Molecular simulation of complex systems

 Molecular dynamics: solve equations of motion

 Monte Carlo: importance sampling



Molecular Dynamics

Basics (4.1, 4.2, 4.3) Liouville formulation (4.3.3) Multiple timesteps (15.3) Computer experiments (4.4)

Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

Molecular dynamics

Is based on Newton's equations.

$$F_i = m_i a_i = m_i \frac{d^2 x_i(t)}{dt^2}$$

for i=1 .. N particles

the force F is given by the gradient of the potential

$$F_i = -\frac{\partial V(r^N)}{\partial r_i}$$

given the potential, one can integrate the trajectory x(t) of the whole system as a function of time.

Numerical integration

This is an N-body problem, which can only be solved numerically (except in very special cases)

$$x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^{2} + \frac{1}{6}\ddot{x}(t)\Delta t^{3} + \dots$$

at least, in principle..

Naïve implementation: truncation of Taylor expansion

$$x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^{2}$$

Wrong!

The naive "forward Euler" algorithm

- is not time reversible
- does not conserve volume in phase space
- suffers from energy drift

Better approach: "Verlet" algorithm

Verlet algorithm

compute position in next and previous time steps

$$\begin{aligned} x(t + \Delta t) &= x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 + \frac{1}{6}\ddot{x}(t)\Delta t^3 + \frac{1}{24}\ddot{x}(t)\Delta t^4 \dots \\ x(t - \Delta t) &= x(t) - \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 - \frac{1}{6}\ddot{x}(t)\Delta t^3 + \frac{1}{24}\ddot{x}(t)\Delta t^4 \dots \end{aligned}$$

$$x(t + \Delta t) + x(t - \Delta t) = 2x(t) + \ddot{x}(t)\Delta t^{2} + \mathcal{O}(\Delta t^{4})...$$

or

$$x(t + \Delta t) = 2x(t) - x(t - \Delta t) + \ddot{x}(t)\Delta t^2$$
 Verlet

Verlet algorithm

- is time reversible
- does conserve volume in phase space
- (is "symplectic")
- does not suffer from energy drift

...but is it a good algorithm?

i.e. does it predict the time evolution of the system correctly???

Molecular chaos

Dynamics of "well-behaved" classical many-body system is chaotic.

Consequence: Trajectories that differ very slightly in their initial conditions diverge exponentially ("Lyapunov instability")



Lyapunov instability

The Lyapunov disaster in action...



Any small error in the numerical integration of the equations of motion, will blow up exponentially....

always...

...and for any algorithm!!

SO:

Why should anyone believe Molecular Dynamics simulations ???

Answers:

- 1. In fact, one should not...
- 2. Good MD algorithms (e.g. Verlet) can also be considered as good Monte Carlo algorithms –they therefore yield reliable STATIC properties ("Hybrid Monte Carlo")
- 3. What is the point of simulating dynamics, if we cannot trust the resulting time-evolution???
- 4. All is well (probably), because of...The Shadow Theorem.

Shadow theorem

- For any realistic many-body system, the shadow theorem is merely a hypothesis.
- It basically states that Good algorithms generate numerical trajectories that are "close to" a REAL trajectory of the many-body system.
- Question: Does the Verlet algorithm indeed generate "shadow" trajectories?
- Take a different look at the problem.
 - Do not discretize NEWTON's equation of motion...
 - ...but discretize the ACTION

Lagrangian Classical mechanics

• Newton:
$$F_i = m_i \frac{d^2 x_i(t)}{dt^2}$$

- Lagrange:
 - Consider a system that is at a point r₀ at time t=0 and at point r_t at time t=t, then the system follows a trajectory r(t) such that:

$$S \equiv \int_0^t dt' \mathcal{L}(r(t'))$$

is an extremum. The Lagrangian L is defined as:

$$\mathcal{L}(r(t)) = K - U(r)$$
 kinetic energy

Langrangian

For example, if we use cartesian coordinates:

$$\mathcal{L}(r(t)) = \sum_{i=1}^{N} \frac{1}{2} m_i \dot{r}_i^2 - U(r_1, r_2, \dots r_N)$$

What does this mean?

