Statistical mechanics for molecular simulations (2015)

2/5/2015

Lecture 4

Molecular dynamics

* Newtonian dynamics. (equations of motion).
* Integrator / numerical propagation
* Constant-temperature ensemble.
* technical issues

In the last lecture we showed that the whole point of carrying out molecular simulations is to sample the configurational space in order to evaluate the configurational integral:

$$\langle A \rangle = \frac{1}{Z} \int dr^N \cdot e^{-\beta U(r^N)} \cdot A(r^N).$$

We described the Metropolis Monte Carlo approach that performs importance sampling through a biased random walk algorithm ⇒ Boltzmann weights is incooperated in the sampling procedure.

Besides MC, there is another means to naturally generate the Boltzmann distribution. This is the constant-temperature molecular dynamics approach.

In today's lecture, we will introduce the fundamentals of MD simulations, with an emphasis on the mathematics of "numerical integrators" and algorithms to ensure constant-temperature condition. ← more precisely, algorithms that can reproduce canonical ensemble.

* Newtonian dynamics

According to Newtonian equation of motion

$$F = ma \quad \Rightarrow \quad a(t) = \frac{1}{m} \cdot F(t)$$

for simplicity We will use a 1-D notation here; it is trivial to generalize all results to N particles here, i.e. $r(t) \to \gamma^{\mu}(t)$.

In practice, vectors & matrices are conveniently used.

obviously, we wish to solve for $\{r(t), v(t)\}$ given the $\{r(0), v(0)\}$ of a system. This is a initial value problem.

Note that the force can be calculated if the interaction potential is know:

$$F(t) = -\nabla \cdot U(r(t))$$

Numerically, this initial value problem is solved by a propagation scheme, i.e. use $r(t) \longrightarrow r(t+\delta t)$ (a short time step).

$$r(t+\delta t) = r(t) + \dot{r}(t) \cdot \delta t + \frac{1}{2} \ddot{r}(t) \cdot \delta t^2 + \cdots$$
$$= r(t) + v(t) \cdot \delta t + \frac{1}{2} a(t) \cdot \delta t^2 + \cdots$$

$$v(t+\delta t) = v(t) + a(t) \cdot \delta t + \frac{1}{2} \dot{a}(t) \cdot \delta t^2 + \cdots$$

This is a finite difference version of the differential equation.

many different methods can be applied to solve this, here we will introduce a few commonly used ones:

① Euler method (not accurate).

A simple minded algorithm is to integrate the linear term straight away:

$$r(t+\delta t) = r(t) + v(t)\cdot\delta t \qquad a(t) = \frac{-1}{m}\nabla\cdot U(r(t))$$
$$v(t+\delta t) = v(t) + a(t)\,\delta t$$

Since everything on the R.H.S. are known, $\{r(t+\delta t), v(t+\delta t)\}$ can be easily calculated, and iterated to reach the desired time.

But error $\sim \delta t^2$, often requires very small δt, and the results are not accurate.

② Verlet algorithm:

It is easy to see that a simple change yields a much better algorithm:

$$r(t+\delta t) = r(t) + v(t)\delta t + \tfrac{1}{2}a(t)\delta t^2 + \tfrac{1}{6}\dot{a}(t)\cdot\delta t^3 + O(\delta t^4)$$
$$r(t-\delta t) = r(t) - v(t)\delta t + \tfrac{1}{2}a(t)\delta t^2 - \tfrac{1}{6}\dot{a}(t)\,\delta t^3 + O(\delta t^4).$$

therefore

$$r(t+\delta t) + r(t-\delta t) = 2r(t) + a(t)\cdot\delta t^2 + O(\delta t^4)$$

$$r(t+\delta t) \simeq 2r(t) - r(t-\delta t) + a(t)\,\delta t^2$$

No $v(t)$ could be a problem if the "state" of the simulation is to be saved and continued later. And very often, $v(t)$ is needed!!

This is the Verlet algorithm. Note that only $r(t)$ is calculated and stored, no velocity. If velocity is needed, one much approximate $v(t)$ from $r(t)$:

$$v(t) = \frac{r(t+\delta t) - r(t-\delta t)}{2\delta t}$$

← only approximately, error ~ $O(\delta t^2)$

This integrator has a local error of $\sim O(\delta t^4)$ for $r(t)$. ⇒ much more accurate.

