FYS-MENA4111 - lab report 2

Mikael B. Kiste

September 25, 2018

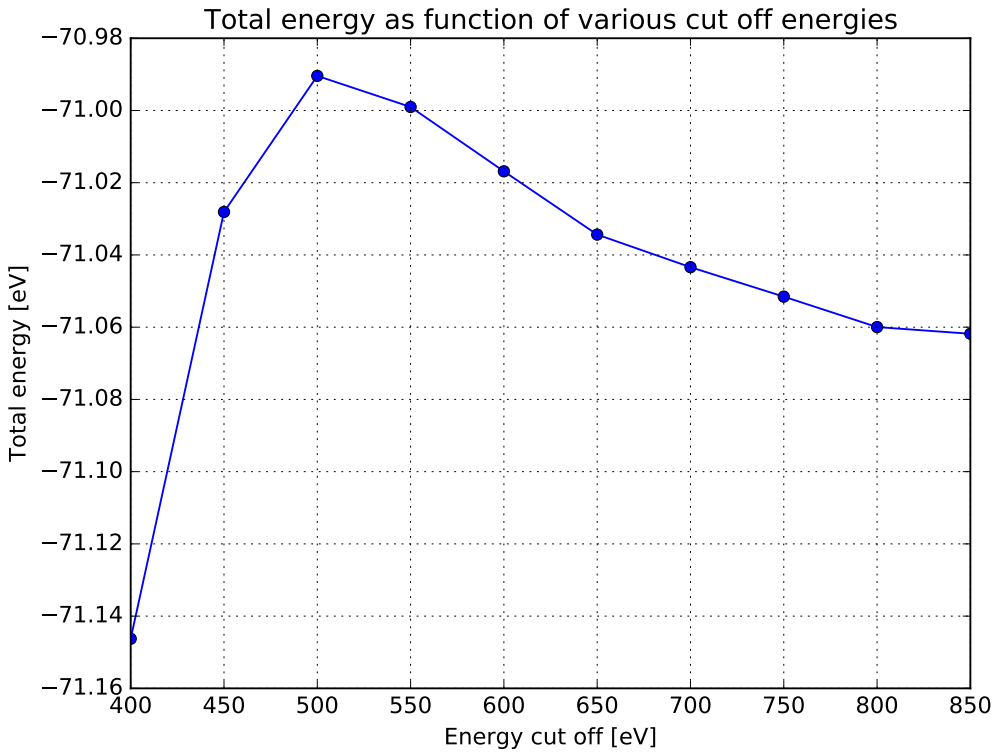
Contents

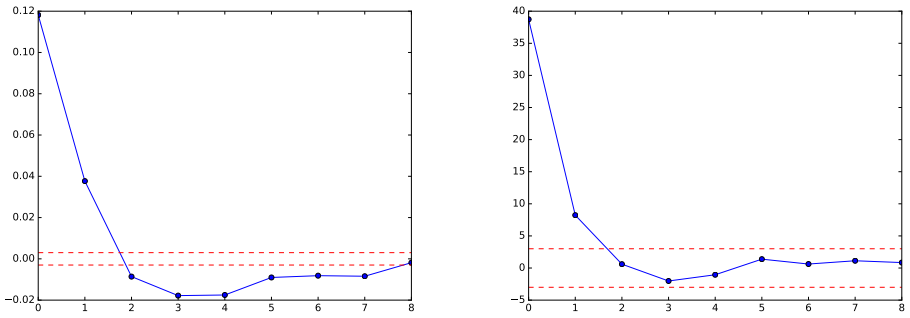
1 Introduction

In this lab we explore the cut-off energy for simulations. The cut-off limits how long the simulation will run to try and stabilize the structure before settling with the result. A too strong requirement will require a long time to compute but too weak of a requirement will not give accurate results. Therefore a balance is needed (also depending on the parameters in the specific simulation)