Consider the "true" path R(t), with R(0)= r_0 and R(t)= r_t . Now, consider a path close to the true path:

$$r(t') = R(t') + \delta r(t')$$

Then the action S is an extremum if

$$\frac{\partial S}{\partial r(t')} = 0 \text{ for all } t$$

what does this mean?

Discretized action

$$S_{cont} = \int_{t_0}^{t_1} dt \mathcal{L}(t)$$
$$S_{disc} = \Delta t \sum_{i=0}^{i_{max}} \mathcal{L}(t_i)$$

$$\mathcal{L}(t_i) = K(t_i) - U(t_i)$$

For a one dimensional system this becomes

$$\mathcal{L}(t_i)\Delta t = \frac{1}{2}m\Delta t \frac{(x_{i+1} - x_i)^2}{\Delta t^2} - U(x_i)\Delta t$$
$$S_{disc} = \sum_{i=1}^{i_{max}} \left[\frac{m(x_{i+1} - x_i)^2}{2\Delta t} - U(x_i)\Delta t\right]$$

Minimize the action

Now do the standard thing: Find the extremum for small variations in the path, i.e. for small variations in all x_i .

$$\frac{\partial S_{disc}}{\partial x_i} = 0 \text{ for all } i$$

This will generate a discretized trajectory that starts at time t_0 at X, and ends at time t at X_t .



Minimizing the action

$$\frac{\partial S_{disc}}{\partial x_i} = \frac{\partial}{\partial x_i} \sum_{i=1}^{i_{max}} \left[\frac{m(x_{i+1} - x_i)^2}{2\Delta t} - U(x_i)\Delta t \right]$$

$$\frac{\partial S_{disc}}{\partial x_i} = \frac{-m(x_{i+1} - x_i) + m(x_i - x_{i-1})}{\Delta t} - \Delta t \frac{\partial U(x_i)}{\partial x_i}$$

$$0 = \frac{m}{\Delta t} \left(2x_i - x_{i+1} - x_{i-1} - \frac{\Delta t^2}{m} \frac{\partial U(x_i)}{\partial x_i} \right)$$

$$0 = 2x_i - x_{i+1} - x_{i-1} - \frac{\Delta t^2}{m} \frac{\partial U(x_i)}{\partial x_i}$$

$$x_{i+1} = 2x_i - x_{i-1} + \frac{\Delta t^2}{m} F(x_i)$$

- which is the Verlet algorithm!
- The Verlet algorithm generates a trajectory that satisfies the boundary conditions of a REAL trajectory –both at the beginning and at the endpoint.
- Hence, if we are interested in statistical information about the dynamics (e.g. time-correlation functions, transport coefficients, power spectra...)
 ...then a "good" MD algorithm (e.g. Verlet) is fine.

Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

Algorithm 3 (A Simple Molecular Dynamics Program)

```
simple MD program
program md
                             initialization
call init
t=0
                             MD loop
do while (t.lt.tmax)
                             determine the forces
   call force(f,en)
                             integrate equations of motion
   call integrate(f,en)
   t=t+delt
                             sample averages
   call sample
enddo
stop
end
```

Comment to this algorithm:

 Subroutines init, force, integrate, and sample will be described in Algorithms 4, 5, and 6, respectively. Subroutine sample is used to calculate averages like pressure or temperature.

Algorithm 4 (Initialization of a Molecular Dynamics Program)

```
initialization of MD program
subroutine init
sumv=0
sumv2=0
                                   \frac{3}{2}k_B T = \frac{1}{2}mv^2
do i=1, npart
                              pla
                                                      ttice
  x(i)=lattice_pos(i)
                              giv
  v(i) = (ranf() - 0.5)
  sumv=sumv+v(i)
                              velocy
                                         rer or mass
                              kir
  sumv2=sumv2+v(i) **2
                                    energy
enddo
                              velocity center of mass
sumv=sumv/npart
                              mean-squared velocity
sumv2=sumv2/npart
fs=sqrt(3*temp/sumv2)
                              scale factor of the velocities
                              set desired kinetic energy and set
do i=1, npart
                              velocity center of mass to zero
  v(i)=(v(i)-sumv)*fs
                              position previous time step
  xm(i) = x(i) - v(i) * dt
enddo
return
end
```

