Calculating thermal conductivity using VASP and phono3py

Created by: Gaurav Arora, Ph.D. ([gauravarora.1100@gmail.com](mailto:gauravarora.1100@gmail.com))

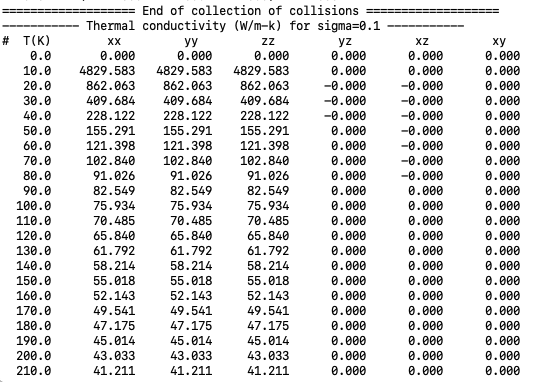
LinkedIn: [www.linkedin.com/in/gaurav-arora-1100](http://www.linkedin.com/in/gaurav-arora-1100)

Github: <https://github.com/Gaurav-Arora-MatSci>

This document includes the detail and commands adapted from different resources and combined in the form of document for easy calculation of thermal conductivity using phono3py and VASP. All the necessary steps are included. Please feel free to reach out for any comments/suggestions and if something is not working.

Modules/packages required: Python, VASP, and phono3py

Calculating thermal conductivity of pure Ni

* 1st step: Relax the structure with high precision using ISIF=3.
  + Detail of files used:
    - EDIFF = 10-8, EDIFFG = -10-5 and higher K-POINTS
    - Initial POSCAR contains only 4 atoms
    - POTCAR used in PBE, KPOINTS = MP 15 15 15
  + Result after relaxing the initial structure:
    - Lattice parameter: 3.5170999 Å
    - Energy/per atom: -5.4674 eV
* 2nd step: Generating supercells with displacements
  + Calculating force constants require generating different supercells where each atom is displaced by a tiny amount (taken care by phonopy code as described below).
  + For accurate calculation, size of the supercell should be in the range of 10 -15Å.
  + For making different POSCARS along with displacements make a different directory named “Phonopy\_cal” and copy the CONTCAR (obtained from the 1st step) into the folder. Rename the CONTCAR to POSCAR and run the below command in “Phonopy\_cal” directory.
    - phono3py -d –-dim=”3 3 3” -c POSCAR
  + Running the above command shall generate bunch of POSCAR files and the information about displacement is written in disp\_fc3.yaml file.
* 3rd step: Preparing folders for running the simulations for generated POSCAR files.
  + Make folders named “disp-000X”, where X represents the id of POSCAR file. This can be done manually or using the script (prepare.sh) mentioned at the end. Please make sure to change the variable called poc in the script depending on the number of POSCAR files generated.
  + Copy the respective POSCAR file, KPOINT, INCAR, POTCAR and script\_running\_vasp into each folder either manually or using prepare.sh script. Make sure to generate vasptun.xml file while running these calculations. Reduce the KPOINT as required as size of structure has now increased. Change IBRION=-1, NSW = 0 and ISIF = 2 in the INCAR file.
  + Submit each job either manually or using submit.sh script, mentioned at the end. Prepare your submit file according to the cluster requirement and change its name in the submit.sh script accordingly.
  + Wait for all the simulations to be done.
* 4th step: Collecting all vasprun.xml files for post-processing
  + This step is done to calculate forces from all the vapsrun.xml files. Use the command below in the same folder.
  + phono3py --cf3 disp-{00001..00109}/vasprun.xml
  + 00109 number can change according to system size and number of POSCARS made at the 2nd stage.
  + This will result in disp\_fc2.yaml, FORCES\_FC2, FORCES\_FC3, and disp\_fc3.yaml files.
    - If FORCES\_FC2 file is not generated, then separate calculations have to be performed for making fc2 dataset.
    - In this case, instead of using phono3py -d –-dim=”3 3 3” -c POSCAR command. Use the following command phono3py -d –dim-fc2=”4 4 4” –dim=”4 4 4” -c POSCAR
    - Make different folder for running simulations for fc2 and run the simulations as described above. After completion of the simulation, run phono3py –cf2 disp\_fc2-{00001..00002}/vasprun.xml
      * Where disp\_fc2-00001 being the name of the folder and change the number depending upon the number of files generated
      * This should generate fc2 set.
  + Create hdf5 files (fc2.hdf5 and fc3.hdf5) to extract thermal conductivity later using
    - phono3py --dim=”3 3 3” -c POSCAR
    - 3 3 3 is the dimension used earlier to create supercell
* 5th step: Calculate the thermal conductivity
  + phono3py --fc3 --fc2 --dim=”3 3 3” --mesh=”9 9 9” --sigma 0.1 --wgp
  + Two files named ir\_grid\_points.yaml and grid\_address-m999.hdf5 would be written.
  + Inspect for grid points to be inspected and chose the grid points for further processing by using
    - grep grid\_point ir\_grid\_points.yaml
    - write down the grid points, it should be 0 1 2 3 and so on
  + Run the following command after getting the grid points
    - phono3py --fc3 --fc2 --dim=”3 3 3” --mesh=”9 9 9” --sigma 0.1 --br --write-gamma --gp=”write all grid points using space”
      * If the program is giving error of “killed”, input 7-8 grid points once and repeat the process until all the grid points are not put into the above commands.
    - example, phono3py --fc3 --fc2 --dim=”3 3 3” --mesh=”9 9 9” --sigma 0.1 --br --write-gamma --gp=”0 1 2 3 4 78 90”
  + phono3py --fc3 --fc2 --dim="3 3 3" --mesh="9 9 9" --sigma 0.1 --br --read-gamma
  + Above command should print the conductivity matrix along with the temperature at the end as shown below
  + 
  + Copy this data and plot accordingly to get the thermal conductivity at a specific temperature.
  + This data will also be written in file named kappa-m999-s0.1.hdf5
  + Use h5py module in python to extract information about the conductivity as described on <https://phonopy.github.io/phono3py/hdf5_howto.html>
* Note: This document should be used just as a reference for getting thermal conductivity using VASP and phono3py. All the parameters chosen here are for illustration purpose only. Please update this document accordingly based on one’s expertise.