2 results

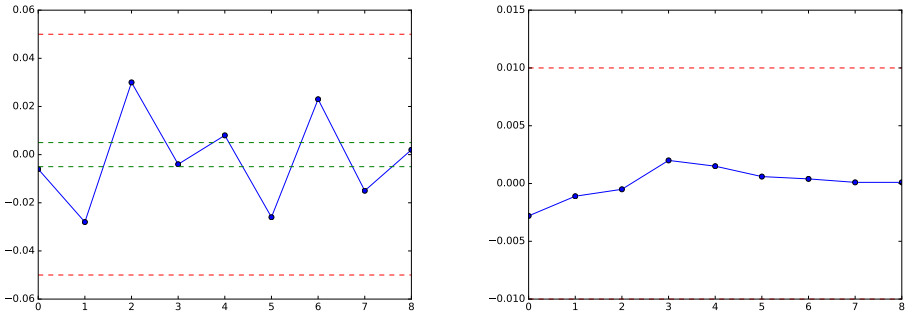
First we can show a plot of how the absolute values of the energy (comparing a system with completely seperated atoms to that of the actual structure with interatomic forces coming into play through the simulations)





(a) Difference in the energies in eV (b) Difference in the pressure in kbar

Figure 1: Here are plots of the differences in relative energies and pressures simulations with different cut-off limits. The difference is found by comparing two different simulations separated by 50ev in the cut-off. For the energies almost all points lie outside the limit interval, only the last point meets the requirement. For the pressure quite a lot of the points are within the interval



(a) Difference in the force in eV/å (b) Difference in the bandgap energies in eV

Figure 2: Here are plots of the differences in relative energies and pressures simulations with different cut-off limits. The difference is found by comparing two different simulations separated by 50ev in the cut-off. The force is safely within the limits of the weak requirement but the strong requirement (sometimes needed e.g. when including phonons) excludes many of the points. For the bandgap, it seems to be well within limits as well

2.1 results 1

Here are the results for the first simulation

MxForce	Drift	Press	TOTEN	Filename
0.604	0.007	33.45	−71.146266	./encut400/OUTCAR
0.598	0.014	72.18	−71.028093	./encut450/OUTCAR
0.570	0.000	80.41	−70.990423	./encut500/OUTCAR
0.600	0.003	81.01	−70.999028	./encut550/OUTCAR
0.596	0.007	78.99	−71.016846	./encut600/OUTCAR
0.604	0.000	77.94	−71.034363	./encut650/OUTCAR
0.578	0.000	79.32	−71.043387	./encut700/OUTCAR
0.601	0.000	79.94	−71.051542	./encut750/OUTCAR
0.586	0.077	81.07	−71.059968	./encut800/OUTCAR
0.588	0.073	81.92	−71.061823	./encut850/OUTCAR

2.2 results 2

We often want to compare one structure with itself including some minor adjustment, for example shifting one atom location, because relative measures are much more enlightening than the absolutes. Doing this

we can tell a lot more about how the structure behaves. Results 2 are from changing the position of the first Si atoms y-coordinate from 0.0 to 0.02 Ångstrøm.

MxForce	Atom #	Pressure	Drift	TOTEN	Filename
4.707	1	39.660000	0.014	−70.871523	encut400/OUTCAR
4.700	1	78.280000	0.019	−70.753411	encut450/OUTCAR
4.706	1	86.500000	0.020	−70.715520	encut500/OUTCAR
4.695	1	87.310000	0.013	−70.724627	encut550/OUTCAR
4.684	1	85.370000	0.043	−70.742064	encut600/OUTCAR
4.701	1	84.240000	0.008	−70.759853	encut650/OUTCAR
4.697	1	85.270000	0.004	−70.769200	encut700/OUTCAR
4.696	1	86.150000	0.012	−70.777371	encut750/OUTCAR
4.690	1	87.040000	0.075	−70.785657	encut800/OUTCAR
4.688	1	87.980000	0.075	−70.787516	encut850/OUTCAR

2.3 results 3

Sometimes we also care about the convergence with respect to the k point density. In these simulations we let the k-point density vary from 1 to 6. The results was as follows.

