(Classical) Molecular Dynamics
Outline

• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications:
  – Vibrations
  – Computing transport properties
• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications
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 \( i = 1 \ldots N \) particles

The force \( F \) is given by the gradient of the potential

\[ F_i = -\frac{\partial V(r^N_i)}{\partial r_i} \]

Given the potential, one can integrate the trajectory \( x(t) \) of the whole system as a function of time.
This is an N-body problem, which can only be solved numerically (except in very special cases) at least, in principle...

\[ x(t + \Delta t) = x(t) + \dot{x}(t) \Delta t + \frac{1}{2} \ddot{x}(t) \Delta t^2 + \frac{1}{6} \dddot{x}(t) \Delta t^3 + \ldots \]

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

\[ x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 + \frac{1}{6}\dddot{x}(t)\Delta t^3 + \frac{1}{24}\dddot{\ddot{x}}(t)\Delta t^4 \ldots \]

\[ x(t - \Delta t) = x(t) - \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 - \frac{1}{6}\dddot{x}(t)\Delta t^3 + \frac{1}{24}\dddot{\ddot{x}}(t)\Delta t^4 \ldots \]

\[ x(t + \Delta t) + x(t - \Delta t) = 2x(t) + \ddot{x}(t)\Delta t^2 + \mathcal{O}(\Delta t^4) \ldots \]

Or

\[ x(t + \Delta t) = 2x(t) - x(t - \Delta t) + \dddot{x}(t)\Delta t^2 \]

Verlet algorithm:
– is time reversible
– does conserve volume in phase space, i.e., it is “symplectic” (conservation of “action element” \( dp \wedge dq \))
– 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")
Why should anyone believe Molecular Dynamics simulations ???

\[ \mathbf{r}(t) = f[\mathbf{r}^N(0), \mathbf{p}^N(0); t] \]

\[ \mathbf{r}'(t) = f[\mathbf{r}^N(0), \mathbf{p}^N(0) + \epsilon; t] \]

\[ |\Delta \mathbf{r}(t)| \sim \epsilon \exp(\lambda t) \]

\[ \epsilon \sim \Delta_{\text{max}} \exp(-\lambda t_{\text{max}}) \]
Why should anyone believe Molecular Dynamics simulations ???

Answers:

1. Good MD algorithms (e.g. Verlet) can also be considered as good (NVE!) Monte Carlo algorithm – they therefore yield reliable STATIC properties (“Hybrid Monte Carlo”)

2. What is the point of simulating dynamics, if we cannot trust the resulting time-evolution???

3. All is well (probably), because of... The Shadow Theorem.
Shadow theorem (hypothesis)
• 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?
• In practice, it follows an Hamiltonian, depending on the timestep, which is close to the real Hamiltonian $\mathcal{H}(x)$, in the sense that for $\Delta t \to 0$, $\tilde{\mathcal{H}}(x, \Delta t)$ converges to $\mathcal{H}(x)$.

• Take a different look at the problem.
  – Do not discretize NEWTON's equation of motion…
  – ...but discretize the ACTION
Lagrangian Classical mechanics

• Newton:

\[ F(x, t) = m\ddot{x} \]

• Lagrange (variational formulation of classical mechanics):

– Consider a system that is at a point \( r_0 \) at time 0 and at point \( r_t \) at time \( t \), then the system follows a trajectory \( r(t) \) such that:

\[ S = \int_{t_b}^{t_e} dt \left[ \mathcal{K} - \mathcal{U} \right] \]

is an extremum.
Lagrangian approach

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

\[ \frac{\partial \mathcal{L}}{\partial \dot{r}} = \frac{\partial K}{\partial \dot{r}} = p \]
\[ \frac{\partial \mathcal{L}}{\partial r} = -\frac{\partial U}{\partial r} = F \]

\[ p = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial \dot{r}} \]
\[ \dot{p} = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial r} \]
Lagrangian

For example, if we use Cartesian coordinates:

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

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

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

what does this mean?
Shadow trajectory

Discretized action

\[ S_{cont} = \int_{t_0}^{t_1} dt \mathcal{L}(t) \]

\[ S_{disc} = \Delta t \sum_{i=0}^{i_{max}} \mathcal{L}(t_i) \quad \mathcal{L}(t_i) = K(t_i) - U(t_i) \]

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 $X_0$, and ends at time $t$ at $X_t$. 
Minimizing the action

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

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

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

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

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

- 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.
Molecular dynamics for sampling the phase space?

