.00000 | 10.00000 |   eV     # [Xd] Energy range
%
% DmRngeXd
  0.10000 |  0.10000 |   eV     # [Xd] Damping range
%
ETStpsXd= 100                   # [Xd] Total Energy steps
% LongDrXd
 1.000000 | 0.000000 | 0.000000 |      # [Xd] [cc] Electric Field
%
```

By reading the databases generated in the setup run `yambo` already knows that there are 19 momenta permitted by the Brillouin zone sampling, and that the `abinit` run has calculated 50 bands. To do a quick calculation we change the value of `QpntsRXd` to `1|10` and `QpntsRXd` to `1|2`. This means we are calculating the dielectric function at Γ point (the first momentum is always the Γ vector) and at the second momentum, that from the `r_setup` we know is $(-0.125, 0.0, -0.125)$.

We are now ready to run the code

```
 <---> [01] Job Setup
 <---> [02] Input variables setup
 <---> [02.01] K-grid lattice
 <---> [02.02] RL shells
 <---> [02.03] Input (E)nergies[ev] & Occupations
 <---> [03] Transferred momenta grid
 <---> [04] Optics
 <---> [WF loader] Wfs (re)loading |##########| [100%] --(E) --(X)
 <---> Dipole (T): |##########| [100%] --(E) --(X)
 <---> [FFT-X] Mesh size: 12  12  12
 <---> [WF loader] Wfs (re)loading |##########| [100%] --(E) --(X)
 <---> [X-CG] R(p) Tot o/o(of R)  :   216   6144    100
 <---> Xo@q[1] 1-100 |##########| [100%] --(E) --(X)
 <---> X @q[1] 1-100 |##########| [100%] --(E) --(X)
 <---> [X-CG] R(p) Tot o/o(of R)  :  1306   6144    100
 <---> Xo@q[2] 1-100 |##########| [100%] --(E) --(X)
 <---> X @q[2] 1-100 |##########| [100%] --(E) --(X)
 <01s> [05] Game Over & Game summary
```

If we now list the contents of the working directory

```
> ls
o.eel_q001-rpa  o.eps_q001-rpa  README        r_setup  Si_out_DS2_KSS
o.eel_q002-rpa  o.eps_q002-rpa  r_optics_chi  SAVE/    yambo.in
```

we see that there is a new report file (`r_optics_chi`) and four `output` files (always called `o*`).

By plotting the second versus the first column of the `o.eps_q001-rpa` and `o.eps_q002-rpa` output files we obtain the well-known absorption spectrum of bulk silicon.

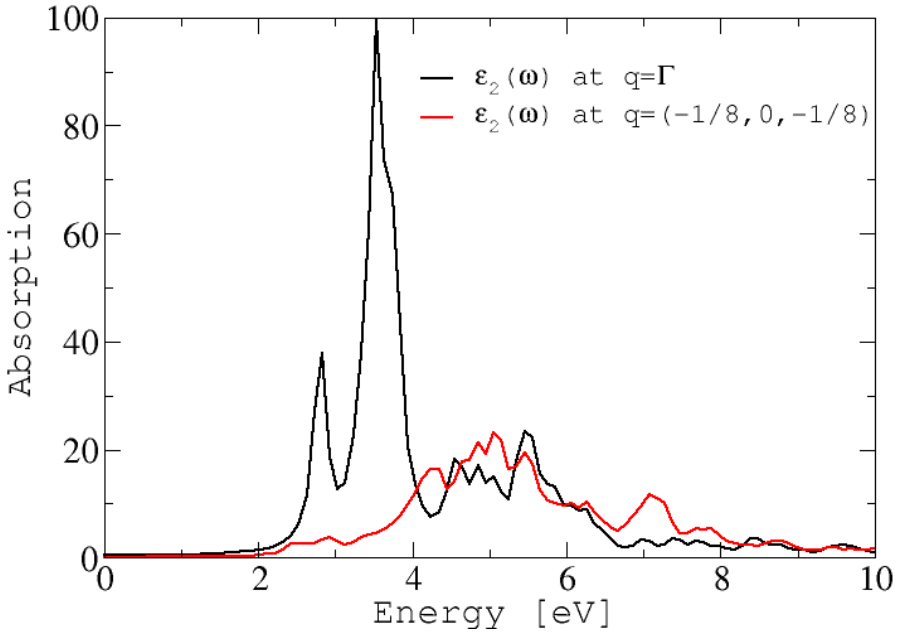

Figure 1 Some physics: zero and finite momentum absorption spectra of bulk Si within the Random Phase Approximation.

4 So what !?

If you think that `yambo` could be useful for your research please visit our web-page. Please, remember that

- We are all non-permanent researchers. This means that we cannot devote all our time to the development of `yambo`.
- For the same reason we cannot ensure direct support to the users. We will shortly open a forum for the users. So if you send us an E-mail, please keep in mind that we could need some days before answering.


Andrea


Myrta


Conor


Daniele