Scripts:

**Prepare.sh**

############################################

#!/bin/bash

P=`pwd`

# number of displacements

poc=9

for i in `seq 1 $poc `;

do

cd $P

mkdir disp-0000"$i"

cd disp-0000"$i"

cp ../KPOINTS .

cp ../INCAR .

cp ../POTCAR .

cp ../POSCAR-0000"$i" POSCAR

echo $i

done

poc=99

for i in `seq 10 $poc `;

do

cd $P

mkdir disp-000"$i"

cd disp-000"$i"

cp ../KPOINTS .

cp ../INCAR .

cp ../POTCAR .

cp ../POSCAR-000"$i" POSCAR

echo $i

done

poc=109

for i in `seq 100 $poc `;

do

cd $P

mkdir disp-00"$i"

cd disp-00"$i"

cp ../KPOINTS .

cp ../INCAR .

cp ../POTCAR .

cp ../POSCAR-00"$i" POSCAR

echo $i

done

**submit.sh**

##################################

#!/bin/bash

P=`pwd`

# number of displacements

poc=9

for i in `seq 1 $poc `;

do

cd $P

cd disp-0000"$i"

cp ../run\_vasp .

sbatch run\_vasp

echo $i

done

poc=99

for i in `seq 10 $poc `;

do

cd $P

cd disp-000"$i"

cp ../run\_vasp .

sbatch run\_vasp

echo $i

done

poc=109

for i in `seq 100 $poc `;

do

cd $P

cd disp-00"$i"

cp ../run\_vasp .

sbatch run\_vasp

echo $i

done