Yes, but: ergodicity

\[
\rho_i(r; \mathbf{r}^N(0), \mathbf{p}^N(0), t) \rightarrow \overline{\rho_i(r)}
\]

\[
\overline{\rho_i(r)} = \lim_{t \to \infty} \frac{1}{t} \int_0^t \mathrm{d}t' \rho_i(r, t')
\]

\[
= \sum_{\text{initial conditions}} \left( \lim_{t \to \infty} \frac{1}{t} \int_0^t \mathrm{d}t' \rho_i(r, \mathbf{r}^N(0), \mathbf{p}^N(0), t') \right)
\]
Molecular dynamics for sampling the phase space?

Yes, but: ergodicity

\[ \sum_{\text{initial conditions}} f\left(\mathbf{r}^N(0), \mathbf{p}^N(0)\right) \xrightarrow{\text{number of initial conditions}} \int_{E} \frac{f\left(\mathbf{r}^N(0), \mathbf{p}^N(0)\right)}{\Omega(N, V, E)} \]

\[ \rho_i(r) = \lim_{t \to \infty} \frac{1}{t} \int_{0}^{t} dt' \rho_i(r, t') \]

\[ = \lim_{t \to \infty} \frac{1}{t} \int_{0}^{t} dt' \left\langle \rho_i(r, \mathbf{r}^N(0), \mathbf{p}^N(0), t') \right\rangle_{NVE} = \left\langle \rho_i(r) \right\rangle_{NVE} \]
1. Read essential parameters (temperature, # of atoms, time step, etc.)

2. Initialize system – positions and velocities

3. Evaluate forces

4. Integrate equations of motion

5. Stop after a given time – enough statistics for your measurement

Central loop
**Algorithm 3 (A Simple Molecular Dynamics Program)**

```
program md
  call init     \[\text{simple MD program}\]
  t=0
  do while (t lt tmax)
    call force(f,en) \[\text{initialization}\]
    call integrate(f,en)
    t=t+delt
    call sample \[\text{MD loop}\]
  enddo
  stop
end
```
subroutine init
sumv=0
sumv2=0
do i=1,npart
  x(i)=lattice_pos(i)
  v(i)=(ranf() - 0.5)
  sumv=sumv+v(i)
  sumv2=sumv2+v(i)**2
endo
sumv=sumv/npart
sumv2=sumv2/npart
fs=sqrt(3*temp/sumv2)
do i=1,npart
  v(i)=(v(i) - sumv)*fs
  x(i)=x(i) - v(i)*dt
endo
return
end
subroutine force(f,en)
    en=0
    do i=1,npart
        f(i)=0
    enddo
    do i=1,npart-1
        do j=i+1,npart
            xr=x(i)-x(j)
            xr=xr-box*nint(xr/box)
            r2=xr**2
            if (r2.lt.rc2) then
                r2i=1/r2
                r6i=r2i**3
                ff=48*r2i*r6i*(r6i-0.5)
                f(i)=f(i)+ff*xr
                f(j)=f(j)-ff*xr
            endif
            en=en+4*r6i*(r6i-1)-ecut
        enddo
    enddo
    return
end
Integrator (Verlet)

```
subroutine integrate(f,en)
    sumv=0
    sumv2=0
    do i=1,npart
        xx=2*x(i)-xm(i)+delt**2*f(i)
        vi=(xx-xm(i))/(2*delt)
        sumv=sumv+vi
        sumv2=sumv2+vi**2
        xm(i)=x(i)
        x(i)=xx
    enddo
    temp=sumv2/(3*npart)
    etot=(en+0.5*sumv2)/npart
    return
end
```

integrate equations of motion

MD loop
Verlet algorithm (4.2.3)
velocity (4.2.4)
velocity center of mass
total kinetic energy
update positions previous time
update positions current time

instantaneous temperature
total energy per particle

\[ T = \frac{\langle 2K \rangle}{k_B 3N} \]
• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications
To be taken care of, when doing MD in practice

- 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 $N^2$ algorithm:
  - neighbor lists, linked cell: Order $N$
  - Electrostatics: Ewald summation $O(N^{1.5})$

- Integrating the equations of motion
  - Controlling the temperature by a Thermostat
  - Verlet or velocity Verlet?
  - Multiple time steps
Clusters ARE different from bulk Surface! In a cube of length L, one particle per unit length:

\[
\frac{[L^3-(L-2)^3]}{L^3} \sim \frac{6L^2}{L^3}
\]
MD in practice: tuning (some) parameters

