Numerical Methods in Physics
Phonons

Jesper Vesterberg (jeve0010@student.umu.se)

March 28, 2015

Abstract

Here we have the abstract, stuff about main results and what not.

Numerical Methods in Physics
supervisor: Peter Olsson

1   Introduction

Placeholder

2   Theory

2.1   Potential in a Crystal

The structure of crystals is typically described by using a lattice $\mathbf{l} = (l_{x1}, l_{x2}, l_{x3})$, which is a vector of integers that describes the position of every possible site[1] in a crystal structure in terms of a lattice constant a. If one then go on to describe a sites displacement in this structure as $\mathbf{u}^l = (u_{x1}^l, u_{x2}^l, u_{x3}^l)$, we can begin to describe the potential energy in the system. By using a simple Taylor expansion of the potential energy in the crystal and expanding it to the second term and noting that the first order term is vanishing we can get the equations of motion[2]

$$m\ddot{u}_\alpha^l = -\sum_{\mathbf{l}'\beta} D_{\alpha\beta}^{\mathbf{l}\mathbf{l}'} u_\beta^{\mathbf{l}'} \tag{1}$$

where α and β can have the integer value of one, two or three[3], m is the mass of whatever inhabit the sites in the crystal[4], $\mathbf{l}$ and $\mathbf{l}'$ are only different site positions in the lattice. Lastly we have term $D_{\alpha\beta}^{\mathbf{l}\mathbf{l}'}$ which actually is written

$$D_{\alpha\beta}^{\mathbf{l}\mathbf{l}'} = \delta_{\mathbf{l}\mathbf{l}'} \sum_{\mathbf{l}''} \frac{\partial^2 \phi(\mathbf{l} - \mathbf{l}'')}{\partial x_\alpha \partial x_\beta} - \frac{\partial^2 \phi(\mathbf{l} - \mathbf{l}')}{\partial x_\alpha \partial x_\beta} \tag{2}$$

where ϕ is the potential force felt by every individual site in the lattice, $\delta_{\mathbf{l}\mathbf{l}'}$ is the kronecker delta and $\mathbf{l}''$ is yet another seperate index for the different sites in the lattice.

2.2   Model for a Rare Gas Crystal

Looking at equation 1 we can see that we have a number of harmonic oscillators, thus we can start to see that we will be able to form an eigenvalue problem to solve for the different frequencies that can propagate through a crystal. But in order to simplify things we will only look at a couple of rare gases: Ne, Ar, Kr and Xe. One beautiful thing with these gases is that the crystal structure is so arranged that all the sites are a $\sqrt{2}a$ distance away from its closest neighbor. We can then use as simple version of the Mie-Lennard-Jones potential to simulate the individual potential forces at each site, this gives a potential of

$$\phi(r) = 2\epsilon \left[\frac{1}{2}\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right] \tag{3}$$

where σ is the distance between two sites in the lattice where the potential is zero, ϵ is the depth of the lennard-jones potential well[5] and r are the distance between two sites. Using this information we can set up the dynamic matrix and solve for the frequency. To

---

[1]Sites are usually where an certain atom resides for solids
[2]By no means a straightforward thing to do
[3]Representing the spacial dimensions
[4]we are only gonna look at monoatomic gases so m should be the same for all the sites
[5]This is expanded on in [1]

start with we write out the whole system described by equation 1 and 3. After that we do the ansatz that

$$u_\alpha^l = \frac{\epsilon_\alpha}{\sqrt{m}} e^{i(\mathbf{k}\cdot\mathbf{r}_l - \omega t)} \tag{4}$$

and we end up with

$$\omega^2 \varepsilon_x = \Big(\frac{1}{2m}[A+B][8 - 4\cos q_x\pi \cos q_y\pi - 4\cos q_x\pi \cos q_z\pi] \tag{5}$$

$$+ \frac{B}{m}[4 - 4\cos q_y\pi \cos q_z\pi]\Big)\varepsilon_x \tag{6}$$

$$+ \frac{1}{2m}[A-B][4\sin q_x\pi \sin q_y\pi]\varepsilon_y \tag{7}$$

$$+ \frac{1}{2m}[A-B][4\sin q_x\pi \sin q_z\pi]\varepsilon_z \tag{8}$$

where

$$A \equiv \frac{1}{r_{nn}}\phi'(r_{nn}) \tag{9}$$

$$B \equiv \phi''(r_{nn}). \tag{10}$$

And we have replaced the vector $\mathbf{k}$ with the vector $\mathbf{q} = \mathbf{k}a/\pi$. And the subindexes x, y, z corresponds to a spacial direction.

