Phonon calculations in ONETEP

Fabiano Corsetti

September 20, 2013 (v2)

1 Theory

We make use of the harmonic approximation, in which the total energy of the system E^{tot} is expanded to quadratic order in the displacement of the ions about their equilibrium positions:

$$E^{\mathrm{tot}} = E^{\mathrm{eq}} + \frac{1}{2} \sum_{a,\alpha,\kappa,a',\alpha',\kappa'} u_{a,\alpha,\kappa} \phi^{\alpha,\alpha'}_{\kappa,\kappa'} \left(a, a' \right) u_{a',\alpha',\kappa'}, \tag{1}$$

where E^{eq} is the equilibrium energy and $u_{a,\alpha,\kappa}$ denotes a small displacement of ion α belonging to unit cell a in the Cartesian coordinate direction κ from its equilibrium position; $\boldsymbol{\phi} \left(a, a' \right)$ is known as the force constants matrix, defined as

$$\phi^{\alpha,\alpha'}_{\kappa,\kappa'} \left(a, a' \right) = \frac{\partial^2 E}{\partial u_{a,\alpha,\kappa} \partial u_{a',\alpha',\kappa'}}. \tag{2}$$

It can be shown that the phonon frequencies $\omega_{\mathbf{q},n}$ at wavevector $\mathbf{q}$ are the eigenvalues of the dynamical matrix $\mathbf{D} \left(\mathbf{q} \right)$, which can be calculated from the Fourier transform of the force constants matrix:

$$D^{\alpha,\alpha'}_{\kappa,\kappa'} \left(\mathbf{q} \right) = \frac{1}{\sqrt{M_\alpha M_{\alpha'}}} \sum_a \phi^{\alpha,\alpha'}_{\kappa,\kappa'} \left(a, 0 \right) \mathrm{e}^{-\mathrm{i} \mathbf{q} \cdot \mathbf{R}_a}, \tag{3}$$

where M_α is the mass of ion α and $\mathbf{R}_a$ is the lattice vector displacement for unit cell a. The vibrational free energy for the unit cell is then given by

$$F \left(T \right) = \frac{1}{2} \sum_{\mathbf{q},n} \omega_{\mathbf{q},n} + k_{\mathrm{B}} T \sum_{\mathbf{q},n} \ln \left(1 - \mathrm{e}^{-\omega_{\mathbf{q},n} / k_{\mathrm{B}} T} \right), \tag{4}$$

where the first term is the zero-point energy of the system, and the second term is the temperature-dependent part of the free energy. In the limit of an infinite periodic system the sum over $\mathbf{q}$ should be replaced by an integral of the phonon dispersion curves over the first Brillouin zone.

The `phonon` module in ONETEP uses the *finite-displacement* method to calculate the phonon frequencies of the system; for molecules (the default), only the Γ-point frequencies $\omega_{\mathbf{0},n}$ are calculated, while for supercells of bulk crystal, any arbitrary q point $\omega_{\mathbf{q},n}$ can be calculated. The elements of the force constants matrix are calculated by a central-difference formula, using either 2 (the default) or 4 displacements:

$$\begin{cases} \phi^{\alpha,\alpha'}_{\kappa,\kappa'} \approx \frac{F^{+}_{\alpha,\kappa} - F^{-}_{\alpha,\kappa}}{2d} & \texttt{phonon\_sampling 1} \text{ (2 disps.)} \\ \phi^{\alpha,\alpha'}_{\kappa,\kappa'} \approx \frac{-F^{2+}_{\alpha,\kappa} + 8F^{+}_{\alpha,\kappa} - 8F^{-}_{\alpha,\kappa} + F^{2-}_{\alpha,\kappa}}{12d} & \texttt{phonon\_sampling 2} \text{ (4 disps.)} \end{cases}, \tag{5}$$

where $F^{\pm}_{\alpha,\kappa}$ is the force on ion α in direction κ caused by a displacement $\pm d$ of ion α' in direction κ', and $F^{2\pm}_{\alpha,\kappa}$ is the same for a displacement $\pm 2d$. Therefore, $6N/12N$ calculations are needed in total, where N is the number of atoms in the system. However, each of these calculations is simply a small perturbation on the equilibrium configuration. Therefore, the converged set of NGWFs $\{\xi_\beta(\mathbf{r})\}$ and density kernel $\mathbf{K}$ that are obtained from a preliminary ground-state calculation on the equilibrium structure are used as the starting guess for each of the displacement calculations.

