

PHON (Version 1.23)

A program to calculate phonons using the small displacement method

Dario Alfè

Department of Earth Sciences and Department of Physics and Astronomy,
Thomas Young Centre@UCL and London Centre for Nanotechnology
University College London, Gower street, WC1E 6BT, London, U.K.

13/01/2010

Abstract

The program PHON calculates force constant matrices and phonon frequencies in crystals. From the frequencies it also calculates various thermodynamic quantities, like the Helmholtz free energy, the entropy, the specific heat and the internal energy of the harmonic crystal. The procedure is based on the small displacement method, and can be used in combination with any program capable to calculate forces on the atoms of the crystal.

1 Program summary

Title of program: PHON

Citation: D. Alfè, Computer Physics Communications **180**,2622-2633 (2009)

URL of program: <http://chianti.geol.ucl.ac.uk/~dario>

URL of tutorial: <http://chianti.geol.ucl.ac.uk/~dario>

Licensing provisions: None

Operating system: Unix

Program Language: FORTRAN 90

Memory requirement: Depends on super-cell size, but usually negligible

External subprograms ZHEEV and DSYEV (Lapack); needs BLAS

Keywords phonons, vibrations, harmonic systems, thermodynamics of harmonic systems

2 Theoretical overview

2.1 Phonon frequencies

The central quantity in the calculation of the phonon frequencies is the force-constant matrix $\Phi_{is\alpha,jt\beta}$, since the frequencies at wavevector \mathbf{k} are the eigenvalues of the dynamical matrix $D_{s\alpha,t\beta}$, defined as:

$$D_{s\alpha,t\beta}(\mathbf{k}) = \frac{1}{\sqrt{M_s M_t}} \sum_i \Phi_{is\alpha,jt\beta} \exp [i\mathbf{k} \cdot (\mathbf{R}_j^0 + \tau_t - \mathbf{R}_i^0 - \tau_s)] . \quad (1)$$

where \mathbf{R}_i^0 is a vector of the lattice connecting different primitive cells and τ_s is the position of the atom s in the primitive cell. If we have the complete force-constant matrix, then $D_{s\alpha,t\beta}$ and hence the frequencies $\omega_{\mathbf{k}s}$ can be obtained at any \mathbf{k} , so that $\bar{\omega}$ can be computed to any required precision. In principle, the elements of $\Phi_{is\alpha,jt\beta}$ are non-zero for arbitrarily large separations $|\mathbf{R}_j^0 + \tau_t - \mathbf{R}_i^0 - \tau_s|$, but in practice they decay rapidly with separation, so that a key issue in achieving our target precision is the cut-off distance beyond which the elements can be neglected.

2.2 Calculation of the force constant matrix

We calculate $\Phi_{is\alpha,jt\beta}$ by the small-displacement method. In harmonic approximation the α Cartesian component of the force exerted on the atom at position $\mathbf{R}_i^0 + \tau_s$ is

$$F_{is\alpha}^{u_{jt\beta}} = - \sum_{jt\beta} \Phi_{is\alpha,jt\beta} u_{jt\beta} \quad (2)$$

where $u_{js\beta}$ is the displacement of the atom in $\mathbf{R}_j^0 + \tau_t$ along the direction β and $F_{is\alpha}^{u_{jt\beta}}$ is the α component of the force induced on the atom at position $\mathbf{R}_i^0 + \tau_s$. The force constant matrix can be calculated as:

$$\Phi_{is\alpha,jt\beta} = - \frac{F_{is\alpha,jt\beta}^{u_{jt\beta}}}{u_{jt\beta}} \quad (3)$$

by displacing once at a time all the atoms of the lattice along the three Cartesian components by $u_{jt\beta}$, and calculating the forces $F_{is\alpha,jt\beta}$ induced on the atoms in $\mathbf{R}_i^0 + \tau_s$. Eqn.(3) computes the force constant matrix using forward differences; for numerical reasons, it can be advantageous in some cases to use central differences, in which case the force constant matrix can be calculated as:

$$\Phi_{is\alpha,jt\beta} = - \frac{F_{is\alpha,jt\beta}^{u_{jt\beta}} - F_{is\alpha,jt\beta}^{-u_{jt\beta}}}{2u_{jt\beta}} \quad (4)$$

Since the crystal is invariant under traslations of any lattice vector, it is only necessary to displace the atoms in one primitive cell and calculate the forces induced on all the other atoms of the crystal. In what follows we will assume this as understood and put simply $j = 0$.