This seeming simple approach is often the most effective one!! So this is very commonly used.

③ Velocity Verlet:

The most commonly used methods for MD.

The velocity Verlet approach propagate r and v in sync:

$$r(t+\delta t) = r(t) + v(t)\,\delta t + \tfrac{1}{2} a(t)\cdot \delta t^2$$

(calculated from $r(t+\delta t)$!!)

$$v(t+\delta t) = v(t) + \tfrac{1}{2}[a(t) + a(t+\delta t)]\cdot \delta t$$

If implemented correctly, the algorithm has the same accuracy as the Verlet algorithm, normally one do

ⓐ compute $v(t+\frac{1}{2}\delta t) = v(t) + \frac{1}{2} a(t)\,\delta t$.
ⓑ " $r(t+\delta t) = r(t) + v(t+\frac{1}{2}\delta t)\,\delta t$
ⓒ " $a(t+\delta t)$ from $r(t+\delta t)$
ⓓ calculate $v(t+\delta t) = v(t+\frac{1}{2}\delta t) + \frac{1}{2} a(t+\delta t)\,\delta t$.

The accuracy of this can be verified. This method is very popular.

④ There are many other methods:
- leap frog (error $\sim O(\delta t^3)$ for $r(t)$, $\sim O(\delta t^2)$ for $v(t)$)
- Beeman's algorithm
- Gear's predictor-corrector method
- multiple time step method

} high-order methods have can use larger δt but also require more experience/empirical inputs —

Normally, the conservation of energy can be used as the test of the accuracy of integrators. Note that a good short-time integrator does not necessary yield the best long time results!!

Wait!! there is a problem here!!

So, there is a problem here. Newtonian dynamics preserves energy, meaning that we can only simulate microcanonical ensemble. NO TEMPERATURE!!

this wouldn't be a problem if the simulation box is infinite in size, but of course all simulations are only carried out with a tiny-tiny number of particles, i.e. way below thermodynamic limits ((which 10^6 to 10^{23}))

* temperature in MD simulation

In MC simulation, canonical ensemble / temperature is naturally introduced in the Boltzmann weight for the accepting probability, but in MD, the idea of temperature, or more strictly, temperature fluctuations, is not so straightforward.

A commonly accepted practice is to use the classical equal partition principle to measure temperature:

$$\langle E_K \rangle = \left\langle \frac{1}{2} m v^2 \right\rangle = \frac{1}{2} k_B T.$$

So $\langle K \rangle = \frac{3N}{2} k_B T$ for N particles.

This gives a definition of temperature in MD:

$$T(t) = \frac{1}{3N \cdot k_B} \cdot \sum_{i=1}^{3N} m_i \cdot v_i^2(t).$$

NVE

Clearly, in a ensemble, total $\langle K\rangle + \langle V\rangle$ is conserved, but T is not fixed !!

To simulate a NVT ensemble, we must allow energy (heat) to flow in/out of the system, such that T is fixed, i.e. fixed in a sense that only small fluctuations $\sim \frac{1}{\sqrt{N}}$ is allowed in T(t). This requires additional work !!

* Controlling temperature.

of course scale T(t) such that $\langle K\rangle$ is constant at all times, is not good !!

① a straighforwadd way to fix/control T(t) is add a "pull" force to rescale the velocities when the desired temperature is not reached.

Isokinetic ensemble !!

At each step when $v(t+\delta t)$ is calculated, one then set $v'(t+\delta t) = v(t+\delta t)\cdot\lambda$

with $\lambda = \sqrt{\frac{T_0}{T(t+\delta t)}}$ ← T_0 : target temp.

This is not good. It does not sample the canonical ensemble !!

② Berendsen ~~weak~~ thermostat

Berendsen proposed a more gentle velocity scaling approach: Instead of scaling to a fixed T, the scaling factor is chosen to be "milder":

$$\chi = \left(1 + \frac{\delta t}{\tau_0}\left(\frac{T_0}{T(t)} - 1\right)\right)^{1/2}$$

(T_0: target T; linear because "heat conduction"; τ_0: coupling "relaxation time", ~1 ps time scale.)

and $v'(t) = v(t) \times \chi$

This method scales velocities towards desired temperature. (Can you see this??) in an exponential factor. It controls the temperature quite effectively, however, it does not simulate the NVT ensemble!! To simulate NVT ensemble, simple velocity scaling scheme is not enough!!