Algorithm 5 (Calculation of the Forces)

```
determine the force
subroutine force(f,en)
                                       and energy
en=0
do i=1,npart
   f(i) = 0
                                       set forces to zero
enddo
do i=1,npart-1
                                       loop over all pairs
  do j=i+1, npart
     xr=x(i)-x(j)
                                       periodic boundary conditions
     xr=xr-box*nint(xr/box)
     r2=xr**2
     if (r2.lt.rc2) then
                                       test cutoff
       r2i=1/r2
       r6i=r2i**3
       ff=48*r2i*r6i*(r6i-0.5)
                                       Lennard-Jones potential
                                       update force
       f(i) = f(i) + ff * xr
        f(j) = f(j) - ff * xr
                                      update energy
        en=en+4*r6i*(r6i-1)-ecut
     endif
  enddo
enddo
return
end
```

Algorithm 6 (Integrating the Equations of Motion)



Lennard Jones potentials

The Lennard-Jones potential

$$u^{LJ}(r) = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right]$$

•The truncated Lennard-Jones potential

$$u(r) = \begin{cases} u^{LJ}(r) & r \le r_c \\ 0 & r > r_c \end{cases}$$

•The truncated and shifted Lennard-Jones potential

$$u(r) = \begin{cases} u^{LJ}(r) - u^{LJ}(r_c) & r \le r_c \\ 0 & r > r_c \end{cases}$$

Phase diagrams of Lennard Jones fluids



Issues related to MD

- Initialization
 - Total momentum should be zero (no external forces)
 - Temperature rescaling to desired temperature
 - Particles/atoms/molecules start on a lattice/ or random positions
- Force calculations
 - Periodic boundary conditions
 - Straightforward force: Order N² algorithm:
 - neighbor lists, linked cell: Order N
 - Truncation and shift of the potential
 - Electrostatics: Ewald summation O(N^{1.5}) or PME: O(NlogN)
- Integrating the equations of motion
 - Controlling the temperature by a Thermostat
 - Verlet or velocity Verlet?
 - Multiple time steps

Periodic boundary conditions



Saving CPU

• Cell list

Verlet List





Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

Lagrangian approach

Lagrangian is sum of two terms

$$\mathcal{L}(\dot{r}, r) = K(\dot{r}) - U(r) = \frac{m\dot{r}^2}{2} + U(r)$$

$$\frac{\partial \mathcal{L}}{\partial \dot{r}} = \frac{\partial K}{\partial \dot{r}} = p \qquad \qquad \frac{\partial \mathcal{L}}{\partial r} = -\frac{\partial U}{\partial r} = F$$

$$p = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial \dot{r}} \qquad \qquad \dot{p} = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial r}$$
Newton : F=ma

Hamiltonian approach

The Hamiltonian is defined as

$$\begin{split} H(p,r) &= p\dot{r} - \mathcal{L}(\dot{r},r) \\ H(p^N,r^N) &= U(r^N) + \sum_i \frac{p_i^2}{2m_i} \end{split}$$

Hamilton's equations are then

$$\dot{r} = \frac{\partial \mathcal{H}(r,p)}{\partial p} = \frac{p}{m}$$

$$\dot{p} = -rac{\partial \mathcal{H}(r,p)}{\partial r} = -rac{\partial U(r^N)}{\partial r}$$

Integrating equations of motion (by Verlet) conserves the Hamiltonian

Conservation of Hamiltonian

$$dH(p,r) = \frac{\partial H}{\partial p} dp + \frac{\partial H}{\partial r} dr$$
$$\frac{\partial H}{\partial p} = \dot{r} \qquad \frac{\partial H}{\partial r} = -\dot{p}$$
$$\frac{dH(p,r)}{dt} = \frac{\partial H}{\partial p} \dot{p} + \frac{\partial H}{\partial r} \dot{r} = \dot{r}\dot{p} - \dot{p}\dot{r} = 0$$

So a solution to the Hamiltonians equation conserves the TOTAL energy

$$E = K + U$$

MD generates **NVE** ensemble

In general the MC phase space density is

$$\rho(x) = e^{-\beta \mathcal{H}(x)} / Z$$

$$Z = \int e^{-\beta \mathcal{H}(x)} dx$$

with $x = \{p^N, r^N\}$

Integrating over momenta gives

$$Z = \frac{1}{N!\Lambda^{3N}} \int e^{-\beta \mathcal{U}(r)} dr$$

N! comes from indistinguishability of particles.