It is important to appreciate that the $\Phi_{ls\alpha,l't\beta}$ in the formula for $D_{s\alpha,t\beta}(\mathbf{k})$ is the force-constant matrix in the infinite lattice, with no restriction on the wavevector \mathbf{k} , whereas the calculations of $\Phi_{ls\alpha,l't\beta}$ can only be done in supercell geometry. Without a further assumption, it is strictly impossible to extract the infinite-lattice $\Phi_{ls\alpha,l't\beta}$ from supercell calculations, since the latter deliver information only at wavevectors that are reciprocal lattice vectors of the superlattice. The further assumption needed is that the infinite-lattice $\Phi_{ls\alpha,l't\beta}$ vanishes when the separation $\mathbf{R}_{l't} - \mathbf{R}_{ls}$ is such that the positions \mathbf{R}_{ls} and $\mathbf{R}_{l't}$ lie in different Wigner-Seitz (WS) cells of the chosen superlattice. More precisely, if we take the WS cell centred on $\mathbf{R}_{l't}$, then the infinite-lattice value of $\Phi_{ls\alpha,l't\beta}$ vanishes if \mathbf{R}_{ls} is in a different WS cell; it is equal to the supercell value if \mathbf{R}_{ls} is wholly within the same WS cell; and it is equal to the supercell value divided by an integer P if \mathbf{R}_{ls} lies on the boundary of the same WS cell, where P is the number of WS cells having \mathbf{R}_{ls} on their boundary. With this assumption, the $\Phi_{ls\alpha,l't\beta}$ elements will converge to the correct infinite-lattice values as the dimensions of the supercell are systematically increased.

It is not always necessary to displace all the atoms in the primitive cell, since the use of symmetries can reduce the amount of work needed. This is done as follows. We displace one atom in the primitive cell, let's call it 'one', and we calculate the forces induced by the displacement on all the other atoms of the supercell. Then we pick up one other atom of the primitive cell, atom 'two'. If there is a symmetry operation S (not necessarily a point group symmetry operation) such that, when S is applied to the crystal atom two is sent into atom one and the whole crystal is invariant under such transformation, then it is not necessary to displace atom two, and the part of the force constant matrix associated with its displacement can be calculated using

$$\Phi_{is,02} = \mathbf{B}(S) \Phi_{\lambda_{is}(S),01} \mathbf{B}(S^{-1}), \quad (5)$$

where $\mathbf{B}(S)$ is the 3×3 matrix representing the point group part of S in Cartesian coordinates, and $\lambda_{is}(S)$ indicates the atom of the crystal where the atom in $\mathbf{R}_i^0 + \tau_s$ is brought because of the action of the symmetry operation S . If there is no symmetry operation connecting atom two to atom one then atom two is displaced and all the induced force field is calculated. The procedure is repeated for all the atoms of the primitive cell.

In principle each atom has to be displaced along the three Cartesian directions. It is sometimes convenient to displace the atoms along some special directions so as to maximize the number of symmetry operations still present in the 'excited' supercell, in this way the calculations of the forces are less expensive. This can always be done, as long as one displaces the atoms along three linearly independent directions. The forces induced by the displacements along the three Cartesian directions is easily reconstructed by the linear combination

$$\mathbf{F}_{is,0t\alpha} = \sum_l \mathbf{A}_{l\alpha} \tilde{\mathbf{F}}_{is,0tk} \quad (6)$$

where $\tilde{\mathbf{F}}_{is,0tk}$ is the force induced on the atom in $\mathbf{R}_i^0 + \tau_s$ due to a displacement of the atom in τ_t along the direction \mathbf{u}_k , and $\mathbf{A} = (\frac{\mathbf{u}_1}{|\mathbf{u}_1|}, \frac{\mathbf{u}_2}{|\mathbf{u}_2|}, \frac{\mathbf{u}_3}{|\mathbf{u}_3|})^{-1}$ is the inverse of the 3×3 matrix whose columns are the normalized displacements in Cartesian coordinates.