Energy fluctuations (arbitrary shifts)

CD$_3$CD$_3$

- Total energy [eV]
- Time [ps]

Verlet $\Delta t = 0.5$ fs

$\Delta t = 1$ fs

$\Delta t = 2$ fs

$\Delta t = 3$ fs

$\Delta t = 4$ fs
MD in practice: tuning (some) parameters

Energy fluctuations (arbitrary shifts)

- What is a good time step?
  - Depends on the highest vibrational frequency (thus mass) of your system \( \omega \approx \sqrt{k/M} \)
  - Typically, choose a time step corresponding to \(~1/(10\omega_{\text{max}})\) (femtosecond time scale)
Outline

• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications
Lagrangian approach

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

\[
\frac{\partial \mathcal{L}}{\partial \dot{r}} = \frac{\partial K}{\partial \dot{r}} = p
\]

\[
\frac{\partial \mathcal{L}}{\partial r} = -\frac{\partial U}{\partial r} = F
\]

\[
p = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial \dot{r}}
\]

\[
\dot{p} = \frac{\partial \mathcal{L}(\dot{r}, r)}{\partial r}
\]
The Hamiltonian is defined as

\[ H(p, r) = pr - L(\dot{r}, r) \]

\[ H(p^N, r^N) = U(r^N) + \sum_i \frac{p_i^2}{2m_i} = U + K \]

Hamilton's equations are then

\[ \dot{r} = \frac{\partial H(r, p)}{\partial p} = \frac{p}{m} \]

\[ \dot{p} = -\frac{\partial H(r, p)}{\partial r} = -\frac{\partial U(r^N)}{\partial r} \]

Integrating equations of motion (by Verlet) conserves the Hamiltonian
Conservation of the Hamiltonian

\[ dH(p,r) = \frac{\partial H}{\partial p} dp + \frac{\partial H}{\partial r} dr \]

\[ \frac{\partial H}{\partial p} = \dot{r} \quad \frac{\partial H}{\partial r} = -\dot{p} \]

\[ \frac{dH(p,r)}{dt} = \frac{\partial H}{\partial p} \dot{p} + \frac{\partial H}{\partial r} \dot{r} = \dot{r}\dot{p} - \dot{p}\dot{r} = 0 \]

A solution to Hamilton’s equations conserves the TOTAL energy

\[ E = U + K \]

MD samples the microcanonical (NVE) ensemble
Sampling the canonical ensemble: thermostats

- The idea: couple the system to a thermostat (heat bath)
- Interesting because:
  - Experiments are usually done at constant temperature
  - Better modeling of conformational changes
Introduce thermostat in MD trajectory:

- deterministic thermostat
  - Nose-Hoover (1984)
  - Nose-Hoover chains (Martyna et al. 1992)

- stochastic thermostats
  - Andersen
  - Langevin
  - Nose-Hoover-Langevin (Leimkuhler et al, 2009)
  - Stochastic Velocity rescaling (Bussi-Donadio-Parrinello, 2007)

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

\[
Z = \frac{1}{N!V^3N} \int e^{-\beta U(r)} dr
\]

These thermostats differ in how they achieve this
Sampling the canonical ensemble: thermostats

Probability distribution of the kinetic energy:

\[ P(E_{kin}) \propto \exp(-E_{kin}/k_B T) \]

The Maxwell-Boltzmann-Distribution is given by:

\[ T < T < T \]

The kinetic energy is given by \( p^2/2M \).

The temperature is given by:

\[ T = \frac{2\langle E_{kin} \rangle}{3Nk_B} \]

where \( N \) is the number of particles.
Andersen thermostat

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

• After collision the particle is given 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)

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

• Downside: momentum not conserved.
Nosé thermostat

goal: compute MD trajectory sampling NVT ensemble.
Take kinetic energy out of the system and store it into a reservoir
The reservoir can be seen as additional variable $s$ that “stores” kinetic energy
Approach: extended phase space

\[
\mathcal{H}_N = \sum_{i=1}^{N} \frac{p_i^2}{2m_is^2} + U(r_1, \ldots, r_N) + \frac{p_s^2}{2Q} + gkT \ln s,
\]

Original system

extended variable

effective mass

Fictitious oscillator

\[
\Omega = \int d^N r \, d^N p \, ds \, dp_s \, \delta \left( \sum_{i=1}^{N} \frac{p_i^2}{2m_is^2} + U(r_1, \ldots, r_N) + \frac{p_s^2}{2Q} + gkT \ln s - E \right)
\]

