computation of lattice parameters of UO2 for temperatures

lattice parameters of uo2 for temperatures (from 250K to 3000K; step size=125K) were computed. lammps computed each volume corresponding to each temperature.
lattice parameter was computed as fllows:

```
c = (initial volume of system)**(1/3)/5.47 # where 5.47 is an initial lattice parameter. lattice parameter at this temperature = (volume at this temperature)**(1/3) / c
```

volume at each temperature was computed with following script:

common header, script_part1.txt used in the python code below

```
#Initialization: atom_style charge units metal boundary dimension

newton
#default is on
#Atom definition:
read_data data.fcc111
mass 1 238
group 1 type 1
mass 2 16
group 2 type 2
#Force fields:
pair_style hybrid/overlay born/coul/long 10.0 morse 10.0
```

```
pair_coeff 1 1 born/coul/long 0.013806784 0.32702
3.26 0.0 0.0 10.0
pair_coeff 1 2 born/coul/long 0.013806827 0.327021
3.54 0.0 0.0 10.0
pair_coeff 1 2 morse 0.57745 1.65 2.369 10.0
pair_coeff 2 2 born/coul/long 0.013806869 0.327022
3.82 3.950633264 0.0 10.0
# the potentials above referred to [lammps-users]
Uranium dioxide, incorrect net charge of system
kspace_style pppm 1.0e-4 # use ewald method
```

the python code, lattice.py: it generates lammps scripts and executes them in a loop

```
p p p 3
on

# impose periodic boundary conditions for three
directions

# uo2 system on fcc 111 lattice
```

lattice.py:

```
import sys
from string import *
import os
from subprocess import *

for i in range(0,23):
temp=250+i*125
fn="in."+str(temp)
call("cat script_part1.txt >"+fn, shell=True)
s="log log."+str(temp)+" "

# use the header above

call("echo " +s+ ">>"+fn, shell=True)
s="velocity all create " +str(float(temp))+ "
78621 dist gaussian"
call("echo " +s+ ">>"+fn, shell=True)
```

```
s="fix NPT all npt temp " +str(float(temp))+ " "
+str(float(temp))+ " 0.005 iso 0.0 0.0 10.0 drag
0.2" call("echo " +s+ ">>"+fn, shell=True)
s="thermo 10 "
call("echo " +s+ ">>"+fn, shell=True)
call("echo " +s+ ">>"+fn, shell=True)
s="run 3000 " # number of time step
call("echo " +s+ ">>"+fn, shell=True)
call("time mpirun -np 12 lmp -sf opt < "+fn,
shell=True) # use optimization for pair_coeff
            # use 12 cores of CPU
```

How to use programs

programs are uploaded on:

https://github.com/chibaf/computing_UO2_lattice_parameters_with_lammps

```
$ python3 fcc111.py > data.fcc111
#generate UO2 lattice
$ python3 lattice.py
# run lammps for each temperatures
```

Next we compute UO2 lattice parameters for temperatures.

```
$ python3 lattice_p.py
```

ref:

2.7. Output Readers — LAMMPS documentation https://docs.lammps.org/Python_formats.html

2.6. Download the LAMMPS source with git https://docs.lammps.org/Install_git.html

remark: we need kspace package of lammps for UO2 lammps calculation

then: cmake ../cmake/ -D PKG_KSPACE=yes

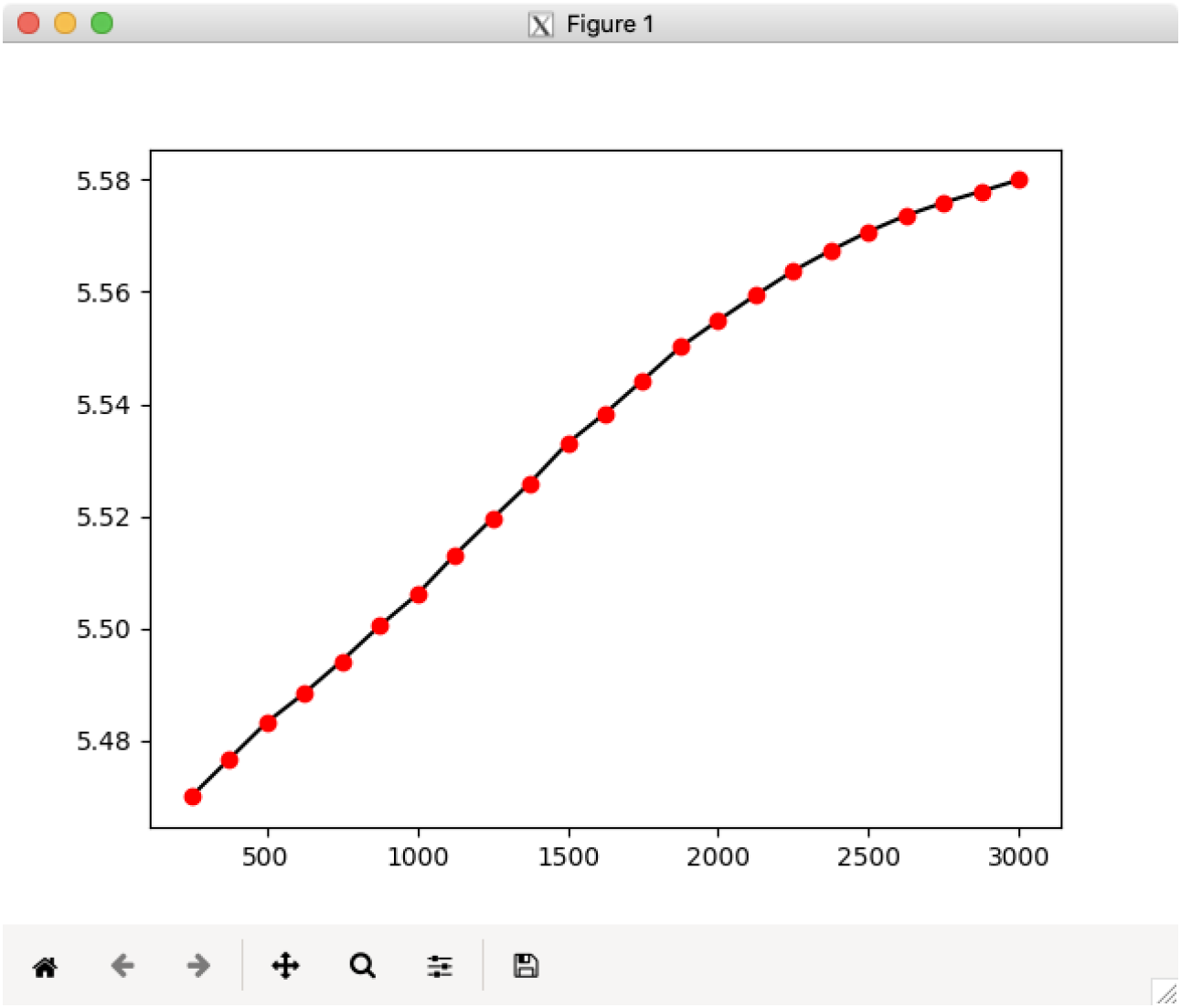
Figure 1
5.58
5.56
5.54
5.52
5.50
5.48
500
1000
1500
2000
2500
3000