Using symmetries it is possible to reduce the number of displacements even further: if applying a point group symmetry operation U to the displacement vector \mathbf{u}_1 one obtains a vector \mathbf{u}_2 which is linearly independent from \mathbf{u}_1 , then the force field that would be induced by the displacement \mathbf{u}_2 can be calculated by

$$\mathbf{F}_{is,0t2} = \mathbf{B}(U)\mathbf{F}_{\lambda_{is}(U^{-1}),0t1}. \quad (7)$$

If a linearly independent direction cannot be found one has to displace the atom along a chosen independent direction and perform an other calculation. This is done until a set of three independent directions is found.

The force constant matrix is invariant under the point group symmetry operations of the crystal. This is not automatically guaranteed by the procedure just described, because in general the crystal is not harmonic, and therefore eqns. (3, 4) are only an approximation. So, the force constant matrix must be symmetrized with respect to the point group operations of the crystal:

$$\Phi_{is,0t} = \frac{1}{N_G} \sum_U \mathbf{B}(U)\Phi_{\lambda_{is}(U),0t}\mathbf{B}(U^{-1}). \quad (8)$$

The symmetrization of the force constant matrix removes all even-order anharmonicities [1]. The harmonic approximation becomes better and better as the displacement are made smaller and smaller. However, if the displacements are small, also the force induced are small, but there is a limit in the accuracy achievable in the calculations, so one cannot make too small displacements. Usually a fraction of a % of the nearest-neighbour distance is a good compromise.

As an example of the procedure just described let's consider the h.c.p. crystal. There are two atoms in the primitive cell, so in principles we would need six independent calculations. We will see that the number of calculations needed is equal to two. In first place one can easily recognize that only one atom needs to be displaced: if we traslate the crystal from one atom to the other and we perform a spatial inversion the crystal remains unchanged. Secondly, by applying a clockwise rotation of 120 degrees, for example, to a displacement in the x direction, one obtains an independent displacement. So only one additional displacement along the z direction is needed.

3 Description of the program and input/output files

The program reads the following files:

- INPHON: contains the input parameters
- POSCAR: contains the crystal parameters and atomic positions, the format is the same as in the program VASP [2], and is the following:

```
SiO2
-47.88
4.2212779176  0.0000000000  0.0000000000
0.0000000000  4.2212779176  0.0000000000
0.0000000000  0.0000000000  2.6869912363
2 4
Direct
0.00000000  0.00000000  0.00000000
0.50000000  0.50000000  0.50000000
0.30657944  0.30657944  0.00000000
0.69342056  0.69342056  0.00000000
0.19342056  0.80657944  0.50000000
0.80657944  0.19342056  0.50000000
```

This example is the Stishovite structure of SiO_2 . The first line is a comment, the second line is a scale factor, if negative is interpreted as the volume of the system (like in this case). Then the three lattice vectors, followed by the number of atoms for each specie (here 2 silicons and 4 oxygens). The next line precedes the atomic coordinates, and tells if these are in unit of the the lattice vectors of in Cartesian coordinates (here they are in unit of crystal coordinates). Finally the atomic coordinates. Note that the crystal must be in its ground state. It is important to relax the primitive cell very accurately before starting any phonon calculation.

