

# grupy

Written by Alex Miller in 2014 at the University of Texas at Austin

## OVERVIEW:

This program (grupy) is capable of calculating and plotting the Gruneisen parameter for a given material across a dispersion of q-points (i.e. the Brillouin zone). This is a post-processing program that requires phonon dispersion calculations to first be performed using Quantum Espresso. This program is written in python, which means you need to install python before using it. You will also need numpy and matplotlib libraries.

## THE GRUNEISEN PARAMETER:

This dimensionless value is defined as:

$$\gamma = \frac{\partial \ln(\omega)}{\partial \ln(V)}$$

Where  $\omega$  is the frequency of the phonon mode in question (i.e. the square root of the eigenvalue of the dynamical matrix) and  $V$  is the volume of the calculation.

## INSTALLATION:

- 1) Install python, numpy, and matplotlib
- 2) Change to the directory containing setup.py.
- 3) Run python setup.py install

Note: if you wish to install grupy to a particular directory (i.e. not the default Python directory) use the option:

**--prefix="path\_to\_directory"**

## BEFORE USE:

This program consists of two scripts: grupy and gruplot. Before using grupy, you must run three separate phonon calculation in Quantum Espresso. These must all be dispersion calculations (ldisp=.true. in ph.x). One calculation is performed at the relaxed (equilibrium) volume. Another is run at some volume slightly smaller than equilibrium. A third is run at some volume slightly larger than equilibrium. It is recommended these volumes be <1% below and above equilibrium volume, respectively. You may wish to do convergence tests on these finite differences.

Place the three calculations in one directory with the following structure:

```
parent directory -- |--- directory with smaller volume calculation
                   |--- directory with equilibrium volume calculation
                   |--- directory larger volume calculation
```

### USING GRUPY:

1) Write a file named *grupy.in* with the following information and place it in the parent directory shown above:

a) DIRS: names of folders containing calculations with spaces between the names

e.g.) DIRS 1.00 0.99 1.01

*Note: the format must be equilibrium then smaller then larger volume. The names must be separated by spaces only.*

b) PATH 'symmetry point 1' q-vector 1 'symmetry point 2' q-vector 2 etc

*Note: PATH does **not** have an equal sign. Also note that Quantum Espresso reads high symmetry points in terms of conventional basis vectors (see: Bilbao Crystallographic Server).*

e.g.) PATH G 0 0 0 X 0 1 0 W 0.5 1 0

2) Now you are all set up. Assuming you have the grupy scripts directory (i.e. the one containing grupy and gruplot) in your \$PATH, run the following:

a) Make the q2r.x and dynmat.x scripts for each calculation (files will be named *q2r.in* and *matdyn.in*, respectively):

**grupy --make**

b) Run those scripts automatically using the Quantum Espresso executables mentioned above:

**grupy --run**

*Note: You should watch for errors when these scripts are running!*

c) Run `grupy`:

### **grupy**

This will create a file called *prefix.grupy.out*, where *prefix* is the name of your material (i.e. whatever your Quantum Espresso prefix is). This contains a dispersion of Gruneisen parameter along the path specified in *grupy.in* for each mode in your material. Remember that this is a dimensionless variable.

*Note: the output of these files is JSON format. This is for legibility of data and easy loading into R or your favorite database. Other formatting options may follow in coming versions.*

d) *As an alternative to part c:*

### **grupy --bands**

will create a file called *prefix.grupy.bands.out*, which contains the phonon dispersion for each of the three calculations you performed. Note that frequency ( $\omega$ ) is plotted (i.e. the square root of the eigenvalue of the dynamical matrix) and these are plotted in THz. Options for other units may follow in coming versions.

You may also choose to process data from a *single* phonon dispersion without wishing to calculate dispersions for two other volumes. This can be done with:

### **grupy --bands -s name\_of\_calculation**

3) The next step is to plot the Gruneisen dispersion or the regular phonon dispersion. This will be done using python's matplotlib library. If you are running this program on a remote server, make sure you have plotting capability (e.g. running X11 via *ssh -Y yourusername@yourserver* )

Again, you will need to be in the parent folder. Once you are there...

a) Plot Gruneisen dispersion (i.e. using all three calculations):

### **gruplot**

b) Plot a particular phonon dispersion:

### **gruplot --bands name\_of\_calculation**

For example, if you label your three folders 1.00, 0.99, 1.01 and you want to plot the equilibrium bands plot:

*gruplot --bands 1.00*

#### **VERSION NOTES:**

*1.1.0:* First stable release.

*1.1.1:* Made plots look better and scrubbed old code that had been commented out. The "SPACE\_GROUP" keyword was also deemed unnecessary and removed.

*1.1.2:* Changed rejection cutoff to a lower value: this will include more data points near gamma. Changed format of data written: it is now one Gruneisen or frequency value per JSON document. Also added option to process and plot data from a single band structure calculation (i.e. without having to run 3 different calculations, which the Gruneisen parameter requires). Also added group velocity to the calculations, but no way to do anything with that data as of yet.