In the following, we introduce two methods that do simulate canonical ensemble.

③ Andersen thermostate (NVT)

A very straightforward way to generate canonical ensemble was proposed by Andersen. The idea is to use stochastic collisions to adjust temperature. The implementation is quite simple:

collision with imaginary "heat bath" particles —

After velocity update:

frequency depending on a tunable collision rate!!

* randomly choose one particle to undergo a collision with the heat bath.
* choose a random number from a Maxwell-Boltzmann distribution with temperature T_0.
* set the velocity of the particle to the random number.

Clearly, this approach leads to Maxwell-Boltzmann velocity distribution in the long run → NVT!!

problem: discontinuous dynamics, → unphysical.

William Hoover's website:
http://williamhoover.info.

④ Nose-Hoover dynamics (NVT)

The previous methods all have its problems, and in a sense all based on ad hoc non-physics.

A more physically rigorous way to perform NVT simulation is to actually incorporate the bath dynamics!! That is, we ~~follow~~ follow what we did in the first lecture and couple a heat bath degree of freedom to our system. This imaginary reservoir allows the "extended system" to be NVE, while the subsystem (real-system) dynamics is NVT.

In the original formulation, Nose introduced a "time scaling" variable as ~~the~~ extended degree of freedom to enforce NVT. But later works removed this

scaling variable and set the whole method (basically all the same) and results in a more solid footing. This is the Nosé-Hoover method.

based on physical intuitions

Here I will present Hoover's version of this method. A mathematically rigorous derivation based on classical Lagrangian formulation is presented in Frenkel & Smit's book.

↑ Hoover's book "computational statistical mechanics" can be downloaded from his website.

Ch. 6. or Tuckerman's very nice paper (on the webpage)

The idea can be formulated with the fundamental physics that a thermal bath (think solvent!) puts frictional force on the system. This leads to new equations of motion that damps the momentum:

$$\dot{r} = \frac{p}{m}$$

$$\dot{p} = F(r) - \xi(r,p)\cdot p$$

($F(r)$: force from pots; $\xi(r,p)\cdot p$: frictional force.)

This is phenomenological, but it is general.

Now the real question is how do we find/postpone $\xi(r,p)$. In other words, we need $\dot{\xi}$.

This problem can be solved by considering the continuity equation in the phase space:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{j} = \sigma = 0$$

(ρ: density; ∇: all variables; $\vec{j}$: flux; σ: source term, $=0$: no particle generated/removed)

Note that the density distribution in the phase $\rho = \rho(r, p, \xi)$.

Therefore:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial r}(\rho \cdot \dot{r}) + \frac{\partial}{\partial p}(\rho \cdot \dot{p}) + \frac{\partial}{\partial \xi}(\rho \cdot \dot{\xi}) = 0$$

($\rho \cdot \dot{r}$: flux; $\dot{r}$: flow of r at r.)

To enforce the canonical ensemble in the dynamics, we require that the canonical probability density $\rho = f$ is steady, i.e.

f: Canonical distribution.

ζ some unknown!! (1/4)

$\frac{\partial f}{\partial t} = 0$, where we know $f = e^{-H(r,p)} \cdot g(\zeta)$

↑ Boltzman

Therefore

$$\frac{\partial(f\dot{r})}{\partial r} + \frac{\partial(f\cdot\dot{p})}{\partial p} + \frac{\partial(f\cdot\dot{\zeta})}{\partial \zeta} = 0.$$

practice from up to bottom.

derive this is a good exercise.

Now we the form of $f(r,p,\zeta)$ and the ~~Hamiltonian equations~~ equations of motion for r & p, we have

$$\frac{p}{m} \times \frac{F(r)}{k_0 T} \times f + \left\{(F(r) - \zeta p)\left(\frac{-p}{m k_0 T}\right) \cdot f - \zeta f\right\} + \dot{\zeta} \cdot f \cdot \frac{d(\ln g)}{d\zeta} = 0$$

linear in ζ ; $\ln g$ must $\sim \zeta^2$

$$\zeta \cdot \frac{p^2}{m k_0 T} - \zeta + \dot{\zeta} \cdot \frac{d(\ln g)}{d\zeta} = 0.$$

must be linear in ζ after $\frac{d}{d\zeta}$.

In order for this equation to have a steady solution, g must be a Gaussian

$$g = C \cdot e^{-\zeta^2 \tau^2/2}$$

↖ Gaussian both.