- FORCES: contains the displacements and the force field for each displacement, the format is the following:

```
4
1  0.005  0.000  0.000
-0.61854500  0.18897800  0.00000000
```

```

0.30803200 -0.00013900 0.00000000
0.01746700 0.01833500 0.00000000
-0.01926700 -0.01606600 0.00000000
0.14414800 -0.08517300 0.00000000
0.16816400 -0.10593400 0.00000000
1 0.000 0.000 0.005
0.00000000 0.00000000 -0.29116700
0.00000000 0.00000000 -0.00207200
0.01356000 0.01356000 0.01857500
-0.01356000 -0.01356000 0.01857500
-0.00997600 0.00997600 0.12804500
0.00997600 -0.00997600 0.12804500
3 0.005 0.000 0.000
-0.00548400 -0.00305700 0.00000000
0.15831200 0.09934800 0.00000000
-0.42565500 -0.18284700 0.00000000
0.08338600 0.08818100 0.00000000
-0.01075400 0.01564000 0.00000000
0.20019500 -0.01726600 0.00000000
3 0.000 0.000 0.005
-0.00001000 -0.00001000 0.01963200
0.00388400 0.00388400 0.12906300
0.01036900 0.01036900 -0.25656500
-0.01400200 -0.01400200 -0.00218800
-0.01406100 0.01382000 0.05502900
0.01382000 -0.01406100 0.05502900

```

The first line is the number of displacements, then, for each displacement, a line containing a number which indicates the position of the atom in the super-cell which has been moved, followed by the displacement (in crystal coordinates), followed by the forces on all the atoms in the super-cell (in units of eV/Å and in Cartesian coordinates). For central differences (LCENTRAL = .T.) the format of the file is the same, but there are twice as many displacements (for each displacement \mathbf{u} there is also $-\mathbf{u}$).

In this example the super-cell is a primitive cell (to economise on space), in general you need to construct a super-cell, displace the atoms in the primitive cell appropriately and calculate the induced forces. The procedure is explained below.

3.1 Super-cell generation

The first step to calculate phonons in a crystal is to construct the super-cell, this can be done using the following setting:

```

LSUPER = .TRUE. (default)
NDIM = NX NY NZ (default: 1 1 1)
NTYPES = 2 (no default)

```

where NX, NY and NZ are positive integer numbers. NTYPES is the number of different atomic species (2 in the Stishovite example). PHON reads the file POSCAR which must contain a super-cell, ($n_x \mathbf{a}_1, n_y \mathbf{a}_2, n_z \mathbf{a}_3$) (possibly $n_x = n_y = n_z = 1$) and generates the file SPOSCAR which contains the super-cell whose lattice vectors are (NX \mathbf{a}_1 , NY \mathbf{a}_2 , NZ \mathbf{a}_3), where ($\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$) are the lattice vectors of the primitive cell. Do not use $NX \neq NY$ if the crystal has a symmetry operation which connects \mathbf{a}_1 with \mathbf{a}_2 . For face-centred-cubic (FCC) and body-centred-cubic (BCC) crystals do not use simple-cubic lattice vectors with 4 and 2 atoms in the primitive cell respectively, use proper FCC and BCC lattice vectors and one atom per primitive cell.

The program prints out a guess for the displacements needed to construct the full force field. These are written on the stout and in the file DISP. The size of the displacement can be changed setting the variable DISP:

DISP = 25 (default, corresponding to displacements of 0.04 Å).

In principle, a single off symmetry displacement could be enough to generate the full force field, as three linearly independent displacement may be generated from this using the symmetry operations of the crystal (this may not be true if the crystal has low symmetry). In practice, however, one would like to use displacements that alter as little as possible the symmetry of the crystal, because this minimises the computational effort in *ab-initio* calculations (maximising the number of symmetry operations in general reduces the number of \mathbf{k} -points needed). For this reason, the displacements suggested

by the program work on the assumption that the first displacement will be along the x -axis, and appropriate additional displacements along y and/or z are suggested if they are needed. The user may suggest his own first displacement using the variable: DXSTART:

DXSTART = xstart ystart zstart

To use central differences (eqn. 4 instead of eqn. 3), set the variable LCENTRAL = .T.

3.2 Phonon dispersions

To calculate the dispersions we need to add the following settings in the INPHON file:

```
LRECIP = .T.
ND = 4; NPOINTS = 50
QI = 0.5 0.5 0.0 0.0 0.0 0.0 0.5 0.0 0.0 0.5 0.5 0.5
QF = 0.0 0.0 0.0 0.5 0.0 0.0 0.5 0.5 0.5 0.0 0.0 0.0
```

MASS = 28.085 16.0.