Change of variable:

\[
\tilde{p}_i = \frac{p_i}{s}
\]
Justification of Nosé thermostat

\[ \Omega = \int d^N \mathbf{r} \ d^N \tilde{\mathbf{p}} \ ds \ dp_s \ s^{dN} \delta \left( \sum_{i=1}^{N} \frac{\tilde{p}_i^2}{2m_i} + U(r_1, \ldots, r_N) + \frac{p_s^2}{2Q} + gkT \ln s - E \right) \]

\[ = \int d^N \mathbf{r} \ d^N \mathbf{p} \ ds \ dp_s \ s^{dN} \delta \left( \mathcal{H}(\mathbf{r}, \mathbf{p}) + \frac{p_s^2}{2Q} + gkT \ln s - E \right) \]

Hamiltonian of the atomic (sub)system

\[ \mathcal{H} = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} + U(r_1, \ldots, r_N) \]

\[ \delta(f(s)) = \frac{\delta(s - s_0)}{|f'(s_0)|} \quad f(s) = \mathcal{H}(\mathbf{r}, \mathbf{p}) + \frac{p_s^2}{2Q} + gkT \ln s - E \]

\[ f(s_0) = 0: \quad s_0 = \frac{e^{(E - \mathcal{H}(\mathbf{r}, \mathbf{p}) - \frac{p_s^2}{2Q})/gkT}}{f'(s_0)} \Rightarrow \frac{1}{|f'(s_0)|} = \frac{1}{gkT} e^{(E - \mathcal{H}(\mathbf{r}, \mathbf{p}) - \frac{p_s^2}{2Q})/gkT} \]

\[ \Omega = \frac{1}{gkT} \int d^N \mathbf{p} \ d^N \mathbf{r} \ dp_s \ e^{(dN+1)(E - \mathcal{H}(\mathbf{r}, \mathbf{p}) - \frac{p_s^2}{2Q})/gkT} \quad \text{if:} \quad g = dN + 1 \]

\[ \Omega = \frac{e^{E/kT} \sqrt{2\pi QkT}}{(dN + 1)kT} \int d^N \mathbf{p} \ d^N \mathbf{r} \ e^{-\mathcal{H}(\mathbf{r}, \mathbf{p})/kT} \]

Which is (proportional to) the canonical partition function for

\[ \mathcal{H} = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} + U(r_1, \ldots, r_N) \]
Nosé thermostat: Effect of mass $Q$

Lennard-Jones fluid

Mean square displacement

Temperature relaxation
Stochastic velocity rescaling thermostat


Combine concepts from velocity rescaling (fast!) with concepts from stochastic thermostats (accurate!)

Target temperature follows a stochastic differential equation:

$$\frac{dT}{\bar{T}} = \left[1 - \frac{T(t)}{\bar{T}}\right] \frac{dt}{\tau} - 2\sqrt{\frac{T(t)}{3\bar{T}N\tau}} \xi(t)$$

Stochastic velocity rescaling thermostat


Combine concepts from velocity rescaling (fast!) with concepts from stochastic thermostats (accurate!)

Target temperature follows a stochastic differential equation:

\[
\frac{dT}{\bar{T}} = \left[1 - \frac{T(t)}{\bar{T}}\right] \frac{dt}{\tau} - 2\sqrt{\frac{T(t)}{3\bar{T}N\tau}} \xi(t)
\]

- Temperature rescaling
- White noise

- Very successful thermostat, weakly dependent on relaxation time \( T \)
- Pseudo-Hamiltonian is conserved

Outline

• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications
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, and modern thermostats such as the Nose-Hoover-Langevin.
Liouville formulation

\[ a(x) \quad \frac{da}{dt} = \sum_{\alpha=1}^{3N} \left[ \frac{\partial a}{\partial q_\alpha} \dot{q}_\alpha + \frac{\partial a}{\partial p_\alpha} \dot{p}_\alpha \right] \]

(implicit dependence on \( t \))

\[ \frac{da}{dt} = \sum_{\alpha=1}^{3N} \left[ \frac{\partial a}{\partial q_\alpha} \frac{\partial H}{\partial p_\alpha} - \frac{\partial a}{\partial p_\alpha} \frac{\partial H}{\partial q_\alpha} \right] \]

Formal solution (useless, unless...)

\[ a(x_t) = e^{iLt}a(x_0) \]