But MD conserves Hamiltonian H= E = constant (and constant total P).

$$ho(x) = \delta[E - \mathcal{H}(x)]/g(E)$$
 $g(E) = \int dx \delta[E - \mathcal{H}(x)]$
with instantaneous temperature $k_B T = \sum_{i=1}^N rac{m v_i^2}{N_f}$

Thermostat: From NVE to NVT

Introduce thermostat in MD trajectory:

- stochastic thermostats
 - Andersen
 - Langevin
 - Bussi (2007)
- deterministic thermostat
 - Nose-Hoover

All of these alter the velocities such that the trajectory samples the canonical NVT ensemble, and the partition function becomes

$$Z = \frac{1}{N!\Lambda^{3N}} \int e^{-\beta \mathcal{U}(r)} dr$$

These thermostats differ in how they achieve this

Andersen Thermostat

• Every particle has a fixed probability to collide with the Andersen demon

 After collision the particle is give a new velocity

$$P(v) = \left(\frac{\beta}{2\pi m}\right)^{3/2} \exp\left[-\beta m v^2/2\right]$$



• The probabilities to collide are uncorrelated (Poisson distribution)

$$P(t;v) = v \exp\left[-vt\right]$$

Nose Hoover thermostat



Nose-Hoover Thermostat

$$\mathcal{H}_{\text{Nose}} = \sum_{i=1}^{N} \frac{p_i^2}{2ms^2} + \frac{p_s}{2Q} + U(r^N) + \frac{g}{\beta} \ln s$$

now define
$$\mathcal{H}(p',r) = \sum_{i=1}^{N} \frac{{p'_i}^2}{2m_i} + U(r^N) \qquad \qquad p' = p/s$$

then it is possible to show that the partition function Z_{nose} is $Z_{nose} \propto \frac{1}{N!} \int dp'^N dr^N \exp\left[-\beta \frac{3N+1}{g} \mathcal{H}(p',r)\right]$

for g=3N+1 the system samples the canonical distribution if p' is interpreted as the real momentum

Nose-Hoover Thermostat

$$\mathcal{H}_{\text{Nose}} = \sum_{i=1}^{N} \frac{p_i^2}{2ms^2} + \frac{p_s}{2Q} + U(r^N) + \frac{g}{\beta} \ln s$$

equations of motion follow from Hamilton's equations.

$$\frac{\mathrm{d}r_{i}}{\mathrm{d}t} = \frac{\partial \mathcal{H}_{\mathrm{Nose}}}{\partial p_{i}} = \frac{p_{i}}{ms^{2}} \qquad \frac{\mathrm{d}p_{i}}{\mathrm{d}t} = -\frac{\partial \mathcal{H}_{\mathrm{Nose}}}{\partial r_{i}} = -\frac{\partial U(r^{N})}{\partial r_{i}}$$
$$\frac{\mathrm{d}s}{\mathrm{d}t} = \frac{\partial \mathcal{H}_{\mathrm{Nose}}}{\partial p_{s}} = \frac{p_{s}}{Q} \qquad \frac{\mathrm{d}p_{s}}{\mathrm{d}t} = -\frac{\partial \mathcal{H}_{\mathrm{Nose}}}{\partial s} = \frac{1}{s} \left(\sum \frac{p_{i}^{2}}{ms^{2}} - \frac{g}{\beta}\right)$$

Effect of mass Q

Lennard-Jones fluid



mean square displacement

Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

Verlet vs Velocity Verlet

-

Verlet algorithm

$$\mathbf{r}(t + \Delta t) \approx 2\mathbf{r}(t) - \mathbf{r}(t - \Delta t) + \frac{\Delta t^2}{m}\mathbf{f}(t)$$

Downside regular verlet algorithm: velocity is not known.