LRECIP = .T. means that the coordinates of the initial (QI) and final (QF) \mathbf{q} -points are in units of reciprocal space lattice vectors. Set LRECIP=.F. if you want to give these numbers in Cartesian coordinates. ND is the number of initial and final points (usually high symmetry points in the Brillouin zone (BZ)), and NPOINTS is the number of points between them. MASS is the mass of the ions. We also need to add:

LSUPER= .F.

which will tell PHON to go beyond the generation of the super-cell (and not generating one). These settings will make PHON generate the file FREQ containing the frequencies in THz, and the file FREQ.cm with the frequencies in cm^{-1} . If the number of atoms in the primitive cell is larger than 16 then FREQ and FREQ.cm are not written, and PHON produces a number of files FREQ1 FREQn, ... containing 48 phonon modes each (in THz units). By setting the variable:

LFORCEOUT=.TRUE. (default = .FALSE.)

The program writes the file HARMONIC which contains the force constant matrix.

3.3 Density of states and thermodynamic quantities

The program also calculates the density of states, writing the file DOS. The input variables:

DOSIN = 0; DOSEND = 25; DOSSTEP = 0.1; DOSSMEAR = 0.02

specify the starting and the ending frequency (in THz), the step and the smear. The smear is the width of the Gaussian which is convoluted with the dos spectrum. The density of states will be calculated using the available frequencies, which in this case are those coming from the dispersion curves. This is not the best way to do, because the DOS is the integral over the whole BZ of the phonon frequencies, and the dispersions along some special directions do not usually provide a good sampling of the BZ.

The correct way to calculate the density of states is to generate a set of points which accurately sample the BZ, and calculate the frequencies at these points. This can be done with the following settings:

```
LFREE = .TRUE.; TEMPERATURE = 1000
LGAMMA = .FALSE.
QA = 11; QB = 11; QC = 11
```

QA, QB and QC are the divisions for a Monkhorst and Pack (MP) [3] special points grid generations, LGAMMA = .TRUE. means that the grid will pass through Γ . With the setting LFREE = .TRUE. the program will also calculate zero-point energy, Helmholtz free energy, internal energy, specific heat and entropy, at the temperature given by the variable TEMPERATURE in degrees Kelvin. The MP special points are written in the file QPOINTS, which can be used for subsequent calculations without generating the points again (dense meshes can take time to be generated) by setting QA to a negative number (default).

A convenient way to generate a set of thermodynamic properties as function of temperature is to set the variable:

PTEMP= 10 90

This will create a file called THERMO, which contains the internal energy, the free energy, the free energy in the classical limit, the entropy and the constant volume specific heat, as function of temperature, in the example above starting from a temperature equal to the value of the input variable TEMPERATURE, in steps of 10 K, for 90 steps.

3.3.1 Partial density of states

In some experiments the measured density of states can be due to the vibrational frequencies of only some of the atoms of the material investigated. For example, this is the case when inelastic X-ray spectroscopy is used to exploit the Mössbauer effect, in which only the atoms having a nuclear energy level which resonate with the incoming X-ray radiation can interact with the radiation. The result is that only the vibrations of these atoms are sampled in these experiment [4].

In order to calculate the partial density of states in a system in which, for example, there are two different atoms, the following setting:

USETHIS = .F. .T.

will calculate the partial density of states due to the second atom only.

3.4 Translational invariance

When all the atoms are moved by the same amount, i.e. the crystal is rigidly shifted, the force *on each atom* must be zero. This is a stronger constraint than the one in which it is the *sum* of the forces on each atom to be zero. The latter is expressed by:

$$\sum_{s,t,i} \Phi_{is\alpha,0t\beta} = 0, \quad (9)$$

where $\Phi_{is\alpha,0t\beta}$ is the force constant matrix, s and t run over the number of atoms N in the primitive cell and i over the M lattice vectors included in the calculation. If this constraint is not satisfied, it is straightforward to impose it by subtracting from the calculated force on each atom the value $\mathbf{F}/(MN)$, where $\mathbf{F} = \sum_{s,i} \mathbf{F}_{si}$, and \mathbf{F}_{si} is the force acting on atom s in primitive cell i .