MxForce	Atom #	Pressure	Drift	TOTEN	Filename
0.605	9	32.560000	0.007	−71.134188	kdensity1/OUTCAR
0.603	9	33.760000	0.007	−71.149782	kdensity2/OUTCAR
0.604	9	33.290000	0.007	−71.144567	kdensity3/OUTCAR
0.604	9	33.450000	0.007	−71.146266	kdensity4/OUTCAR
0.604	9	33.440000	0.007	−71.146055	kdensity5/OUTCAR
0.604	9	33.350000	0.007	−71.145211	kdensity6/OUTCAR

2.4 python code

```
from numpy import *
from matplotlib.pyplot import *

#importing results
file = open("results.txt", "r")
lines = file.readlines()
file.close()
for i in range(len(lines)):
    lines[i] = lines[i].split()
lines = array(lines)

#declaring variable arrays
MxForce = lines[1:,0].astype(float)
Drift = lines[1:,1].astype(float)
Press = lines[1:,2].astype(float)
TOTEN = lines[1:,3].astype(float)
Filename = lines[1:,4]
cutOff = linspace(400,850,(850-400)/50+1)

#importing bandgaps
file = open("bandgaps.txt","r")
lines = file.readlines()
file.close
for i in range(len(lines)):
    lines[i] = lines[i].split()
lines = array(lines)

#declaring bandgap array
BandGap = lines[1:,0].astype(float)
print(BandGap)

#writing differences in total energies from one cut off to the next to file
totEnDiff = open("totEnergyDifference.txt", "w")
for i in range(len(TOTEN)-1):
    totEnDiff.write(str((TOTEN[i+1]-TOTEN[i])*1e3) + " ")
totEnDiff.close()

def plot_converg(lim,data,name):
        figure()
        style = ['--r','--g']
```

```
        plot(data[1:]-data[:-1],'-o')
        for i in range(len(lim)):
                plot(lim[i]*ones(9),style[i])
                plot(-lim[i]*ones(9),style[i])
        savefig(name)


plot_converg([3e-3],TOTEN,"energyDiff.pdf")
plot_converg([3],Press,"pressDiff.pdf")
plot_converg([5e-2,5e-3],MxForce,"forceDiff.pdf")
plot_converg([1e-2],BandGap,"bandgapDiff.pdf")
#plotting
"""
figure()
subplot(2,2,1)
plot(TOTEN[1:]-TOTEN[:-1],'-o')
plot(3e-3*ones(9),'--r')
plot(-3e-3*ones(9),'--r')
subplot(2,2,2)
plot(Press[1:]-Press[:-1],'-o')
plot(3*ones(9),'--r')
plot(-3*ones(9),'--r')
subplot(2,2,3)
plot(MxForce[1:]-MxForce[:-1],'-o')
plot(5e-2*ones(9),'--r')
plot(-5e-2*ones(9),'--r')
plot(5e-3*ones(9),'--g')
plot(-5e-3*ones(9),'--g')
subplot(2,2,4)
plot(BandGap[1:]-BandGap[:-1],'-o')
plot(1e-2*ones(9),'--r')
plot(-1e-2*ones(9),'--r')
ylim(-0.015,0.015)
savefig("converge.pdf")
"""

figure()
plot(cutOff,TOTEN,'-o')
grid()
title("Total energy as function of various cut off energies")
xlabel("Energy cut off [eV]")
ylabel("Total energy [eV]")
savefig("TOTENplot.pdf")

figure()
plot(cutOff,MxForce,'-o')
grid()
title("Maximum force as function of various cut off energies")
xlabel("Energy cut off [eV]")
ylabel("Total energy [eV]")
savefig("MxForcePlot.pdf")
```