

MD++ Tour with Sample Scripts

Keonwook Kang and Wei Cai

November 29, 2005

1 Finite Temperature Simulation

Relaxation introduced in the manual M04 is energy minimization scheme at 0K. It can be used not only to calculate physical properties at 0K such as vacancy formation energy, surface energy, and Peierls stress but also to give an initial structure to perform nonzero finite temperature simulation or molecular dynamics (MD) simulation. MD simulations can be categorized as microcanonical(NVE) ensemble, canonical(NVT) ensemble, and other extended ensembles according to what physical quantities are kept to be constant. Currently, MD++ can deal with NVE and NVT ensembles, which will be explained one by one.

1.1 Microcanonical(NVE) Ensemble

In the following example script,¹ the BCC molybdenum structure is initially relaxed using `relax` command and then is simulated at 300K using NVE ensemble. As long as the system's total energy does not drift, the timestep(`timestep=0.001`) is taken as big as possible. Total time steps(`totalsteps=1000`) are usually chosen large enough for the system to reach dynamical steady state. The reason why the initial temperature is doubled(`DOUBLE_T = 1`) is like this. Since the initial atomic configuration is in equilibrium at 0K by the relaxation, part of kinetic energy always goes to potential energy during finite temperature simulation and this portion is exactly same as half of the kinetic energy. ($\Delta E_{pot} = \Delta E_{kin} = \frac{3}{2}Nk_B T$ for 3D) due to the equipartition theorem. So, to get the final equilibrium temperature to be the desired one, the initial temperature is taken as twice high as the objective temperature. (See Fig.1.) Initial velocity distribution obtained by `initvelocity` does not follow Boltzman-Maxwell distribution, but during the simulation the velocity becomes redistributed and eventually shows Boltzmann-Maxwell distribution. Run the script by typing

```
$ bin/fs_gpp scripts/mo_NVE.script
```

¹uploaded as "mo_NVE.script" in the coursework website.

```

# -*-shell-script-*-
setnolog
setoverwrite
dirname = runs/mo-example
#-----
# Read the potential file
potfile = ~/Codes/MD++/potentials/mo_pot readpot
#-----
# Create Perfect Crystal
crystalstructure = body-centered-cubic
latticeconst = 3.1472      # in Angstrom for Mo
latticesize = [ 1 0 0 5
                0 1 0 5
                0 0 1 5 ]
makecrystal finalcnfile = "perf5X5X5.cn" writecn
#-----
# Plot Configuration
:
      (omitted)
#-----
# Conjugate-Gradient relaxation
conj_ftol = 1e-7      # tolerance on the residual gradient
conj_fevalmax = 1000 # max. number of iterations
conj_fixbox = 1      # fix the simulation box
relax      # CG relaxation
finalcnfile = relaxed5X5X5.cn writecn
#-----
# MD settings
T_OBJ = 300      # Desired Temperature (in Kelvin)
atommass = 95.94 # Molar Mass (g/mol)
timestep = 0.001 # (in ps)
totalsteps = 1000
DOUBLE_T = 1 # Initial temperature is set as twice high as T_OBJ
initvelocity # Initialize atoms' velocities
integrator_type = 0      # if 0, Gear 6th integrator
                    # if 1, velocity Verlet integrator
#-----
# run MD
writeall = 1
saveprop = 1      # If saveprop=0, property file not written
savepropfreq = 10 # the frequency of writing properties
outpropfile = 300K_5X5X5.out # Property File Name
openpropfile      # Open Property File
equilsteps = 0      # Only if current step > equilsteps,
                    # property files will be written
run      # Run MD simulation

```

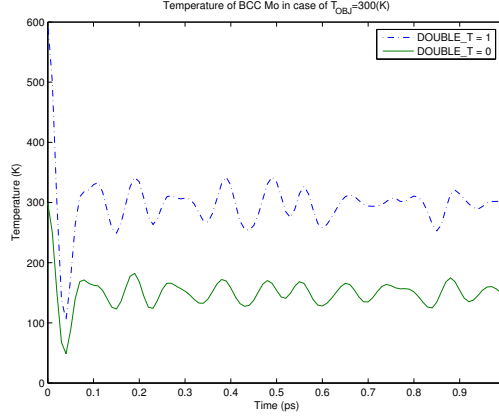


Figure 1: MD simulation of microcanonical(NVE) ensemble of BCC Mo. To obtain the desired temperature (300K), the initial temperature needs to be doubled or 600K.

```
finalcnfile = 300K_5X5X5.cn writecn
sleep quit
```

By setting `integrator_type` as 0 or 1, you can use Gear 6th integrator or velocity Verlet integrator. `writeall = 1` stores the following information in the cn file: number of atoms, the reduced coordinates, reduced velocities, individual potential energy, whether-or-not fixed, topology, atom species, and the box matrix \mathbf{H}^2 . Without declaration of `writeall = 1`, only number of atoms, the reduced coordinates, and the matrix \mathbf{H} will be saved. You can check this difference by comparing the *relaxed5X5X5.cn* file and the *300K_5X5X5.cn* file. Alternatively, `writevelocity = 1` can be used, which stores reduced coordinates and velocities. By default, only reduced coordinates and species are written in the configuration file, which will lose the kinetic information of the system. During the simulation, the properties such as current step, kinetic energy, potential energy, Virial stress, Helmholtz energy, and temperature can be written in .out file by the command `openpropfile` at every certain steps specified in `savepropfreq`. Using the matlab file³, some of properties are plotted in Fig.2.

