Computer Tutorial 1: Molecular Dynamics

Nisanth N. Nair

Department of Chemistry
Indian Institute of Technology Kanpur

Course Outline

Hands on to Dynamics

- MD of small molecules (using QM)
  - Running (msindo.x)
  - Basic Analysis (gnuplot/xmgrace,VMD)

Your Own MD Code

- Algorithm
- Demo of writing an MD code
- Writing own MD code (assignment!)

Required Software/Hardware

The following instructions are based on the assumption that you have a laptop with Linux.
For downloads see `http://172.26.30.42/UMS2010/ums.html` and click on the link `Computer Tutorial-1`

Installing MSINDO

1. Download msindo.x
2. Copy/Move/Link this to ˜/bin
3. Type msindo.x in a terminal or shell or konsole. If you see the following print out, installation is complete. Type Ctr-C to exit

```
*****************************************************************************
*****************************************************************************
***                                                                       ***
***                          PROGRAM MSINDO                               ***
***

. . .
```

Required Software/Hardware (2)

GNUPLOT/XMGRACE

Gnuplot and Xmgrace are distributed freely with linux
Else look for rpm packages or go to
http://www.gnuplot.info
http://plasma-gate.weizmann.ac.il/Grace

FORTRAN/C–Compiler

- Installing Intel Fortran Compiler (ifort)
  and select `Intel Fortran Compiler Professional Edition for Linux`.
  Follow instructions in the above weblink to install.
- Installing C Compiler
  GCC is always available in your linux installation.

Required Software/Hardware (3)

VMD/MOLDEN

- MOLDEN: you may download `molden` from tutorial website and move it to ~/bin .
  If a window pops up on typing `molden` in a terminal, installation is OK.
- VMD: http://www.ks.uiuc.edu/Research/vmd/ .
  If a window pops up on typing `vmd` in a terminal, installation is OK.

Cluster Login

```
ssh ums_1@172.26.30.128
```

Hands on to Dynamics

Make an Input File

Run the Code

Generate Trajectories

Analyze

CT1: Excercise 1

- Semi–empirical code (MSINDO)
- Approximate/parametrized Hartree–Fock Quantum Mechanics to compute forces

Input File

Dowload from tutorial web link: `h2o_NVE.inp`

Running

```
msindo.x < h2o_NVE.inp > h2o_NVE.out
```

Analyzing

```
gnuplot
plot 'mddata.out' us 1:2 w l                  ⇐ T
plot 'mddata.out' us 1:4 w l                  ⇐ U
plot 'mddata.out' us 1:4 w l, '' us 1:6 w l
              ⇐ U & E
quit
```

Analyzing (...cont)

```
ln -s OH2.molden OH2.xyz
vmd -f OH2.xyz
     Display → Orthorhombic
     Graphics → Representations
              → Drawing Method → CPK
```

Points to Notice

1. Time step
2. Temperature & Potential Energy fluctations
3. Total energy conservation
4. Vibrational & rotational motion

CT1: Excercise 2

Input

Make an input that is far from equilibrium and run as before

Look at the Fluctuations

Any difference??
Compare fluctuations of bond distances in Ex 1 and 2

CT1: Home Work Set 1

1. Total energy conservation with time step 20 a.u., 40 a.u., and 60 a.u.
2. Alter coordinates of any atom by 1.0E-4 Å. Compare it with previous simulation.

Velocity Verlet Algorithm

Step 1

$$\mathbf{R}_I(t+\Delta t) = \mathbf{R}_I(t) + \dot{\mathbf{R}}_I(t)\Delta t + \frac{\mathbf{F}_I(t)}{2M_I}\Delta t^2 \quad (1)$$

Step 2

$$\dot{\mathbf{R}}_I(t+\frac{\Delta t}{2}) = \dot{\mathbf{R}}_I(t) + \frac{\mathbf{F}_I(t)}{2M_I}\Delta t \quad (2)$$

Step 3

$$\dot{\mathbf{R}}_I(t+\Delta t) = \dot{\mathbf{R}}_I(t+\frac{\Delta t}{2}) + \frac{\mathbf{F}_I(t+\Delta t)}{2M_I}\Delta t \quad (3)$$

Algorithm of MD (1)

1. Initialization

$\dot{\mathbf{R}}_I(t=0) \Leftarrow$ Random numbers
$\mathbf{R}_I(t=0) \Leftarrow$ Intuition/Expt.
$\mathbf{F}_I(t=0) \Leftarrow$ Der. of Potential at $\mathbf{R}_I(t=0)$