A choice for \( a(x) \)

\[ a(x) = x \quad \Rightarrow \quad x_t = e^{iLt}x_0 \]

\[ iL = iL_1 + iL_2 \quad iL_1 = \sum_{\alpha=1}^{N} \frac{\partial H}{\partial p_\alpha} \frac{\partial}{\partial q_\alpha} \quad iL_2 = -\sum_{\alpha=1}^{N} \frac{\partial H}{\partial q_\alpha} \frac{\partial}{\partial p_\alpha} \]

\[ iL_1 iL_2 \phi(x) \neq iL_2 iL_1 \phi(x) \]
Liouville formulation

\[ iL_1 = \sum_{\alpha=1}^{N} \frac{\partial \mathcal{H}}{\partial p_{\alpha}} \frac{\partial}{\partial q_{\alpha}} \quad \text{and} \quad iL_2 = - \sum_{\alpha=1}^{N} \frac{\partial \mathcal{H}}{\partial q_{\alpha}} \frac{\partial}{\partial p_{\alpha}} \]

\[ \mathcal{H} = \frac{p^2}{2m} + U(x) \quad iL_1 = \frac{p}{m} \frac{\partial}{\partial x} \quad iL_2 = F(x) \frac{\partial}{\partial p} \quad F(x) = -\frac{dU}{dx} \]

\( L_1 \) and \( L_2 \) do not commute:

\[ iL_1 iL_2 \phi(x,p) = \frac{p}{m} \frac{\partial}{\partial x} F(x) \frac{\partial}{\partial p} \phi(x,p) = \frac{p}{m} F(x) \frac{\partial^2 \phi}{\partial p \partial x} + \frac{p}{m} F'(x) \frac{\partial \phi}{\partial p} \]

\[ iL_2 iL_1 \phi(x,p) = F(x) \frac{\partial}{\partial p} \frac{p}{m} \frac{\partial}{\partial x} \phi(x,p) = F(x) \frac{p}{m} \frac{\partial^2 \phi}{\partial p \partial x} + F(x) \frac{1}{m} \frac{\partial \phi}{\partial x} \]

\[ [iL_1, iL_2] \phi(x,p) = \frac{p}{m} F'(x) \frac{\partial \phi}{\partial p} - \frac{F(x)}{m} \frac{\partial \phi}{\partial x} \]
Trotter decomposition in general:

\[ [A, B] \neq 0 \quad \Rightarrow \quad e^{A+B} = \lim_{P \to \infty} \left[ e^{B/2P} e^{A/P} e^{B/2P} \right]^P \]

For \( iL = iL_1 + iL_2 \):

\[ e^{iLt} = e^{(iL_1+iL_2)t} = \lim_{P \to \infty} \left[ e^{iL_2t/2P} e^{iL_1t/P} e^{iL_2t/2P} \right]^P \]

Introducing the small, discrete time step \( \Delta t \):

\[ \Delta t = t/P \quad e^{iLt} = \lim_{P \to \infty, \Delta t \to 0} \left[ e^{iL_2\Delta t/2} e^{iL_1\Delta t} e^{iL_2\Delta t/2} \right]^P \]

\[ e^{iLt} \approx \left[ e^{iL_2\Delta t/2} e^{iL_1\Delta t} e^{iL_2\Delta t/2} \right]^P + \mathcal{O} \left( P \Delta t^3 \right) \]

Since \( P = t/\Delta t \), then the error at time \( t \) goes as \( \Delta t^2 \).

While, per time step:

\[ e^{iL\Delta t} \approx e^{iL_2\Delta t/2} e^{iL_1\Delta t} e^{iL_2\Delta t/2} + \mathcal{O} \left( \Delta t^3 \right) \]
Let's start from a classical Hamiltonian

\[ \mathcal{H} = \frac{p^2}{2m} + U(x) \]

\[ e^{iL_1\Delta t} \approx e^{iL_2\Delta t/2} e^{iL_1\Delta t} e^{iL_2\Delta t/2} + \mathcal{O}(\Delta t^3) \]

\[ iL_1 = \frac{p}{m} \frac{\partial}{\partial x} \quad iL_2 = F(x) \frac{\partial}{\partial p} \]

\[ \exp(iL_1\Delta t) \approx \exp \left( \frac{\Delta t}{2} F(x) \frac{\partial}{\partial p} \right) \exp \left( \Delta t \frac{p}{m} \frac{\partial}{\partial x} \right) \exp \left( \frac{\Delta t}{2} F(x) \frac{\partial}{\partial p} \right) \]

\[ \begin{pmatrix} x(\Delta t) \\ p(\Delta t) \end{pmatrix} \approx \exp \left( \frac{\Delta t}{2} F(x(0)) \frac{\partial}{\partial p(0)} \right) \times \exp \left( \Delta t \frac{p(0)}{m} \frac{\partial}{\partial x(0)} \right) \times \exp \left( \frac{\Delta t}{2} F(x(0)) \frac{\partial}{\partial p(0)} \right) \begin{pmatrix} x(0) \\ p(0) \end{pmatrix} \]