2 Overview of the `phonon` module

A phonon calculation in ONETEP is divided into three stages:

1. A ground-state calculation is performed for the unperturbed configuration, as specified in the input file. The forces on the ions are then calculated, and the code checks that the magnitude of the force on every ion is smaller than the value specified by `phonon_fmax`, as the starting configuration must correspond to a minimum in the energy landscape for the phonon calculation to be meaningful. If this requirement is not met, the calculation is interrupted.

2. Each ion is displaced in turn in the +ve and −ve x-, y-, and z-directions by a distance d (and, optionally, $2d$). For each displacement a separate ground-state calculation is performed. The initial description of the electronic structure is read in each time from the converged files `<seedname>.dkn` and `<seedname>.tightbox_ngwfs` for the unperturbed structure that have been obtained from Stage 1; the overwriting of these files is therefore disabled at the start of Stage 2. After each set of +ve/−ve displacements, one row of the force constants matrix is calculated and written to the file `<seedname>.force_consts_<i>`, where `<i>` is the number identifier of the row (going from 1 to $3N$ for an N-atom system). It is important to note that *not all rows are necessarily computed*, if some vibrational degrees of freedom are switched off (Sec. 3), and/or a supercell calculation of bulk crystal is being performed (Sec. 4); however, `<i>` retains the same value as it would have if all $3N$ rows were to be used.

3. The rows of the force constants matrix are read back in from the files `<seedname>.force_consts_<i>`, and the full force constants matrix is constructed. The dynamical matrix can then be calculated for the desired q points and diagonalized to find the phonon frequencies. First, the phonon frequencies are calculated on a regular grid of q points (Γ only for a molecule); in either case, the Γ-point frequencies only are printed to standard output. Then, the following thermodynamic quantities are calculated on the full grid and printed to standard output: the zero-point energy, and the free energy, entropy, internal energy, and specific heat within a user-specified temperature range. The phonon DOS is also calculated on the full grid and written to the file `<seedname>.qdos`. Additionally, the user can specify a list of arbitrary q points, for which the phonon frequencies (and, optionally, the corresponding eigenvectors) are calculated and written to the file `<seedname>.phonon_freqs`. Finally,

a list of Γ-point modes <j> can be specified for which animation files <seedname>.phonon_<j>.xyz are written.

This division in stages is done so as to allow for a *task farming* approach (see Sec. 5 for details).

3 Selecting degrees of freedom

The input file allows the user to select only a subset of the complete vibrational degrees of freedom of the entire system, as well as to specify different finite-displacement options for each (α, κ) pair. This is controlled through two keywords: phonon_vib_free and phonon_exception_list.

phonon_vib_free is an integer parameter controlling the global default of which Cartesian directions are 'switched on' for all ions. The options are listed in Table 1. The default option is 7, corresponding to all three Cartesian directions being switched on (i.e., all vibrational degrees of freedom are allowed).

phonon_exception_list is a block in which the user can list specific (α, κ) pairs with options differing from the global defaults defined by phonon_vib_free, phonon_sampling, and phonon_finite_disp. An example of doing so is as follows:

```
phonon_vib_free 3
phonon_sampling 1
phonon_finite_disp 1.4e-1 bohr

%block phonon_exception_list
10 3 1 2 0.9
15 1 0 1 1.0
36 2 0 1 1.0
%endblock phonon_exception_list
```

Here, we have first defined the global defaults; phonon_vib_free 3 corresponds to only the x- and y-directions being selected for the calculation, and the z-direction being switched off. Then, in the phonon_exception_list block, we list three exceptions:

phonon_vib_free	x	y	z
0	F	F	F
1	T	F	F
2	F	T	F
3	T	T	F
4	F	F	T
5	T	F	T
6	F	T	T
7	T	T	T

Table 1: Allowed options for keyword phonon_vib_free.

- displacement of ion 10 in the z-direction (3) is switched on (1), with a value of phonon_sampling of 2, and a value of phonon_finite_disp of 0.9 × the global value (i.e., 1.26e-1 bohr);
- displacement of ion 15 in the x-direction (1) is switched off (0), with the last two parameters not being read;
- displacement of ion 36 in the y-direction (2) is switched off (0), with the last two parameters not being read.