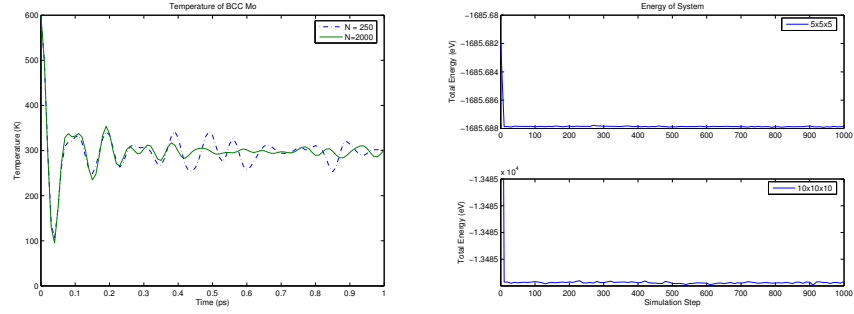
1.2 Canonical(NVT) Ensemble

In the following example script⁴, the molybdenum at 300K will be simulated using Nose-Hoover thermostat[1]. Run the script by typing

²See the manual M02_MD++.pdf for the definition of \mathbf{H} .

³Uploaded as “plotprop.m” in the coursework website.

⁴Uploaded as “mo_NVT.script” in the coursework website.



(a) Temperature

(b) Total energy

Figure 2: By changing `latticesize` in the “`mo_NVE.script`”, 10x10x10 BCC Mo ($N=2000$) is also simulated at $T_{OBJ} = 300(K)$ and (a) the temperature is plotted as simulation time together with that of 5X5X5 BCC Mo ($N=250$). It is observed that the bigger system shows smaller fluctuation at the equilibrium state. ($\sigma_T = 18.413K$ for 5X5X5 and $\sigma_T = 6.2254K$ for 10X10X10.) (b) The total energy is conserved at both cases.

```
$ bin/fs_gpp scripts/mo_NVT.script

# --shell-script--
setnolog
setoverwrite
dirname = runs/mo-example
potfile = ~/Codes/MD++/potentials/mo_pot readpot
incnfile = 300K_5X5X5.cn readcn # read cn file
#-----
# Plot Configuration
:
(omitted)
:
#-----
# MD settings
T_OBJ = 300 # Desired Temperature (in Kelvin)
atommass = 95.94 # Molar Mass (g/mol)
timestep = 0.001 # (in ps)
totalsteps = 1000
usenosehoover=1 # Use Nose-Hoover thermostat
vt2 = 1e28 # thermal reservoir
integrator_type = 0
```

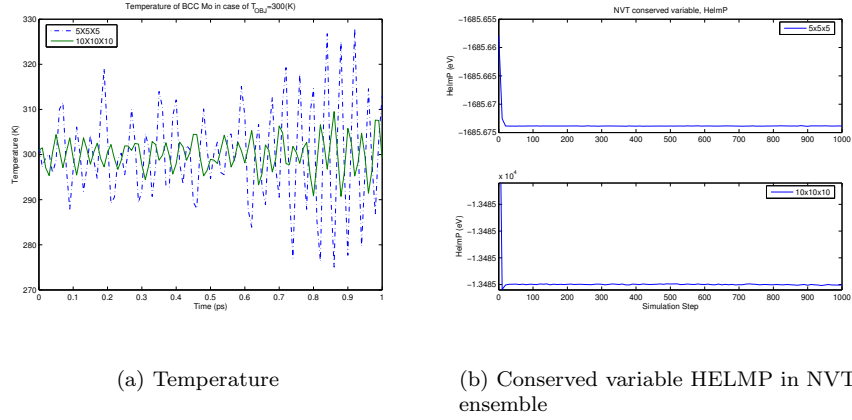


Figure 3: Using Nose-Hoover thermostat, (a)temperature ($\sigma_T = 12.87\text{K}$ for 5X5X5 and $\sigma_T = 4.268\text{K}$ for 10X10X10) and (b)HelmP of molybdenum BCC structures (5x5x5 and 10x10x10) are plotted as simulation step.

```
#-----
writeall=1 equilsteps = 0
saveprop = 1 savepropfreq = 10
outpropfile = 300K_5X5X5_NVT.out openpropfile
savecn = 1          # If savecn=0, intercn file not written
savecnfreq = 100    # the frequency of writing intercn files
intercnfile = inter.cn # Intercn File Name
openintercnfile      # Open Intercn File
run
finalcnfile = 300K_5X5X5_NVT.cn writecn
sleep quit
```

This time, instead of making BCC Mo, the configuration file “300K_5X5X5.cn” is read by the command `readcn`. When you read the cn file which was generated in the previous simulation, the timestep should be the same. For Nose-Hoover thermostat, `usenosehoover = 1` and `vt2 = 1e28` are included in the script. The bigger `vt2` is, the faster the response of the system is to keep the temperature constant. `openintercnfile` enables the configuration files to be saved during the simulation at every certain step defined in `savecnfreq`. The configuration files will be saved as *inter0000.cn*, *inter0001.cn*, and so on.

References

- [1] S. Nose, “A Molecular Dynamics Method for Simulations in the Canonical Ensemble”, *Molecular Physics*, **52** 255 (1984)