Evaluation of the exponential operator:

\[ \exp \left( c \frac{\partial}{\partial x} \right) g(x) = \sum_{k=0}^{\infty} \frac{1}{k!} \left( c \frac{\partial}{\partial x} \right)^k g(x) = \sum_{k=0}^{\infty} \frac{1}{k!} c^k g^{(k)}(x) \]

\[ \Rightarrow \quad \exp \left( c \frac{\partial}{\partial x} \right) g(x) = g(x + c) \]
Liouville formulation: integrator for classical Hamiltonian

\[
\begin{pmatrix}
  x(\Delta t) \\
  p(\Delta t)
\end{pmatrix}
\approx \exp \left( \frac{\Delta t}{2} F(x(0)) \frac{\partial}{\partial p(0)} \right) \times \exp \left( \Delta t \frac{p(0)}{m} \frac{\partial}{\partial x(0)} \right) \times \exp \left( \frac{\Delta t}{2} F(x(0)) \frac{\partial}{\partial p(0)} \right)
\begin{pmatrix}
  x(0) \\
  p(0)
\end{pmatrix}
\]

\[
\exp \left( c \frac{\partial}{\partial x} \right) g(x) = g(x + c)
\]

Stepwise application of the Liouville operator for a classical Hamiltonian

\[
\exp \left( \frac{\Delta t}{2} F(x) \frac{\partial}{\partial p} \right) \begin{pmatrix}
  x \\
  p
\end{pmatrix}
= \begin{pmatrix}
  x \\
  p + \frac{\Delta t}{2} F(x)
\end{pmatrix}
\]

\[
\exp \left( \Delta t \frac{p}{m} \frac{\partial}{\partial x} \right) \begin{pmatrix}
  x \\
  p + \frac{\Delta t}{2} F(x)
\end{pmatrix}
= \begin{pmatrix}
  x + \Delta t \frac{p}{m} \\
  p + \frac{\Delta t}{2} F(x + \Delta t \frac{p}{m})
\end{pmatrix}
\]

\[
\exp \left( \Delta t F(x) \frac{\partial}{\partial p} \right) \begin{pmatrix}
  x + \Delta t \frac{p}{m} \\
  p + \frac{\Delta t}{2} F(x + \Delta t \frac{p}{m})
\end{pmatrix}
= \begin{pmatrix}
  x + \frac{\Delta t}{m} \left( p + \frac{\Delta t}{2} F(x) \right) \\
  p + \frac{\Delta t}{2} F(x) + \frac{\Delta t}{2} F \left[ x + \frac{\Delta t}{m} \left( p + \frac{\Delta t}{2} F(x) \right) \right]
\end{pmatrix}
\]
Liouville formulation: integrator for classical Hamiltonian

\[
\exp \left( \frac{\Delta t}{2} F(x) \frac{\partial}{\partial p} \right) \begin{pmatrix} x \\ p \end{pmatrix} = \begin{pmatrix} x \\ p + \frac{\Delta t}{2} F(x) \end{pmatrix}
\]

\[
\exp \left( \Delta t \frac{p}{m} \frac{\partial}{\partial x} \right) \begin{pmatrix} x \\ p + \frac{\Delta t}{2} F(x) \end{pmatrix} = \begin{pmatrix} x + \Delta t \frac{p}{m} \\ p + \frac{\Delta t}{2} F \left( x + \Delta t \frac{p}{m} \right) \end{pmatrix}
\]

\[
\exp \left( \Delta t F(x) \frac{\partial}{\partial p} \right) \begin{pmatrix} x + \Delta t \frac{p}{m} \\ p + \frac{\Delta t}{2} F \left( x + \Delta t \frac{p}{m} \right) \end{pmatrix} = \begin{pmatrix} x + \frac{\Delta t}{m} \left( p + \frac{\Delta t}{2} F(x) \right) \\ p + \frac{\Delta t}{2} F \left( x + \frac{\Delta t}{m} \left( p + \frac{\Delta t}{2} F(x) \right) \right) \end{pmatrix}
\]