If one were to look at the crystal structure for these gases we can see that they are symetric in x,y and z-direction. Thus with some simple permutations we get

$$\omega^2 \varepsilon_y = \Big(\frac{1}{2m}[A+B][8 - 4\cos q_y\pi \cos q_z\pi - 4\cos q_y\pi \cos q_x\pi] \tag{11}$$

$$+ \frac{B}{m}[4 - 4\cos q_z\pi \cos q_x\pi]\Big)\varepsilon_y \tag{12}$$

$$+ \frac{1}{2m}[A-B][4\sin q_y\pi \sin q_z\pi]\varepsilon_z \tag{13}$$

$$+ \frac{1}{2m}[A-B][4\sin q_y\pi \sin q_x\pi]\varepsilon_x \tag{14}$$

$$\omega^2 \varepsilon_z = \Big(\frac{1}{2m}[A+B][8 - 4\cos q_z\pi \cos q_x\pi - 4\cos q_z\pi \cos q_y\pi] \tag{15}$$

$$+ \frac{B}{m}[4 - 4\cos q_x\pi \cos q_y\pi]\Big)\varepsilon_z \tag{16}$$

$$+ \frac{1}{2m}[A-B][4\sin q_z\pi \sin q_x\pi]\varepsilon_x \tag{17}$$

$$+ \frac{1}{2m}[A-B][4\sin q_z\pi \sin q_y\pi]\varepsilon_y \tag{18}$$

$$\tag{19}$$

With this we have eigenvalue problems in every direction to solve for so that we can get the frequencies of the phonons in a rare gas crystal.

2.3   Volume Dependency of Phonon frequencies

We want to look at how the volume of a crystal changes the frequencies of the phonons in the crystal. Trough approximation we say that σ and ϵ in equation 3 is not affected by

this change in volume. This will make calculations way simpler. We can then calculate the volume dependence as

$$\gamma_j(\mathbf{q}) = -\frac{\partial \ln \omega(\mathbf{q}, j)}{\partial \ln V} \tag{20}$$

where V is the volume of the crystal structure in the lattice. Why we have some ln terms is basically due to the fact that everyone else are doing it[6]. If one expands the derivate we end up with the expression

$$\gamma_j(\mathbf{q}) = -\frac{V}{\omega(\mathbf{q}, j)}\frac{\partial \omega(\mathbf{q}, j)}{\partial V} \tag{21}$$

Which then can be solved using a finite difference approximation scheme for the derivate; for example

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x-h)}{2h} \tag{22}$$

2.4   Heat Capacity

One can derive expression for heat capacity using both thermal physics and solid state physics. Historically the models described by thermal physics has been empirically correct in the relatively high temperatures that we humans tend to live in [4]. But for lower temperature the theory seem to break apart. Solid state physics can describe what happens at lower temperatures to heat capacity [2]. To make sure that the heat capacity derived by solid state theory approaches the heat capacity derived through thermal physics in the high temperature limit we will derive values from both theories and compare them. If they converge at higher temperatures it would clearly indicate the correctness of the written software.

2.4.1   Solid State Physics

One can derive the following expression for the heat capacity using solid state physics models[2]

$$C_V = k_B \sum_{\mathbf{q},j} \left[\frac{\hbar\omega(\mathbf{q}, j)}{k_B T}\right]^2 \frac{\exp\left[\frac{\hbar\omega(\mathbf{q},j)}{k_B T}\right]}{\left(\exp\left[\frac{\hbar\omega(\mathbf{q},j)}{k_B T}\right] - 1\right)^2} \tag{23}$$

where k_B is the Boltzmann constant and j just represent each spacial direction.

Where not quite happy yet though, the most relevant information would be heat capacity per unit volume. This can be done for macroscopically large crystals by approximating the discrete $\mathbf{k}$ sum as an integral, due to the intricacies of solid state physics this approximation becomes

$$\sum_{\mathbf{k}} \to \frac{V}{(2\pi)^3}\int d\mathbf{k} \tag{24}$$

And voilà, we get an expression for the volume we can divide away. But to make things numerically simpler we will abuse symmetry and only integrate over 1/48 part of the Brillouin zone[7]. We can then multiply by 48 do obtain the correct result. To calculate

---

[6]Yes, the author would jump of a cliff if everyone else did it

[7]If you don't know what the author talks about it is recommended that you read [2] since describing this is outside the scope of this document

the integral numerically we then go on and approximate it by a sum of unit volumes $\Delta\mathbf{k}$ times the integrand. Basically right now we are at the expression

$$\frac{C_V}{V} = \frac{1}{(2\pi)^3} \sum_j \sum_\mathbf{q} \Delta\mathbf{k} f_j(\mathbf{q}) \tag{25}$$