Velocity verlet (Andersen 1983):

$$\mathbf{r}(t + \Delta t) \approx \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{\Delta t^2}{2m}\mathbf{f}(t)$$
$$\mathbf{v}(t + \Delta t) \approx \mathbf{v}(t) + \frac{\Delta t}{2m}[\mathbf{f}(t + \Delta t) + \mathbf{f}(t)]$$

(Is based on Trotter decomposition of Liouville operator formulation, also basis of Multiple time steps).

Liouville formulation



$$iL = iL_{r} + iL_{p} = \dot{\mathbf{r}} \frac{\partial}{\partial \mathbf{r}} + \dot{\mathbf{p}} \frac{\partial}{\partial \mathbf{p}}$$

$$f(t) = \exp(iL_{r}t)f(0)$$

$$= \exp(\dot{\mathbf{r}}(0)t\frac{\partial}{\partial \mathbf{r}})f(0)$$

$$= \exp(\dot{\mathbf{r}}(0)t\frac{\partial}{\partial \mathbf{r}})f(0)$$

$$= \sum_{n=0}^{\infty} \frac{(\dot{\mathbf{r}}(0)t)^{n}}{n!}\frac{\partial^{n}}{\partial \mathbf{r}^{n}}f(0)$$

$$= f(\mathbf{p}^{N}(0), (\mathbf{r}(0) + \dot{\mathbf{r}}(0)t)$$
Shift of coordinates
$$\mathbf{r}(0) \rightarrow \mathbf{r}(0) + \dot{\mathbf{r}}(0)t$$

$$= \dot{\mathbf{r}}(0) \rightarrow \mathbf{p}(0) + \dot{\mathbf{p}}(0)t$$

$$= f(\mathbf{p}^{N}(0) + \dot{\mathbf{r}}(0)t)$$

$$iL_{r} \Rightarrow \mathbf{r}(0) \rightarrow \mathbf{r}(0) + \dot{\mathbf{r}}(0)t$$

$$iL_{p} \Rightarrow \mathbf{p}(0) \rightarrow \mathbf{p}(0) + \dot{\mathbf{p}}(0)t$$

$$f\left(\mathbf{p}^{T} \text{ We have noncommuting operators!} \atop e^{A+B} \neq e^{A}e^{B} \atop Trotter identity} \right))),$$

$$e^{A+B} = \lim_{P \to \infty} \left(e^{A/2P}e^{B/P}e^{A/2P}\right)^{p}$$

$$e^{A+B} \approx \left(e^{A/2P}e^{B/P}e^{A/2P}\right)^{p}$$

$$\frac{A}{P} = \frac{iL_{p}t}{P} \qquad \frac{B}{P} = \frac{iL_{r}t}{P} \qquad \Delta t = \frac{t}{P}$$

$$f\left(\mathbf{p}^{N}(t), \mathbf{r}^{-}(t)\right) = \left(e^{\tau} + e^{\tau}e^{\tau} + e^{\tau}e^{\tau}\right) = \int \left(\mathbf{p}^{N}(0), \mathbf{r}^{N}(0)\right)$$

$$iL_{r}\Delta t \Rightarrow \mathbf{r} \rightarrow \mathbf{r} + \dot{\mathbf{r}}\Delta t$$

$$iL_{p}\Delta t \Rightarrow \mathbf{p} \rightarrow \mathbf{p} + \dot{\mathbf{p}}\Delta t$$

$$e^{(iL_{p}\Delta t/2)}f(\mathbf{p}^{N}(0),\mathbf{r}^{N}(0)) = f\left[\left[\mathbf{p}(0) + \frac{\Delta t}{2}\dot{\mathbf{p}}(0)\right]^{N},\mathbf{r}^{N}(0)\right]$$

$$Velocity Verlet!$$

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2m}\mathbf{F}(t)\Delta t^{2}$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \frac{\Delta t}{2m}\left[\mathbf{F}(t) + \mathbf{F}(t + \Delta t)\right]$$

$$\mathbf{p}(0) \rightarrow \mathbf{p}(0) + \frac{\Delta t}{2}\left[\dot{\mathbf{p}}(0) + \dot{\mathbf{p}}(\Delta t)\right]$$

$$\mathbf{r}(0) \rightarrow \mathbf{r}(0) + \Delta t\dot{\mathbf{r}}(\Delta t/2) = \mathbf{r}(0) + \Delta t\dot{\mathbf{r}}(0) + \frac{\Delta t^{2}}{2m}\mathbf{F}(0)$$

$$45$$

Velocity Verlet:

 $\rho(iL_p\Delta t/2)\rho(iL_r\Delta t)\rho(iL_p\Delta t/2)$

Call force(fx) Do while (t<tmax) $e^{(iL_p\Delta t/2)}: \mathbf{v}\left(t+\frac{\Delta t}{2}\right) \rightarrow \mathbf{v}\left(t\right)+\frac{\Delta t}{2m}\mathbf{f}\left(t\right)$ vx=vx+delt*fx/2 $e^{(iL_r\Delta t)}$: $\mathbf{r}(t + \Delta t) \rightarrow \mathbf{r}(t) + \Delta t \mathbf{v}(t + \Delta t/2)$ x=x+delt*vxCall force(fx) $e^{(iL_p\Delta t/2)}$: $\mathbf{v}(t + \Delta t) \rightarrow \mathbf{v}(t + \Delta t/2) + \frac{\Delta t}{2m}\mathbf{f}(t + \Delta t)$ vx=vx+delt*fx/2

enddo

Liouville Formulation

Velocity Verlet algorithm:

$$\mathbf{p}(t + \Delta t) = \mathbf{p}(t) + \frac{\Delta t}{2} \left[\dot{\mathbf{p}}(t) + \dot{\mathbf{p}}(t + \Delta t) \right]$$
$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \Delta t \dot{\mathbf{r}}(t) + \frac{\Delta t^2}{2m} \mathbf{F}(t)$$

Three subsequent coordinate transformations in either **r** or **p** of which the *Jacobian* is one: *Area preserving*

$$\mathbf{p}(t + \Delta t/2) = \mathbf{p}(t) + \frac{\Delta t}{2} \mathbf{F}(\mathbf{r}) \qquad \mathbf{p}(t + \Delta t/2) = \mathbf{p}(t + \Delta t/2) \qquad \mathbf{p}(t + \Delta t/2) = \mathbf{p}(t + \Delta t/2) \qquad \mathbf{p}(t + \Delta t) = \mathbf{p}(t + \Delta t/2) + \frac{\Delta t}{2} \mathbf{F}(\mathbf{r}(t)) \qquad \mathbf{r}(t) = \mathbf{r}(t) \qquad \mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \frac{\Delta t}{m} \mathbf{p}(t + \Delta t/2) \qquad \mathbf{r}(t) = \mathbf{r}(t) \qquad \mathbf{r}(t)$$

Other Trotter decompositions are possible!

Multiple time steps

• What to use for stiff potentials:



$$e^{i(L_{long} + L_{short} + L_{r})\Delta t} \approx e^{iL_{long}\Delta t/2} \left[e^{iL_{short}\delta t/2} e^{iL_{r}\delta t} e^{iL_{short}\delta t/2} \right]^{n} e^{iL_{long}\Delta t/2}$$
$$iL_{long}\Delta t/2 \Rightarrow v \Rightarrow v + F_{long}\Delta t/2m$$
$$iL_{short}\delta t/2 \Rightarrow v \Rightarrow v + F_{short}\delta t/2m$$
$$iL_{r}\delta t \Rightarrow r \Rightarrow r + v\delta t$$
First

$$e^{iL_{\text{long}}\Delta t/2} f[r(0), v(0)] = f[r(0), v(0) + F_{\text{long}}(0)\Delta t/2m]$$

Now *n* times:

Now *n* times:

$$\left[e^{iL_{\text{short}}\,\delta t/2}e^{iL_r\delta t}e^{iL_{\text{short}}\,\delta t/2}\right]^n f\left[r(0),v(0)+F_{\text{long}}(0)\Delta t/2m\right]$$

$$e^{(iL_{plong}\Delta t/2)}: \mathbf{v}\left(t + \frac{\Delta t}{2}\right) \rightarrow \mathbf{v}(t) + \frac{\Delta t}{2m} \mathbf{f}_{long}(t)$$
Call force (fx_1)
$$e^{iL_{long}\Delta t/2} \left[e^{iL_{short}\delta t/2} e^{iL_{short}\delta t/2}\right]^n e^{iL_{long}\Delta t/2}$$

