MD simulation of LJ system

MD-NpT simulation of a system of atoms interacting via a Lennard-Jones potential.

One is assumed to work in the folder **MD_LAMMPS/LJ** or in a folder at the same level (one level below **MD_LAMMPS**).

==========
PREPARATION OF INITIAL CONFIGURATION
==========

1. `../SourceF/lattice_atoms` the starting configuration of a system of LJ atoms (coordinates of atomic centers in an orthorombic lattice) is created and stored in file *`filename`*`.data`

INPUT number of atoms and center-to-center distance along the lattice X, Y, Z axes; name of file (with extension `data`)

OUTPUT file *`filename`*`.data` starting configuration (for LAMMPS)

2. VMD

VMD MAIN – EXTENSIONS - TKCONSOLE

- to generate a topology

```
topo readlammpsdata filename.data atomic
```

- to create the topology file *`filename`*`.psf`

```
animate write psf filename.psf
```

- to visualize the simulation box

```
pbc box -center origin
```

- to read the box size

```
pbc get
```

==========

MD TRAJECTORY CALCULATION

==========

`lmp_serial -in filename.inp` to run a simulation

* All input files (see below) must be in the folder !

INPUT *filename*.inp — input parameters of simulation (modifiable)
filename.data — initial configuration
or *filename*.restart — data to continue a previous trajectory

OUTPUT *filename*.dcd — atomic coordinates (trajectory coordinates)
log.lammps — thermodynamic quantities
filename.restart — atomic coordinates and information to continue a trajectory

`lmp_serial -in filename.restart` to continue a simulation

LJ.inp

what follows is a comment

in red: parameters hat can be changed

```
#-------------------------------------------------------------------------------
boundary     p p p          # boundary condition along x, y, z [p=periodic]
units        lj             #  units
atom_style   atomic         #  style: atomic, molecular, ..

read_data LJ.data           # starting configuration

# force field definition
mass         1 1.0          # atom typ and mass
pair_style   lj/cut 2.5     # LJ potential cut
pair_modify  shift yes      # shifted potential
pair_coeff   1 1 1.0 1.0    # for atom type .. and ..: epsilon, sigma

# set starting velocity
velocity     all create 0.8 87287  mom yes rot yes dist gaussian      #  temperature,
seed for random number generator, tot linear momentum =0, tot angular momentum =0, gaussian distribution

#  check of neighbor list
neighbor 0.3 bin                                #  skin length and algorithm used [bin]
neigh_modify      every 20 delay 0 check no     # build neighbor list every ... ts
timestep          0.004

fix 1 all  npt  temp 0.8 0.8 2 iso 1. 1. 5   # NPT ensemble  T_i T_f tau_T   p_i p_f tau_p

dump dumpeq all dcd 1000 LJ.dcd              # save coordinates in *.dcd file every ... ts

thermo   1000                                 # thermodynamic output every ... ts
thermo_style custom step temp press pe ke etotal density   # list of quantities in
thermodynamic output

run 50000                                     # number of ts in equilibration run

write_restart LJ.end1.restart                 # write a restart file for a next simulation
```

==========

TO VISUALIZE THE TRAJECTORY

==========

VMD

- load the topology (VMD MAIN - FILE - NEW MOLECULE - BROWSE - LOAD) *filename*.psf
- load the trajectory (BROWSE - LOAD) *filename*.dcd
- visualize the trajectory (using the selected representation)

==========

TO CALCULATE *g*(*R*)

==========

VMD

- load the topology (VMD MAIN-FILE-NEW MOLECULE- BROWSE - LOAD) *filename*.psf
- load the trajectory *filename*.dcd
- (VMD MAIN -EXTENSIONS-ANALYSIS) select RADIAL PAIR DISTRIBUTION FUNCTION g(r):

> SELECTION 1: name 1 > SELECTION 2: name 1

> FRAMES: select range of configurations to be used to calculate *g*(r)

> HISTOGRAM PARAMETERS: selection of histogram parameters

> COMPUTE g(r)

to save data: select SAVE to FILE and give name of output file *filename*.dat)

* DO NOT give the box size ('unit cell dimension'), since this is contained in the file *filename*.dcd

- data contained in *filename*.dat can be plotted, e.g. using **gnuplot**