where f_j is the terms in equation 23 without the summation sign. Left now is to move from using the $\mathbf{k}$ and $\mathbf{q}$ representation for each $\mathbf{k}$ vector to only use $\mathbf{q}$. Ontop of that we need to take care to not over summing the $\mathbf{k}$ vectors that lies on the boundary of the Brillouin zone. There is also extra care needed to be taking since we only actually integrate over $1/48$ part of the Brillouin zone. Doing the conversion of $\mathbf{k} \to \mathbf{q}$ and caring about not over summing we get the expression

$$\frac{C_V}{V} = \frac{1}{(2\pi)^3} \frac{4}{1000} \left(\frac{\pi}{a}\right)^3 \sum_j \sum_\mathbf{q} W(\mathbf{q}) f_j(\mathbf{q}) \tag{26}$$

where $W(\mathbf{q})$ is a weight that take care of not over summing, and the $\mathbf{q}$ we are summing over are specified in the code as 48 values with weights in the beginning of the program.

2.4.2 Thermal Physics

We define the heat capacity at constant volume as

$$C_V = \left(\frac{\partial U}{\partial V}\right)_V \tag{27}$$

where U is the energy of the gas. Using the equipartition theorem we end up with an expression for the energy in a gas[4]

$$U = \frac{N f k_B T}{2} \tag{28}$$

where N is the number of particles, f is the degrees of freedom, k_B is the Boltzmann constant and T is the temperature. One can rewrite $N = nV$ where V is the volume and n is the number of particles per unit volume. Putting this together we end up with expression

$$\frac{C_V}{V} = \frac{n f k_B}{2} \tag{29}$$

Now in this report we only look at rare gases that confusingly enough are solids because of the cold temperatures we will be looking at. This yeilds six degrees of freedoms. Beyond that we also have to calculate n. Because of this we need to look at the crystal structure the rare gases we are looking at form when they go over into the solid state. This is depicted in figure 1. If we say that the length of the sides in the cube are $2a$ and think about the radius of the spherical volume that each site occupy must be the distance to the nearest neighbor (r_{nn}) divided by half we get an expression for n
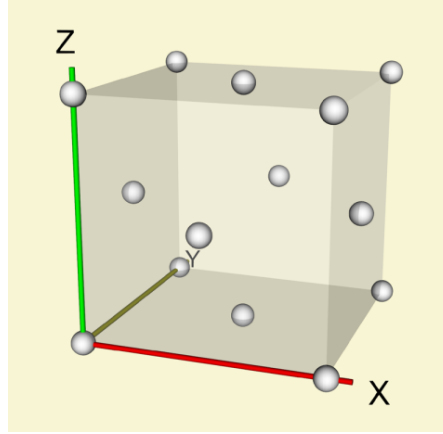
$$n = \frac{4}{(2a)^3} \tag{30}$$

Noting that $a = \frac{r_{nn}}{\sqrt{2}}$ we get

$$n = \frac{\sqrt{2}^3}{2 r_{nn}^3} \tag{31}$$

and by using this equation in equation 29 we finally get

$$\frac{C_V}{V} = \frac{3k_B\sqrt{2}^3}{2r_{nn}^3} \tag{32}$$

Figure 1 – A picture showing the fcc structure of the rare gas solids considered in this document.

3 Code

In order to obtain usuable values for the volume dependency of phonon frequencies in the rare gases discussed and the heat capacity using solid state physics theory, we have developed some code that calculate this numerically. Here we go through the code and its design.

3.1 Algorithm design

The code is written in C and has been divided into two files: `phonons.c` and `frequencies.c`. In `frequencies.c` we calculate the phonon frequencies possible in certain rare gases. This is done by solving the eigenvalue problem described by equation 8 and 19. The declaration of the frequencies function that does this calculation in this file is as follows

```
1   void frequencies(double A, double B, double m, double *q,
2                    double *omega, double *eps);
```

where `A` and `B` are the constants described by equation 10, `m` is the mass of the atoms at the sites in the lattice, `q` are the modified k vector values, `omega` and `eps` are the return values where `omega` are the frequencies in the tree different directions and `eps` are the vector corresponding to each frequency describing its direction. This is a separate function since it is essential in not only calculating the frequency but the volume dependency of the frequency as well as the heat capacity of the rare gas. Its worth to also note that the eigenvalues `omega` and its corresponding eigenvector `eps` is sorted so by eigenvalue value in descending order.

In `phonons.c` the rest of the program resides. Its main function is to use the frequency function to print usable data, calculate the volume dependency of the phonon frequencies

as well as calculating heat capacity using solid state theory of the rare gases. In order to keep track of all the different gas properties a substance structure is declared in `phonons.c` as follows

```
1  typedef struct sp{
2      double sigma;
3      double eps;
4      double rnn;
5      double m;
6  }sp;
```

where `sigma`, `eps`, `rnn` and `m` represents σ, ϵ, r and m respectively from equation 3. These different substance properties structures are then passed to different functions that calculates the volume dependency of the phonon frequencies and heat capacity of the different gases.