The former condition is:

$$\sum_{s,i} \Phi_{is\alpha,0t\beta} = 0; \quad \text{for each } t = 1, N. \quad (10)$$

Clearly, Eq. 9 implies Eq. 10, but the opposite is not true in general. However, it is Eq. 10 to imply that at $\mathbf{q} = (0,0,0)$ the three acoustic branches have identically zero frequencies.

The constraint in Eq. 10 has to be imposed in such a way that the force constant matrix remains symmetric: $\Phi_{\alpha,\beta}(\mathbf{R}_j + \tau_t - \mathbf{R}_i - \tau_s) = \Phi_{\beta,\alpha}(-[\mathbf{R}_j + \tau_t - \mathbf{R}_i - \tau_s])$. In the PHON code this is done iteratively, in a number of steps in which the symmetry is re-imposed at each step.

To impose translational invariance as described in set with the variable:

NTI = 20

Translational invariance is imposed iteratively, and ~ 20 iterations are usually enough.

The amount of output printed by the PHON is controlled by the variable IPRINT. IPRINT=0 will produce a minimal output, IPRINT=3 a verbose output, which also includes the dynamical matrix and its eigenvectors.

3.5 Graphical representation of phonons at Γ

Setting the variable LEIGEN = .T. the program produces the files EIGEN.axsf and MODE???.axsf, to be used with the program XCrySden [5] to visualise phonon vibrations at Γ . The file EIGEN.axsf contains the eigenvectors of the dynamical matrix. These can be visualised with XCrySden by typing:

```
xcrysden -axsf EIGEN.axsf
```

and then ticking in the “Display” menu the entry “Forces”. Atomic displacements for each mode will then be visualised as arrows.

To visualise vibrations dynamically, use the files MODE???.axsf. Typing:

```
xcrysden -axsf MODE004.axsf
```

Input variable	Meaning	Variable type	Default
NTYPES	Number of atomic types	Integer	No default
MASS	Mass of atoms	Real*NTYPES	1.0
LSUPER	Generate supercell	Logical	.T.
NDIM	Supercell dimensions	Integer*3	(1,1,1)
LFREE	Thermodynamic properties	Logical	.F.
TEMPERATURE	Temperature	Real	No default
PTEMP	Temperature increment, number of increments	Real*2	(0.0, 1.0)
LFORCEOUT	Writes force constant matrix file	Logical	.F.
LSYMM	Symmetrises force constant matrix	Logical	.T.
SYMPREC	Precision threshold	Real	1d-6
DXSTART	Initial displacement	Real*3	(1.0, 0.0, 0.0)
DISP	One over size of displacement (in Å ⁻¹)	Integer	25 (0.04 Å)
LCENTRAL	For central differences	Logical	.F. (Forward differences)
QA, QB, QC	Number of divisions for q-points grid	Integers	No defaults
LGAMMA	q-points grid through Γ	Logical	.F.
DOSIN, DOSEND,	Density of states parameters	Reals	0.0, 25.0, 0.1 (THz)
DOSSTEP		Real	0.02 (THz)
DOSSMEAR	Width of Gaussian convoluted with DOS	Real	0.02 (THz)
USETHIS	Partial density of states	Logic*NTYPES	.T. (total DOS)
ND	Number of q-points in dispersion segments	Integer	0
QI, QF	Initial and final points for dispersion segments	Real*3	(0.0,0.0,0.0), (0.0,0.0,0.0)
LRECIP	q-points in reciprocal lattice coordinates	Logical	.T.
NTI	Imposes translational invariance	Integer	1 (no TI imposed)
IPRINT	Controls verbosity of output	Integer	0 (little output)
LEIGEN	Graphical representation of vibrations	Logical	.F. (no graphics)
NAME?	Name of atoms in the primitive cell	Character*NTYPES	H
EIGSIZE	Amplitude of vibrations	Real	1.0
NCYCLESEIG	Number of vibration cycles	Integer	2

Table 1: Input variables

will, for example, visualise mode number 4. The number of cycles of vibrations is two by default, and it can be changed using the variable NCYCLESEIG. The amplitude of the displacement can be altered using the variable EIGSIZE (default = 1). The variables NAME? can be used to name the atoms of the primitive cell (default: NAME? = H).