ζ is a collective coordinate!

Where τ is a relaxation time that can be adjusted to tune the coupling between system & bath, i.e. amplitude of frictional force.

Now let's derive the EOM for ζ, from the continuity equation:

$$\frac{\partial f}{\partial t} + \frac{\partial (f\cdot \dot{r})}{\partial r} + \frac{\partial (f\cdot \dot{p})}{\partial p} + \frac{\partial f\dot{\zeta})}{\partial \zeta} = 0$$

with $f(r,p,\zeta) = e^{-\beta H(r,p)} \cdot g(\zeta)$.

f is the stationary distribution that we want to have for the solution of the continuity equation.

So, $\frac{\partial f}{\partial t} = 0$.

$\frac{\partial \dot{x}}{\partial x} = \frac{\partial}{\partial x}\left(\frac{\partial x}{\partial t}\right)$
$= \frac{\partial}{\partial t}\left(\frac{\partial x}{\partial x}\right)^{=1}$
$= 0$.

We further calculate: ($H = \frac{p^2}{2m} + U(r)$)

$$\frac{\partial (f\dot{r})}{\partial r} = \dot{r}\cdot\frac{\partial f}{\partial r} = \frac{p}{m}\cdot f\cdot(-\beta)\left(\frac{\partial H}{\partial r}\right) = +\frac{\beta f\cdot p}{m} \times F(r)$$

$$= \frac{p}{m}\times\frac{F(r)}{k_B T}\times f$$

$\dot{r} = \frac{p}{m}$

$\dot{p} = F(r) - \zeta\cdot p$

note that this is not simple Hamiltonian. So $\frac{\partial \dot{p}}{\partial p} \neq 0$.

$$\frac{\partial (f\dot{p})}{\partial p} = \frac{\partial}{\partial p}\left[f\cdot(F(r)-\zeta p)\right]$$

$$= [F(r)-\zeta p]\cdot\frac{\partial f}{\partial p} + f\cdot\frac{\partial}{\partial p}[F(r)-\zeta\cdot p]$$

$$= [F(r)-\zeta p]\cdot f\cdot(-\beta)\cdot\frac{p}{m} - \zeta f$$

$$\frac{\partial (f\dot{\zeta})}{\partial \zeta} = \dot{\zeta}\cdot\left(\frac{\partial f}{\partial \zeta}\right) = \dot{\zeta}\cdot\underbrace{e^{-\beta H(r,p)}\times g(\zeta)}_{f}\times\boxed{\frac{1}{g(\zeta)}\times\frac{dg(\zeta)}{d\zeta}} = \dot{\zeta}\cdot f\cdot\frac{d\ln g}{d\zeta}$$

$= \frac{d\ln g}{d\zeta}$

finally we reach the feedback equation:

$$\dot{\zeta} = \frac{1}{\tau^2}\left[\frac{p^2}{mk_BT} - 1\right]$$

← when $p^2 < mk_BT$ for a long time, ζ can be negative!!

← quite easy → propagate.

for many particles: $\dot{\zeta} = \frac{1}{\tau^2}\cdot\sum\left[\frac{p_i^2}{mk_BT} - 1\right]$.

This equation represents a "integral feedback" that can be used to propagate ζ.

The total extended-system equations of motion are time reversible, smooth, and follows Hamilton's equations ⇒ i.e. real Newtonian dynamics.

The simple Nose-Hoover method describe here has a problem though → the bath DoF is too small, so the dynamics can be only quasi-ergodic!! This problem can be solved by attaching more bath coordinates:

⇒ Nose-Hoover chain.

recall in order to generate real Boltzmann distribution, the bath DoFs must be much more than the system DoFs. i.e. $\Omega_b \gg \Omega_s$.

NPT ensemble-

* Controlling pressure ← can be done similar to temp. control

* other issues.

Well, there are many more important issues in MD simulations, but I feel that the two thing that we covered today, i.e. propagator & constant temperature dynamics are two issues that are more difficult to understand fully. For other issues and solutions, please read the book or McCammon's review paper.

the classical mechanics part is easier to understand!!

good intro!! on the webpage!!

not technical, suitable for package runners!!

You don't need to know all the theories if you just want to run the dynamics. But, you should be curious about what exactly is going on (I suppose).

Finally: Cautious message on p. 72 of Frenkel & Smit.