4 Bulk crystal supercell calculations

Phonon calculations for crystalline systems can be performed in ONETEP using a supercell approach, with either a real-space truncation of the force constants matrix, or a Slater-Koster style interpolation. The size of the supercell is chosen by the user; obviously, larger supercells will produce more accurate results at arbitrary q points.

It is the responsability of the user to provide the correct supercell lattice vectors and atomic coordinates in the usual way. Additionally, the supercell block must be specified to inform the phonon module that the system is a supercell of bulk material; otherwise, it will be assumed to be a molecule. An example of doing so is as follows:

```
%block lattice_cart
ang
5.3938105 5.3938105 0.0000000
5.3938105 0.0000000 5.3938105
0.0000000 5.3938105 5.3938105
%endblock lattice_cart

%block positions_abs
ang
Si 0.000000000 0.000000000 0.000000000
Si 0.000000000 2.696905250 2.696905250
Si 2.696905250 0.000000000 2.696905250
Si 2.696905250 2.696905250 5.393810500
Si 2.696905250 2.696905250 0.000000000
Si 2.696905250 5.393810500 2.696905250
Si 5.393810500 2.696905250 2.696905250
Si 5.393810500 5.393810500 5.393810500
Si 1.348452625 1.348452625 1.348452625
Si 1.348452625 4.045357875 4.045357875
Si 4.045357875 1.348452625 4.045357875
Si 4.045357875 4.045357875 6.742263125
Si 4.045357875 4.045357875 1.348452625
Si 4.045357875 6.742263125 4.045357875
Si 6.742263125 4.045357875 4.045357875
Si 6.742263125 6.742263125 6.742263125
%endblock positions_abs
```

```
%block supercell
2 2 2
1
9
%endblock supercell
```

Within the `supercell` block, the first line gives the shape of the supercell (2×2×2), and subsequent lines list the ions in the `positions_abs` block that belong to the 'base' unit cell (of course, this supercell is too small to give sensible results for a phonon calculation, and is probably too small to run in ONETEP anyway; a 1000-atom cubic supercell of Si gives excellent results however!)

When a supercell calculations is specified, only the ions within the unit cell are displaced, although the forces on all ions in the system are used to calculate the elements of the dynamical matrix from Eq. 3. It is also possible to specify `phonon_vib_free` and `phonon_exception_list` in a supercell calculation, although only the ions listed in the `supercell` block can be included in the `phonon_exception_list` block.

5 Task farming

The most efficient way of performing a phonon calculation is by task farming, as the full force constants matrix is built up from many perturbed-structure calculations, each of which is completely independent. This can be done with the following steps:

1. Run `phonon_farming_task 1` as a single job; this is essentially a standard single-point energy-and-force ONETEP calculation. Find the line in the main output file which gives the `Number of force constants` needed for the phonon calculation you have specified (this will be between 1 and $3N$).

2. Divide the total number of force constants that need to be calculated between the desired number of jobs. Prepare the ONETEP input file for each job specifying `phonon_farming_task 2` and a subset of the force constant calculations in the `phonon_disp_list` block. Make sure every job has access to the files `<seedname>.dkn` and `<seedname>.tightbox_ngwfs` obtained from the unperturbed calculation in the previous step.

3. Collect all the `<seedname>.force_consts_<i>` files and place them in the same directory. Finally, run `phonon_farming_task 3` as a single job, to construct the full force constants matrix and perform the post-processing calculations.

6 Keywords

The phonon calculation is selected by specifying `task phonon`. All other keywords related to the module are optional. They are:

- `phonon_farming_task` [Integer]
  Select which stage to perform (as described in Sec. 5). Can be either `1`, `2`, `3` for a single stage, or `0` for all stages. Default is `0`.

- phonon_sampling [Integer]
  Finite-difference formula to use (see Eq. 5). Default is 1.

- phonon_finite_disp [Physical]
  Ionic displacement distance d. Default is 1.0e-1 bohr.

- phonon_fmax [Physical]
  Maximum ionic force allowed in the unperturbed system. Default is 5.0e-3 'ha/bohr'.

- phonon_energy_check [Logical]
  Perform a sanity check that the total energy doesn't decrease upon ionic displacement. Default is F.

- phonon_vib_free [Integer]
  Default allowed vibrational degrees of freedom for all ions (see Sec. 3 for details). Default is 7.