$$vx = vx + delt * fx_1$$
Do ddt=1, n
$$e^{(iL_{pShort}\delta t/2)}: \mathbf{v}\left(t + \frac{1}{2m}\right) = iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{long}\Delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{short}\delta t/2m$$

$$iL_{r}\delta t \Rightarrow r \rightarrow r + v\delta t$$

$$vx = vx + delt * fx_short/2$$

$$e^{(iL_{r}\delta t)}: \mathbf{r}\left(t + \delta t\right) \rightarrow \mathbf{r}\left(t\right) + \delta t\mathbf{v}\left(t + \Delta t/2 + \delta t/2\right)$$

$$x = x + ddelt * call force$$

$$e^{(iL_{pShort}\delta t/2)}: \mathbf{v}\left(t + \frac{1}{2m}\right) = e^{iL_{long}\Delta t/2} \left[e^{iL_{short}\delta t/2}e^{iL_{r}\delta t}e^{iL_{short}\delta t/2}\right]^{n} e^{iL_{long}\Delta t/2}$$

$$e^{(iL_{pShort}\delta t/2)}: \mathbf{v}\left(t + \frac{1}{2m}\right) = iL_{short}\delta t/2 = v \rightarrow v + F_{long}\Delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{long}\Delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{short}\delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{short}\delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{short}\delta t/2m$$

$$iL_{short}\delta t/2 \Rightarrow v \rightarrow v + F_{short}\delta t/2m$$

```
Multiple time step, f_long is
 subroutine
                                        the long-range part and f_short
+
     multi(f_long,f_short)
                                        the short-range part of the force
                                        velocity Verlet with time step \Delta t
 vx=vx+0.5*delt*flong
                                        loop for the small time step
 do it=1,n
                                        velocity Verlet with timestep \Delta t/n
    vx=vx+0.5*(delt/n)*f_short
    x=x+(delt/n)2*vx
                                        short-range forces
    call force_short(f_short)
    vx=vx+0.5*(delt/n)*f_short
 enddo
                                        all forces
 call force_all(f_long,f_short)
 vx=vx+0.5*delt*f_long
 return
 end
```

Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

Ensemble averages by ergodicity

time averages over a NVT MD trajectory

$$\bar{A} = \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} A(t) dt$$

ensemble average

$$\langle A \rangle = \frac{\int dr^N A(r^N) \exp(-\beta \mathcal{U}(r^N))}{\int dr^N \exp(-\beta \mathcal{U}(r^N))}$$

Ergodicity theorem states that for an 'ergodic system'

$$\bar{A} = \langle A \rangle$$

Computing equilibrium properties

Ensemble averages follow from time averages

$$\langle A \rangle = \bar{A} = \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} A(t) dt$$

Temperature follows from equipartition: $\frac{1}{2} k_B T$ per d.o.f.

$$T = \frac{\left\langle 2K\right\rangle}{k_B f} \qquad \qquad K = \frac{1}{2} \sum_{i=1}^N m v_i^2$$

Where f is number of degrees of freedom

Pressure follows from virial expression

$$P = \frac{Nk_BT}{V} + \frac{1}{3V} \left\langle \sum_{i < j}^{N} \mathbf{f}_{ij} \mathbf{r}_{ij} \right\rangle$$

Transport coefficients: Diffusion

Diffusion equation (Fick's second law)

$$\frac{\partial c(x,t)}{\partial t} = D \frac{\partial^2 c(x,t)}{\partial x^2}$$

Solution for an initial $c(x,0)=\delta(0)$: all molecules at origin

$$c(x,t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(-\frac{x^2}{4Dt}\right)$$

Mean square displacement of the molecules

$$\langle x^2(t) \rangle = \frac{\int dx x^2 c(x,t)}{\int dx c(x,t)} = \int dx \frac{x^2 e^{-\frac{x^2}{4Dt}}}{\sqrt{4\pi Dt}} = 2Dt$$