2. Temperature

$$T(t) = \frac{1}{k_B N_f} \sum_{I=1}^{N} M_I \dot{\mathbf{R}}_I^2(t) \tag{4}$$

Algorithm of MD (2)

3. Velocity Verlet 1

$$\begin{aligned}\mathbf{R}_I(t+\Delta t) &= \mathbf{R}_I(t) + \dot{\mathbf{R}}_I(t)\Delta t + \frac{\mathbf{F}_I(t)}{2M_I}\Delta t^2 \\ \dot{\mathbf{R}}_I(t+\frac{\Delta t}{2}) &= \dot{\mathbf{R}}_I(t) + \frac{\mathbf{F}_I(t)}{2M_I}\Delta t\end{aligned}$$

4. Update Force

$\mathbf{F}_I(t+\Delta t) \Leftarrow$ Der. of Potential at $\mathbf{R}_I(t+\Delta t)$

Algorithm of MD (3)

5. Velocity Verlet 2

$$\dot{\mathbf{R}}_I(t+\Delta t) = \dot{\mathbf{R}}_I(t+\frac{\Delta t}{2}) + \frac{\mathbf{F}_I(t+\Delta t)}{2M_I}\Delta t$$

6. Update Time and Loop

$t = t + \Delta t$
Go to step 2 or stop

Pseudo Code

```
CALL read_coordinates(r)
CALL random_velocities(v)
CALL assign_mass(m)
CALL forces(r,e,f)
DO istep=1,nstep
     CALL temperature(v,t)
     CALL velocity_verlet_1(r,v,f,dt)
     CALL forces(r,e,f)
     CALL velocity_verlet_2(r,v,f,dt)
     CALL print_data(istep,r,v,t,e)
END DO
```

Home Work Set 2

Write an MD code for the Lennard–Jones potential

$$U^{*\mathrm{LJ}}(r_{ij}^*) = 4\left[\left(\frac{1}{r_{ij}^*}\right)^{12} - \left(\frac{1}{r_{ij}^*}\right)^{6}\right] \tag{5}$$

in reduced units with any number of particles within a finite box with hard walls.
See Daan Frenkel's lecture for reduced units (or his book, page 41, 2nd ed.)

Tip 1

Generating Random Numbers

Random number uniformly between 0 and 1.
Fortran `ran(seed)`, where `seed` is a large odd integer

```
iseed=289394743
do i=1,5
     print, ran(iseed)
end do
```

Output:

```
0.8667530
0.7660332
0.1477236
0.1212504
0.6471642
```

Tip 1 (...cont)

Generating Random Numbers (...cont)

Time dependent seed

```
call itime(tarray)
iseed=1+2*(tarray(1)*tarray(2)*tarray(3))
do i=1,5
     print, ran(iseed)
end do
```

Output varies with time of execution

Generate between -1 and 1, then change $ran(iseed)$ to $2 * ran(iseed) - 1$

Tip 1 (...cont)

Generating Random Numbers (...cont)

Gaussian distribution (Box–Muller transformation: Numerical Recipies)

```
do i=1,natom
     v(i)=sqrt(-2.0*log(ran(iseed)))*
       cos(2*π*ran(iseed))
end do
```

Multiply with $\sqrt{k_\mathrm{B} T_0 / M_I}$ and rescale the velocities
Remove center of mass velocities

Tip 2

Trajectory

Print in XYZ format

name.xyz

3 ⇐ Number of atoms

⇐ Blank line

O 0.000 0.0000 0.000 ⇐ label, x, y, z coordinates in Å

H 0.900 0.0000 0.000 ⇐

H 0.000 0.9000 0.000 ⇐

Tip 3

Force

$$\nabla_{\mathbf{R}_I} R_{\mathrm{AB}} = \begin{cases} -\mathbf{e}_{\mathrm{AB}} & \text{if } I \text{ is atom A} \\ \mathbf{e}_{\mathrm{AB}} & \text{if } I \text{ is atom B} \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{6}$$

$\mathbf{e}_{\mathrm{AB}} = \mathbf{R}_{\mathrm{AB}}/R_{\mathrm{AB}}$
$\mathbf{R}_{\mathrm{AB}} = \mathbf{R}_{\mathrm{B}} - \mathbf{R}_{\mathrm{A}}$

Tip 4

Wall

For e.g.
if ($x \geq x_{+wall}$) OR ($x \leq x_{-wall}$) then $force$ =a large number

if ($x \geq x_{+wall}$) OR ($x \leq x_{-wall}$) then $velocity = -velocity$