- phonon_exception_list [Block]
  List of exceptions to the global defaults defined by phonon_vib_free, phonon_sampling, and phonon_finite_disp (see Sec. 3 for details). Default is unspecified.

- supercell [Block]
  Definition of the supercell used for crystalline material (see Sec. 4 for details). Default is unspecified.

- phonon_disp_list [Block]
  List of force constant calculations to perform for Stage 2. Note that the total number of force constant calculations is given in the main output file in the line Number of force constants; this will be less than or equal to $3N$. The numbers listed in the phonon_disp_list block should go from 1 to this number; *they can only be equated to the label* <i> *if all* $3N$ *force constants are calculated.* If unspecified, all displacements are performed. Default is unspecified. Example:

  ```
  %block phonon_disp_list
  1
  3
  5
  %endblock phonon_disp_list
  ```

- phonon_grid [Block]
  Definition of the regular grid of q points used for the computation of thermodynamic quantities and the phonon DOS. Default is 1 1 1 (i.e., Γ point only). Example:

  ```
  %block phonon_grid
  10 10 10
  %endblock phonon_grid
  ```

- phonon_SK [Logical]
  Use a Slater-Koster style interpolation for q points instead of a real-space cutoff of the force constants matrix elements. Default is F.

- `phonon_tmin` [Physical]
  Lower bound of the temperature range for the computation of thermodynamic quantities, expressed as an energy ($k_{\mathrm{B}}T$). Default is `0.0 hartree`.

- `phonon_tmax` [Physical]
  Upper bound of the temperature range for the computation of thermodynamic quantities. Default is `2.0e-3 hartree` ($\simeq$ 632 K).

- `phonon_deltat` [Physical]
  Temperature step for the computation of thermodynamic quantities. Default is `1.5e-5 hartree` ($\simeq$ 5 K).

- `phonon_min_freq` [Physical]
  Minimum phonon frequency for the computation of thermodynamic quantities, expressed as an energy ($\hbar\omega$); frequencies lower than this are discarded. Default is `3.6e-6 hartree` ($\simeq$ 5 cm^{-1}).

- `phonon_DOS` [Logical]
  Calculate the phonon DOS and write to file. Default is `T`.

- `phonon_DOS_min` [Real]
  Lower bound of the phonon DOS range (in cm^{-1}). Default is `0.0`.

- `phonon_DOS_max` [Real]
  Upper bound of the phonon DOS range (in cm^{-1}). Default is `1000.0`.

- `phonon_DOS_delta` [Real]
  Frequency step for the phonon DOS calculation (in cm^{-1}). Default is `10.0`.

- `phonon_qpoints` [Block]
  List of additional q points for which to calculate the phonon frequencies, in fractional coordinates of the reciprocal unit cell vectors. For non-supercell calculations only the Γ point can be specified. Default is unspecified. Example:

  ```
  %block phonon_qpoints
  0.0 0.0 0.0
  0.0 0.0 0.1
  0.0 0.0 0.2
  0.0 0.0 0.3
  0.0 0.0 0.4
  0.0 0.0 0.5
  %endblock phonon_qpoints
  ```

- `phonon_write_eigenvecs` [Logical]
  Write the eigenvectors as well as the phonon frequencies to file for the additional q points. Default is `F`.

- `phonon_animate_list` [Block]
  List of Γ-point modes (where `1` is the lowest) for which to write xyz animation files. Default is unspecified. Example:

```
%block phonon_animate_list
2
6
33
34
%endblock phonon_animate_list
```

- phonon_animate_scale [Real]
  Relative scale of the amplitude of the vibration in the xyz animation. Default is 1.0.

7 Additional notes

Phonon calculations are quite sensitive to the accuracy of the ionic forces calculated for the perturbed structures. Therefore, it is advisable to make sure that the forces are well-converged with respect to the usual parameters: cut-off energy, number and radius of NGWFs, and spatial cut-off of the density kernel.

Furthermore, it is also important to make sure that for a given set of parameters the forces are properly converged at the end of the energy minimization procedure, and that the numerical noise is reduced to a minimum; the code will not check this automatically, and the forces generally converge slower than the total energy. To ensure an accurate result, therefore, the following values for the convergence threshold parameters are suggested:

- ngwf_threshold_orig 1.0e-7.

- lnv_threshold_orig 1.0e-11.