Dive into computational physical chemistry

Lecture 3: Introduction to Molecular Dynamics Simulations and running jobs on HPC

Glen Hocky
September 20, 2022

What is a classical Molecular Dynamics (MD) Simulation?

Classical particles obey Newton's Equations of Motion

$$F = m\,a$$

But really 3N differential equations

$$F_i^x = m_i \ \frac{d^2 x_i}{dt^2}; F_i^y = m_i \ \frac{d^2 y_i}{dt^2}; F_i^z = m_i \ \frac{d^2 z_i}{dt^2}$$

What are the forces?

$$\vec{F} = -\nabla U$$

$$\Rightarrow F_i^x = -\frac{dU(x_1,y_1,z_1,\ldots,x_N,y_N,z_N)}{dx_i}, \text{etc.}$$

Good reference: https://livecomsjournal.org/index.php/livecoms/article/view/v1i1e5957

Two key variables, position (q) and momentum (p)

Momentum is "mass times velocity" – total momentum is conserved (doesn't change in time) if there are no external forces.

$$\overrightarrow{q_i} = \{x_i, y_i, z_i\}, \overrightarrow{p_i} = m_\mathrm{i} \frac{dq_i}{dt}$$

"Microstate" or "phase space point" of the system is current position and momentum: $\boldsymbol{X}(t) = \{\overrightarrow{q_1}, \overrightarrow{q_2}, \dots, \overrightarrow{q_N}, \overrightarrow{p_1}, \overrightarrow{p_2}, \dots, \overrightarrow{p_N}\}$

If following Newton's equations - knowing $U(\boldsymbol{X})$ and $\boldsymbol{X}(0)$, know everything from $\boldsymbol{X}(-\infty)$ to $\boldsymbol{X}(\infty)$

Cannot “solve” the equations

There is no formula for what will happen to the system. Instead, we start at time=0 with some initial configuration and “integrate” forward in time in small increments of time.

The essence of integrating forward is a truncated Taylor series:

$$\boldsymbol{X}(t+dt) \approx \boldsymbol{X}(t) + dt\ \frac{d\boldsymbol{X}(t)}{dt}\ (+\frac{1}{2}dt^2\ \frac{dX^2(t)}{dt^2} + \cdots)$$

$$\frac{d\boldsymbol{X}(t)}{dt} = \{\overrightarrow{v_1}, \overrightarrow{v_2}, \ldots, \overrightarrow{v_N}, \overrightarrow{F_1}, \overrightarrow{F_2}, \ldots, \overrightarrow{F_N}\}$$

Equivalent to high school physics: $d = vt, v = at$

Better integrators

In reality, better “integration” algorithms have been developed which have smaller error for fixed value of dt and sometimes other nice properties.

Example: velocity-verlet. Alternate the following, storing position and velocity

$$x(t + dt) = x(t) + v(t)dt + \frac{1}{2}a(t)dt^2$$

$$v(t + dt) = v(t) + \frac{F(t) + F(t + dt)}{2\,m}dt$$

Constant temperature

Real experiments are at constant temperature or pressure. What we really want in MD is to “sample” configurations with the correct probabilities. At constant temperature, this is given by the Boltzmann equation:

$$P(\boldsymbol{X}) \propto e^{-\left(\frac{H(\boldsymbol{X})}{k_B T}\right)}$$

where $H(\boldsymbol{X}) = U(\boldsymbol{X}) + K.E.$

What we usually care about is

$$P(q_1, \dots, q_N) \propto e^{-\frac{U(q_1, \dots, q_N)}{k_B T}}$$

Thermostats

How do we get constant temperature in our MD simulations? **Thermostats**

General strategy – increase or decrease the velocity of (some or all) particles until average kinetic energy is consistent with temperature

$$\left\langle \frac{3N}{2} m v^2 \right\rangle = \frac{3N}{2} k_B T$$

Alternative – Langevin, random forces on each atom consistent

Choosing a good thermostat very important for getting right results

Barostats similar idea, but enforce constant pressure

Forcefields

What is *U*? The potential energy function is called the **force field**

Approximation to known physical laws - bonds, angles, torsions, van der Waals forces, and coulomb forces between partial charges

E.g. for atomic biomolecules: AMBER, CHARMM, OPLS, GROMOS

$$
\begin{aligned}
V(r^N) = & \sum_{i\in\text{bonds}} k_{b\,i}(l_i - l_i^0)^2 + \sum_{i\in\text{angles}} k_{a\,i}(\theta_i - \theta_i^0)^2 \\
& + \sum_{i\in\text{torsions}} \sum_{n} \frac{1}{2} V_i^n \left[1 + \cos(n\omega_i - \gamma_i)\right] \\
& + \sum_{j=1}^{N-1} \sum_{i=j+1}^{N} f_{ij} \left\{ \epsilon_{ij} \left[\left(\frac{r_{ij}^0}{r_{ij}} \right)^{12} - 2 \left(\frac{r_{ij}^0}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right\}
\end{aligned}
$$

Constraints vs. restraints

Constraints – things that are kept constant in the molecule, such as C-H bonds, rigid water models

Restraints – restrictions on motion, such as extra harmonic interactions keeping a molecule in place during ‘equilibration’

Simulation “box”

Bunch of molecules are put in a box

Typically: **periodic boundary conditions** – particles interact with their closest periodic image. Note: This prevents **edge effects** but not **finite size effects**.

After simulation: have to **wrap molecules into box** for visualization and certain kinds of analysis

Stages of MD simulations

Minimization – lower the energy of the configuration by doing conjugate gradient, steepest descent, FIRE

Equilibration – slowly raise the temperature of the system from T=0 to the target temperature while restraining some parts of molecule. (Sometimes this isn't needed, just 'sample' velocities). Run for some amount of time to let things settle down.

Production – Long simulation starting from "equilibrated" configuration, often discard beginning to let things actually equilibrate

MD simulation software

Many MD software, some of which have specialized purposes:

- Bio systems: GROMACS, AMBER, NAMD, OPENMM, ...

- General/materials: LAMMPS, HOOMD-BLUE

Speed: these are all optimized to greater or lesser degrees to parallelize computation across many CPUs and/or on GPU

Running MD on the cluster

- **Interactive jobs**

```
srun --tasks-per-node 4 --cpus-per-task 1 --mem 8GB –t 4:00:00 --pty /bin/bash
```

- **Batch jobs – “sbatch job.sbatch”. job.sbatch:**

```
#/bin/bash
#SBATCH --job-name=run-gromacs
#SBATCH --nodes=1
#SBATCH --tasks-per-node=4
#SBATCH --mem=8GB
#SBATCH --time=04:00:00
##SBATCH --gres=gpu:1 ## To ask for a gpu, remove #, don’t need right now
module purge
module load gromacs/openmpi/intel/2020.4
mpirun gmx_mpi pdb2gmx -f 1AKI_clean.pdb -o 1AKI_processed.gro -water spce
#note, srun command replaces mpirun for multi node jobs
```

Today

1. Pull updates on comp-lab-class github page; see Week 3 assignment

https://github.com/hockyg/comp-lab-class/blob/main/Week3/Assignment.md

2. See https://sites.google.com/nyu.edu/nyu-hpc/training-support/general-hpc-topics/slurm-submitting-jobs

3. Get first experience with MD simulations using GROMACS, following tutorial