In order to use the `frequencies` function and write out different phonon frequencies propagating at certain positions in a crystal a function declared as

```
1  double* freqEval(sp sub, double* q)
```

has been written. The inputs are simply a substance properties structure `sub` and a `q` vector, describing for which point the frequencies should be evaluated. The function returns an array with the three frequencies at the position described by `q`. In order to calculate multiple frequencies evenly distributed between two `q` vectors a function declared as

```
1  void nEval(sp sub, double *q1, double *q2, int n, double* (*evalFunc)(sp,
       double*))
```

has been written. The inputs are the substance properties structure `sub`, two `q` vectors `q1` and `q2`, the number of `n` evenly spaced points to evaluate between `q1` and `q2` and a function pointer `evalFunc` that gets called at every point. If the frequencies is what is interesting the `freqEval` is the function passed as the last argument.

In order to calculate the volume dependency of the phonon frequencies a function declared as

```
1  double* volDepEval(sp sub, double *q)
```

has been written. The input are the substance properties structure `sub` and a vector `q` again describing the position we are interested in. The return value are the volume dependencies for the phonon frequencies. In order to calculate the volume dependency equation 21 has been used, then to estimate the derivate equation 22 is used. In order to calculate multiple frequencies evenly distributed between two `q` vectors we can use `nEval` again as done previously with `freqEval`. The difference is that the last argument in `nEval` becomes `volDepEval`.

If one instead are only interested in the heat capacity for one of the gases the function declared as

```
1  double cvEval(sp sub, double T)
```

has been written. Here the input is only the substance properties structure `sub` and the temperature in Kelvin `T`. The function returns the heat capacity. The heat capacity is calculated using equation 23 and a some predefined values for `q` and the weights that holds for all gases that the program is implemented for. Since this equation do not follow the same structure as `freqEval` and `volDepEval` a separate function to calculate multiple heat capacities between two temperatures has been written. This function is declared as

```
1  void nCvEval(sp sub, double T1, double T2, int n)
```

where the input is the substance properties structure `sub`, the two temperatures `T1` and `T2` which between the heat capacity shall be calculated in `n` points.

Outside of that the code is responsible for user input/output. Making sure that the input is correctly interpreted and that the output follows a proper output formatting that was described in the program specification. This is mainly done via the nested switch statement in the `main` function and the `printVal` function.

3.2   Typical Program Execution

When the arguments are passed to the program at execution the proper substance structure is loaded and the calculation starts. Depending on what one are interested in, phonon frequencies, volume dependency of phonon frequencies or heat capacity of the gas, different functions are called in a nested switch statement in the main function. The code has been clearly divided for the different use cases so the sturcture should be very clear for a beginner who reads the code. There is also basic error checking made to make sure that the input is properly formatted.

An example command could be

```
1  >phonons Xe cv 50 100
```

Then the program loads the substance structure related to `Xe` and the function `nCvEval` is be called so that it evaluate the heat capacity of a Xe-gas at ten points between 50 and 100 Kelvin spaced equally. `nCvEval` in turns call `cvEval` for every point that needs to be evaluated and `cvEval` calculate the heat capacity with help of the `frequencies` function and outputs

```
1   50 678837
2   55 685602
3   60 690814
4   65 694912
5   70 698189
6   75 700850
7   80 703039
8   85 704861
9   90 706393
10  95 707694
```

Jesper Vesterberg (jeve0010@student.umu.se)

3.3   Solving the Eigenvalue Problem

In order to solve the eigenvalue problem in the `frequencies` function we have used the GSL library. More specifically the function declared as

```
int gsl_eigen_symmv (gsl_matrix * A, gsl_vector * eval, gsl_matrix * evec
    , gsl_eigen_symmv_workspace * w)
```

where the the input is the matrix `A` and the output `eval` and `evec` gives the eigenvalues and eigenvectors respectively. The last input in the function `w` is nothing more than some work space for the function to do its numerical scheme. The function uses the symmetric bidiagonalization and QR reduction method described in section 8.3 of [3].

4 Results

5 Discussion

A Phonon Frequencies in Symmetry Directions

B   Volume Depdence for Phonon Frequencies in Symmetry Direction

Jesper Vesterberg (jeve0010@student.umu.se)

C   Heat Capacities for all Considered Rare Gases

D   Conclusions

References

[1] [2015-03-15] `http://en.wikipedia.org/wiki/Lennard-Jones_potential`

[2] Neil W. Ashcroft , N. David Mermin, *Solid State Physics* (Cengage Learning 1976)

[3] Gene H. Golub, Charles F. Van Loan *Matrix Computations* (JHU Press 1996)

[4] Daniel V. Schroeder *An introduction to thermal physics* (Addison Wesley 2000)