3.6 Input variables and Input/Output files

A full list of input variables is reported in Table 1, and a list of the input/output files needed/generated by PHON is reported in Table 2

4 Test case

I now describe one of the three examples distributed with the program. This is located in the subdirectory “examples/Al”. The example is Al in the FCC structure. The POSCAR contains already a $4 \times 4 \times 4$ super-cell (64 atoms), and the file FORCES has been already constructed using forces calculated with a density functional theory code. The INPHON file is set to calculate phonon density of states and thermodynamic properties in the range of temperature 1000 - 1890 K. Running PHON with this INPHON file will produce the files DOS, DOS.meV and DOS.cm, containing the density of states in units of number of frequencies/THz, or number of frequencies/meV or number of frequencies/cm⁻¹ respectively, and the file THERMO which contains the internal energy, the free energy, the free energy in the classical limit, the entropy and the constant volume specific heat. The following are the first lines of the file THERMO:

```
# T(K) E(eV/cell) F(eV/cell) Fc(eV/cell) S(kB/cell) Cv(kB/cell)
1000.00 0.26047710 -0.33379199 -0.33477073 6.89616445 2.97739731
1010.00 0.26304303 -0.33974746 -0.34071653 6.92579274 2.97784007
1020.00 0.26560934 -0.34572835 -0.34668793 6.95513343 2.97826997
1030.00 0.26817601 -0.35173440 -0.35268468 6.98419200 2.97868750
1040.00 0.27074303 -0.35776537 -0.35870653 7.01297378 2.97909314
.....
```

File name	Input/Output	Description of data
INPHON	Input	Input variables
POSCAR	Input	Supercell
FORCES	Input	Forces on all atoms (eV/Å), for each displacement
QPOINTS	Input/Output	q-points for thermodynamics
DISP	Output	Suggestions for displacements
SPOSCAR	Output	File containing supercell
FREQ, FREQ.cm	Output (if natoms ≤ 16)	Phonon dispersions (THz and cm ⁻¹)
FREQi, i=1,natoms/16	Output (if natoms > 16)	Phonon dispersions in groups of 48 (THz)
DOS, DOS.cm, DOS.meV	Output	Density of states (THz, cm ⁻¹ , meV)
THERMO	Output	Thermodynamic properties
HARMONIC	Output	Force constant matrix
EIGEN.axsf	Output	Eigenvectors (for representation with XCrySDen)
MODE???.axsf	Output	Phonon vibrations (for representation with XCrySDen)

Table 2: Input/Output files

Now set LFREE=.F. in the INPHON file and run the code again. The files FREQ and FREQ.cm will appear, which contain the phonon frequencies along three special directions in the Brillouin zone. The file FREQ contains the frequencies in units of THz and FREQ.cm in units of cm⁻¹. The following are the first lines of the file FREQ:

```

0.00000000  -0.00000  -0.00000  -0.00000
0.01428499  0.11682   0.12905   0.22737
0.02856997  0.23376   0.25796   0.45456
0.04285496  0.35096   0.38659   0.68141
0.05713994  0.46853   0.51481   0.90775
.....

```

References

- [1] G. Kresse, J. Furthmüller and J. Hafner. *Europhys. Lett.* **32** (1995), p. 729.
- [2] G. Kresse and J. Furthmüller. *Phys. Rev. B* **54** (1996), 11169.
- [3] H. J. Monkhorst and J. D. Pack. *Phys. Rev. B* **13** (1976), 5188.
- [4] W. Sturhahn, T. S. Toellner, E. E. Alp, X. Zhang, M. Ando, Y. Yoda, S. Kikuta, M. Seto, C. W. Kimball, B. Dabrowski. *Phys. Rev. Lett.* **74** (1995), p. 3832.
- [5] A. Kokalj, *Comp. Mater. Sci.*, 2003, Vol. 28, p. 155. Code available from <http://www.xcrysden.org/>