Time derivative gives

$$2D = \lim_{t \to \infty} \frac{d\left\langle x^2(t)\right\rangle}{dt}$$

General c(x,t)

$$\frac{\partial}{\partial t} \int dx \ x^2 c(x,t) = D \int dx \ x^2 \frac{\partial^2 c(x,t)}{\partial x^2}.$$
$$2D = \lim_{t \to \infty} \frac{d \left\langle x^2(t) \right\rangle}{dt}$$

$$D = \frac{1}{6} \lim_{t \to \infty} \frac{d \langle \mathbf{r}^2(t) \rangle}{dt}$$

$$\Delta x(t) = \int_0^t dt' \, v_x(t').$$
 to velocity

1

1 -

Relation to velocity

$$\Delta x(t) = \int_0^t dt' v_x(t').$$

$$\langle 2D = \left\langle \left(\int_0^t dt' v_x(t') \right)^2 \right\rangle = 2^2 \right\rangle$$

$$= \int_0^t \int_0^t dt' dt'' \left\langle v_x(t') v_x(t'') \right\rangle$$

$$\langle x^2(x^2) = 2 \int_0^t \int_0^{t'} dt' dt'' \left\langle v_x(t') v_x(t'') \right\rangle.$$

$$\left\langle v_x(t')v_x(t'')\right\rangle = \left\langle v_x(t'-t'')v_x(0)\right\rangle.$$

$$\left\langle \left(\int_{0}^{t} dt' v_{x}(t') \right) \right\rangle =$$

$$= \int_{0}^{t} \int_{0}^{t} dt' dt'' \left\langle v_{x}(t')v_{x}(t'') \right\rangle$$

$$\frac{\partial \langle x^{2(t+1)} - f^{t} - f^{t'} - f$$

Green – Kubo relation

Also exists for other transport coefficients, such as viscosity and conductivity

Outline

- Basic MD
- Practical MD
- Ensembles
 - MD generates the NVE ensemble
 - The canonical NVT ensemble: thermostats
 - ensemble averages by ergodicity
- Integrating the equations of motion
 - Verlet or velocity Verlet?
 - Multiple time steps
- Computing transport properties
- Example biomolecules

All-atom force fields for biomolecules

Potential energy for protein



vdW interactions only between non-bonded |i-j|>4

Currently available empirical force fields

- CHARMm (MacKerrel et 96)
- AMBER (Cornell et al. 95)
- GROMOS (Berendsen et al 87)
- OPLS-AA (Jorgensen et al 95)
- ENCAD (Levitt et al 83)
- Subtle differences in improper torsions, scale factors 1-4 bonds, united atom rep.
- Partial charges based on empirical fits to small molecular systems
- Amber & Charmm also include ab-initio calculations
- Not clear which FF is best : top 4 mostly used
- Water models also included in description
 - TIP3P, TIP4P
 - SPC/E
- Current limit: 10⁵ atoms, 1000 ns

Photoactive Yellow Protein



Absorption of a blue-light photon triggers the photo cycle

J. Vreede et al. Biophys. J. 2005





Replica Exchange works for MD

High barriers in energy landscape: difficult to sample

Barriers effectively low: easy to sample



phase space

Consider M replica's in the NVT ensemble at a different temperature.



$$\frac{i \rightarrow j}{j \rightarrow i} = \exp \left[(\beta_i - \beta_j) (U(j) - U(i)) \right]$$

A swap between two systems of different temperatures (T_i, T_j) is accepted if their potential energies overlap.

Advantage: better sampling phase space

Partial unfolding

Loss of α-helical structure
Exposure of hydrophobic groups
Increased flexibility in parts of the protein backbone





A REMD trajectory



$$\eta = \frac{1}{Vk_BT} \int_0^\infty dt \, \left\langle \sigma^{xy}(0) \sigma^{xy}(t) \right\rangle$$
$$\sigma^{xy} = \sum_{i=1}^N \left(m_i v_i^x v_i^y + \frac{1}{2} \sum_{j \neq i} x_{ij} f_y(r_{ij}) \right)$$

$$\lambda_T = \frac{1}{Vk_B T^2} \int_0^\infty dt \ \langle j_z^e(0) j_z^e(t) \rangle$$
$$j_z^e = \frac{d}{dt} \sum_{i=1}^N z_i \frac{1}{2} \left(m_i v_i^2 + \sum_{j \neq i} v(r_{ij}) \right)$$