\[
x(\Delta t) = x(0) + \Delta tv(0) + \frac{\Delta t^2}{2m} F(x(0))
\]

\[
v(\Delta t) = v(0) + \frac{\Delta t}{2m} [F(x(0)) + F(x(\Delta t))]
\]

\[
p(\Delta t/2) = p(0) + \frac{\Delta t}{2} F(x(0))
\]

\[
x(\Delta t) = x(0) + \frac{\Delta t}{m} p(\Delta t/2)
\]

\[
p(\Delta t) = p(\Delta t/2) + \frac{\Delta t}{2} F(x(\Delta t))
\]

Velocity Verlet!

Gives directly the implementation:

\[
p = p + 0.5 \times \Delta t \times F
\]

\[
x = x + \Delta t \times p/m
\]

Recalculate the force

\[
p = p + 0.5 \times \Delta t \times F.
\]
Liouville formulation: multiple time-scale integrator

Hamiltonian with “fast” and “slow” degrees of freedom

\[
\dot{x} = \frac{p}{m}, \\
\dot{p} = F_{\text{fast}}(x) + F_{\text{slow}}(x)
\]

\[
iL = \frac{p}{m} \frac{\partial}{\partial x} + [F_{\text{fast}}(x) + F_{\text{slow}}(x)] \frac{\partial}{\partial p}
\]

\[
iL = iL_1 + iL_2
\]

\[
iL_1 = \frac{p}{m} \frac{\partial}{\partial x}
\]

\[
iL_2 = [F_{\text{fast}}(x) + F_{\text{slow}}(x)] \frac{\partial}{\partial p}
\]

\[
iL = iL_{\text{fast}} + iL_{\text{slow}}
\]

\[
iL_{\text{fast}} = \frac{p}{m} \frac{\partial}{\partial x} + F_{\text{fast}}(x) \frac{\partial}{\partial p}
\]

\[
iL_{\text{slow}} = F_{\text{slow}}(x) \frac{\partial}{\partial p}
\]

\[
\exp(iL\Delta t) = \exp\left(iL_{\text{slow}} \frac{\Delta t}{2}\right) \exp(iL_{\text{fast}}\Delta t) \exp\left(iL_{\text{slow}} \frac{\Delta t}{2}\right)
\]
Liouville formulation: multiple time-scale integrator

Introduction of a sub-timestep \( \delta t = \Delta t / n \)

\[
\exp(iL_{\text{fast}} \Delta t) = \left[ \exp \left( \frac{\delta t}{2} \frac{F_{\text{fast}}}{m} \frac{\partial}{\partial x} \right) \exp \left( \frac{\delta t}{2} \frac{p}{m} \frac{\partial}{\partial x} \right) \exp \left( \frac{\delta t}{2} \frac{F_{\text{fast}}}{m} \frac{\partial}{\partial p} \right) \right]^n
\]

\[
\exp(iL \Delta t) = \exp \left( \frac{\Delta t}{2} \frac{F_{\text{slow}}}{m} \frac{\partial}{\partial p} \right) \left[ \exp \left( \frac{\delta t}{2} \frac{F_{\text{fast}}}{m} \frac{\partial}{\partial p} \right) \exp \left( \frac{\delta t}{2} \frac{p}{m} \frac{\partial}{\partial x} \right) \exp \left( \frac{\delta t}{2} \frac{F_{\text{fast}}}{m} \frac{\partial}{\partial p} \right) \right]^n
\]

\[
\times \exp \left( \frac{\Delta t}{2} \frac{F_{\text{slow}}}{m} \frac{\partial}{\partial p} \right)
\]

\[
p = p + 0.5 \times \Delta t \times F_{\text{slow}}
\]

for \( i = 1 \) to \( n \)

\[
p = p + 0.5 \times \delta t \times F_{\text{fast}}
\]

\[
x = x + \delta t \times p / m
\]

Recalculate fast force

\[
p = p + 0.5 \times \delta t \times F_{\text{fast}}
\]

endfor

Recalculate slow force

\[
p = p + 0.5 \times \Delta t \times F_{\text{slow}}.
\]
Outline

• Basic MD
  – Chaos
  – Shadow trajectories
  – Ergodicity

• Practical MD

• Ensembles
  – MD generates the NVE ensemble
  – The canonical NVT ensemble: thermostats

• Integrating the equations of motion
  – Verlet or velocity Verlet?
  – Liouville formulation
  – Multiple time steps

