

Phonon dispersions with Density-Functional Perturbation Theory

These examples illustrate the applications of Density-Functional Perturbation Theory (DFPT) to the calculation of phonon dispersions and Interatomic Force Constants in simple semiconductors (Silicon).

GETTING READY

- The following command must have been executed:

```
module load qe/5.1
```

- You may download the entire exercise file [examples_disp.tgz](#) in a directory of your choice. Uncompress and unpack the file and enter in the resulting directory:

```
tar -zxvf examples_disp.tgz
cd examples_disp
```

INTERATOMIC FORCE CONSTANTS IN REAL SPACE

Whenever the entire phonon dispersion is needed, it is convenient to calculate *Interatomic Force Constants* (IFC) in real space. This calculation involves three steps:

1. the usual scf step for the unperturbed system
2. the calculation of phonons and dynamical matrices on a grid of \mathbf{q} -vectors (this can be done in a single step)
3. the transformation of the dynamical matrices from \mathbf{G} - to \mathbf{R} -space

Since the range of IFC's in real space is short for nonpolar materials, only a relatively coarse grid of \mathbf{q} -vectors in the Irreducible Brillouin Zone is needed. For polar materials, a suitable dipolar term is subtracted out from the IFC's; the remaining analytic term is short-ranged and undergoes the same procedure as in the nonpolar case; finally the non-analytic contribution is added back to the dynamical matrices when needed. Once the IFC's are calculated, it is possible to calculate phonon frequencies and eigenvalues at any wavevector, by just reconstructing the dynamical matrix. If the grid of \mathbf{q} -vectors is dense enough, the "reconstructed" dynamical matrix is very close to what one gets by direct calculation, also for a \mathbf{q} -vector that was *not* in the grid of \mathbf{q} used to calculate the IFC's in real space. Let us call this a "Fourier interpolation". Coming back to our three steps:

1. scf step as usual:

```
pw.x -in si.scf.in > si.scf.out
```

2. Calculation of dynamical matrices on a grid of \mathbf{q} -vectors (will require several minutes).

Sample input file: [si.ph.in](#). One has to specify the grid of \mathbf{q} -vectors as in a Monkhorst-Pack grid, with variables "nq1", "nq2", "nq3". The grid must contain Gamma and it must be a subdivision of the reciprocal lattice (i.e. it must correspond to a supercell in \mathbf{R} -space). It is automatically generated. One has also to specify "ldisp=.true." to instruct the code to loop over all \mathbf{q} -vectors.

```
ph.x -in si.ph.in > si.ph.out
```

Note that for each \mathbf{q} -vector the dynamical matrices are saved with a different name ("fildyn" + 1-8), while "fildyn" + 0 contains the information on the \mathbf{q} -vector grid (type of grid and number of points)

3. Calculation of IFC's in real space.

This task is performed by auxiliary program "q2r.x". All dynamical matrices are read and Fourier-transformed. The input file for q2r.x contains very little data: file name "fildyn" of dynamical matrices (the same given in input to the phonon code), whether to apply ASR (better to do it: asr='simple' or 'crystal' are ok for simple crystals), the name ("flfrc") of the output file containing the force constants.

```
q2r.x
&input fildyn='si.dyn', zasn='simple', flfrc='si444.fc' /
```

PHONON DISPERSIONS FROM INTERATOMIC FORCE CONSTANTS

Once the IFC's in real space are available, one can calculate phonons at any value of \mathbf{q} using code "matdyn.x" (do not confuse with "dynmat.x"). It can calculate phonon Density Of States (DOS) as well. Before starting more serious calculations, it is wise to check whether the quality of the Fourier interpolation performed in the previous step is sufficiently good, i.e. if you need a denser \mathbf{q} -vector grid (i.e. a longer range for the IFC's) or not. This can be easily checked by comparing the results of matdyn.x and of ph.x for a \mathbf{q} -vector that was *not* in the grid used in the Fourier interpolation. Once you are convinced that your IFC set is ok, let us proceed to two different types of calculations;

1. A plot of the phonon dispersions along selected high-symmetry lines.

Sample file: [si.matdyn.in](#). You *must* specify the name of the file ("flfrc") containing the IFC's, and a list of \mathbf{q} -points for which the frequencies are to be calculated. You may specify various forms of ASR ('simple' or 'crystal' are ok in this case), where to write the frequencies ("flfrq").

```
matdyn.x < si.matdyn.in
```

The output is contained in two files, specified oin variable flfrc: "si.freq", containing a list of frequencies in a format that can be read by code "plotband.x", and "si.freq.gp", can be directly plotted using gnuplot or xmgrace, e.g.:

```
gnuplot
gnuplot> plot './si.freq.gp' u 1:2 w lp, '' u 1:3 w lp, '' u 1:4 w lp,
'' u 1:5 w lp, '' u 1:6 w lp, '' u 1:7 w lp
```

Does it look like you expected it?

2. A plot of the phonon DOS

If you choose option "dos=.true." you must specify the grid of \mathbf{q} -vectors used to calculate the DOS (variables "nk1", "nk2", "nk3", using the Monkhorst-Pack logic). You should also specify where you want the DOS written ("fldos") Sample file: [si.phdos.in](#).

```
matdyn.x < si.phdos.in > si.phdos.out
```

The file selected in fldos, "si.phdos", contains the DOS in a format that can be plotted using xmgrace or gnuplot. Can you make the connection between the phonon dispersions with the phonon DOS?

POLAR MATERIALS

The calculation of IFC's in polar materials is basically the same as for nonpolar materials. You just have to remember to specify the appropriate option ("epsil=.true.") in the input of the phonon code. Can you produce a phonon dispersion plot for AlAs?

PHONONS WITH ULTRASOFT PSEUDOPOTENTIALS

Phonon calculations with Ultrasoft pseudopotentials are basically the same as for norm-conserving ones. You just have to be aware of the following tricks:

- The cutoff for the density, "ecutrho", in the self-consistent calculation, often needs to be quite high (i.e. higher than what is good for total-energy or structural-optimization calculations). In the example provided for Diamond: [c.scf.in](#), a cutoff of 450 Ry is used, while the cutoff for the wavefunction is just 27 Ry.
- The threshold for convergence in phonon calculations ("tr2_ph") should be set to quite low values, somewhat lower than what is good for norm-conserving pseudopotentials. In the example provided for the Gamma-point calculation in Diamond: [c.ph.in](#), tr2_ph is set to 1.0E-16.

Can you calculate phonons at Gamma for Diamond?

PHONONS IN MAGNETIC SYSTEMS

The example for Ni shows how to calculate phonon dispersions in a magnetic system (incidentally one that is also metallic and with ultrasoft pseudopotentials). The procedure is the same: self-consistency (pw.x) first (sample data: [ni.scf.in](#)), followed by the phonon calculation (ph.x) (sample data: [ni.ph.in](#)) and by inverse Fourier transform (q2r.x) to get the IFC's in real space (sample data: [ni.q2r.in](#)). Once you have the IFC's, you can plot the phonon dispersions using "matdyn.x" (sample data: [ni.matdyn.in](#)) and "plotband.x".