• Applications:
  – Vibrations
  – Computing transport properties
Calculating vibrations via MD

Time autocorrelation functions can give information about vibrations
– From Fermi's golden rule, the dipole time auto correlation function gives the intensities of IR active frequencies

\[ I(\omega) = \int_{-\infty}^{\infty} dt \, e^{i\omega t} \langle \vec{M}(t) \cdot \vec{M}(0) \rangle_t \]

– Velocity time autocorrelation gives all frequencies of vibration.

\[ \text{VDOS}(\omega) = \sum_{i=1}^{N} \int_{-\infty}^{\infty} dt \, e^{i\omega t} \langle \vec{v}_i(t) \cdot \vec{v}_i(0) \rangle_t \]

– Possible to assign to individual atoms displacements and project on eigenmodes.

Application: IR and power spectra
Application: IR and power spectra

LMG et al. NJP (2013)
Application: IR and power spectra

LMG et al. NJP (2013)
Vibrational spectroscopy: Ac-Ala$_{15}$-LysH$^+$

α-helical Ac-Ala$_{15}$-LysH$^+$

Experiment:
von Helden, Kupser, Bierau, Meijer, Molecular Physics, FHI Berlin

Infrared multiphoton dissociation spectroscopy, FELIX free electron laser

Room temperature

180 atoms

Experiment (IRMPD, 300K) x5

Intensity

1000  1200  1400  1600  1800

wave number [cm$^{-1}$]
Vibrational spectroscopy: Ac-Ala_{15}-LysH^+

α-helical Ac-Ala_{15}-LysH^+

Experiment:
von Helden, Kupser, Bierau, Meijer,
Molecular Physics, FHI Berlin

Infrared multiphoton dissociation spectroscopy, FELIX free electron laser

Room temperature

Intensity

1000 1200 1400 1600 1800
wave number [cm^{-1}]

Experiment (IRMPD, 300K)
Theory (harmonic) x5

180 atoms
Vibrational spectroscopy: Ac-Ala$\text{$_{15}$}$-LysH$^+$

\[ I(\omega) \propto \omega^2 \int_{-\infty}^{\infty} dt \langle \vec{M}(t) \cdot \vec{M}(0) \rangle e^{i\omega t} \]

dipole-dipole time correlation function

Rossi et al., JCP Lett. (2010)
Vibrational spectroscopy: water (Ice)

Vibrational density of states of proton (H) in ice

Bussi, Donadio, Parrinello JCP (2007)
Application: 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

\[ 2D = \lim_{t \to \infty} \frac{d \langle x^2(t) \rangle}{dt} \]
Transport coefficients: Diffusion

General $c(x,t)$

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

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

Diffusion in 3 dimensions

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

Relation to velocity:

\[ \Delta x(t) = \int_0^t dt' \, v_x(t') \]

\[ \langle x^2(t) \rangle = \left\langle \left( \int_0^t dt' \, v_x(t') \right)^2 \right\rangle \]

\[ = \int_0^t \int_0^t dt' \, dt'' \left\langle v_x(t') v_x(t'') \right\rangle \]

\[ = 2 \int_0^t \int_0^{t'} dt' \, dt'' \left\langle v_x(t') v_x(t'') \right\rangle \]
Transport coefficients: Diffusion

\[ \frac{\partial \langle x^2(t) \rangle}{\partial t} = 2 \frac{\partial}{\partial t} \int_0^t dt' \int_0^{t'} dt'' \langle v_x(t') v_x(t'') \rangle \]

\[ \langle v_x(t') v_x(t'') \rangle = \langle v_x(t' - t'') v_x(0) \rangle \]

\[ 2D = \lim_{t \to \infty} 2 \int_0^t dt'' \langle v_x(t - t'') v_x(0) \rangle \]

Define: \( \tau = t - t'' \)

\[ D = \int_0^\infty d\tau \langle v_x(\tau) v_x(0) \rangle \]

Green – Kubo relation
Also exists for other transport coefficients, such as viscosity and conductivity
In general the microcanonical phase space density is

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

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

with \( x = \{p^N, r^N\} \)

Integration 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 \( H = E = \text{constant} \) (and constant total \( P \)).

\[ \rho(x) = \delta[E - \mathcal{H}(x)] / g(E) \]

\[ g(E) = \int dx \delta[E - \mathcal{H}(x)] \]

with instantaneous temperature

\[ k_B T = \sum_{i=1}^{N} \frac{m v_i^2}{N_f} \]