Computer Simulation Methods

University of Cambridge Part II Natural Sciences Tripos

Yue Wu

Yusuf Hamied Department of Chemistry
Lensfield Road,
Cambridge, CB2 1EW

yw628@cam.ac.uk

Acknowledgements

Nothing in these lecture notes is original. They are largely based on the notes by Prof. Rosana Collepardo, who lectured this course in 2025. Moreover, they are nowhere near accurate representations of what was actually lectured, and in particular, all errors are almost surely mine.

Contents

1 Molecular Dynamics

1.1 Integrating the Equations of Motion

A famous conclusion in classical mechanics is that the motion of three bodies under gravitational interaction has no general analytic solution in closed form. The problem will only get worse if we introduce more bodies into the system or consider more complex forms of interaction. It is not uncommon for a system in physics and chemistry to have more than billions of interacting particles, so we usually have no choice but to treat their interactions numerically on a computer. The aim of *Molecular Dynamics* (MD) is to study a system by recreating it on the computer as close to nature as possible.

1.1.1 Newton's Equations of Motion

We start from the most fundamental law in classical mechanics — Newton's equation. If we have a system of N particles, and their positions are given by $\{\mathbf{r}_i\}_{i=1}^N$ which we collectively denote as $\mathbf{r}^N$, then the interactions between the particles will be completely determined by the positions of the particles, specified by the potential

$$V = V(\mathbf{r}_1, \ldots, \mathbf{r}_N) \equiv V(\mathbf{r}^N)\,. \tag{1.1}$$

The *force* $\mathbf{f}_i$ acting on particle i is the negative gradient of V, which is again a function of the configuration $\mathbf{r}^N$:

$$\mathbf{f}_i(\mathbf{r}^N) = -\boldsymbol{\nabla}_i V(\mathbf{r}^N) = -\frac{\partial V(\mathbf{r}^N)}{\partial \mathbf{r}_i}\,. \tag{1.2}$$

Then the future evolution of the system is given by the Newton's second law

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i(\mathbf{r}^N)\,, \tag{1.3}$$

where m_i is the mass of particle i. This is a system of N coupled second-order differential equations. Additionally, it is often useful to introduce the *momentum* of a particle. In Cartesian coordinates, the momentum of a particle is given by

$$\mathbf{p}_i = m_i \dot{\mathbf{r}}_i\,. \tag{1.4}$$

When modelling a system composed of atoms and molecules, interatomic interactions, as opposed to the forces related to chemical bonds keeping molecules together, are relatively weak. A good and extremely common approximation for intermolecular interactions is that they are *pairwise additive*. Moreover, for atoms, these pair potentials can be assumed to be central so that they depend only on the interatomic distances. Then the total potential V can be resolved into a sum of pair potentials v_{ij} between all pairs of atoms i and j, where v_{ij} is only a function of the interatomic distance $r_{ij} \equiv \|\mathbf{r}_{ij}\| := \|\mathbf{r}_i - \mathbf{r}_j\|$

$$V(\mathbf{r}^N) = \sum_{i=1}^{N} \sum_{j>i}^{N} v_{ij}(r_{ij})\,. \tag{1.5}$$

The sum is taken over $j > i$ so that each pair of atoms i and j is summed exactly once. This restriction can be lifted by noting v_{ij} and v_{ji} both describes the potential between particle i and j so they should be the same, and $r_{ij} = r_{ji}$, so

$$V(\mathbf{r}^N) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j\neq i}^{N} v_{ij}(r_{ij})\,. \tag{1.6}$$

The factor of a half cancels with the problem of double counting, and self interactions with $i = j$ are excluded.

If we identify the force acting on particle i by particle j is

$$\mathbf{f}_{ij} \coloneqq -\frac{\partial v_{ij}}{\partial \mathbf{r}_i}\,, \tag{1.7}$$

then it is easy to show that the total force acting on particle i is the sum of the forces acted by all other particles using the form of potential given by (1.6)

$$\begin{aligned}\mathbf{f}_i &= -\frac{\partial V}{\partial \mathbf{r}_i}\\ &= -\frac{1}{2}\sum_{j\neq i}^{N}\left(\frac{\partial v_{ij}(r_{ij})}{\partial \mathbf{r}_i} + \frac{\partial v_{ji}(r_{ji})}{\partial \mathbf{r}_i}\right)\\ &= -\sum_{j\neq i}^{N}\frac{\partial v_{ij}(r_{ij})}{\partial \mathbf{r}_i} = \sum_{j\neq i}^{N}\mathbf{f}_{ij}\,. \end{aligned} \tag{1.8}$$

It is also not difficult to show that the pair forces satisfy the Newton's third law

$$\mathbf{f}_{ij} = -\mathbf{f}_{ji}\,. \tag{1.9}$$

1.1.2 Properties of Classical Dynamics

Energy Conservation

A fundamental property of mechanical systems, for pair or many-body interactions, provided that they can be derived from a potential invariant in time, is that the total energy is conserved during the motion. The total energy is the sum of the potential energy V and the kinetic energy K defined by

$$K = \sum_{i=1}^{N}\frac{1}{2}m_i\dot{\mathbf{r}}_i^2\,. \tag{1.10}$$

We define the *Hamiltonian* of a system to be

$$H(\mathbf{r}^N, \dot{\mathbf{r}}^N, t) \coloneqq K + V = \sum_{i=1}^{N}\frac{1}{2}m_i\dot{r}_i^2 + V(\mathbf{r}^N, t)\,, \tag{1.11}$$

For our purposes, it is just another way of saying the total energy of the system.[1] In our cases, the potential V and hence the whole Hamiltonian has no explicit time dependence, meaning t does not appear explicitly in the expression of both K and V, although both $\mathbf{r}$ and $\dot{\mathbf{r}}$ evolve in time. If this is the case, then the Hamiltonian (total energy) remains constant over time. This is the conservation of energy.

Theorem 1.1 (Energy Conservation). If $H(\mathbf{r}_i, \dot{\mathbf{r}}_i, t)$ has no explicit time dependence,

$$\frac{\partial H}{\partial t} = 0\,, \tag{1.13}$$

then H is a constant of motion,

$$\frac{\mathrm{d}H}{\mathrm{d}t} = 0\,. \tag{1.14}$$

[1] Formally in classical mechanics, the Hamiltonian is defined as the Legendre transform of the Lagrangian, by

$$H = \sum_{i=1}^{n} p_i\dot{q}_i - L(q_i, \dot{q}_i, t)\,. \tag{1.12}$$

p_i is the *conjugated momentum* of the *generalised coordinate* q_i. Since the generalised coordinate is not necessarily the Cartesian coordinates (e.g. spherical, or even non-orthogonal), H may not be the energy.

Proof. By chain rule,

$$\begin{aligned}\frac{\mathrm{d}H}{\mathrm{d}t} &= \frac{\partial H}{\partial t} + \sum_{i=1}^{N} \frac{\partial H}{\partial \mathbf{r}_i}\frac{\partial \mathbf{r}_i}{\partial t} + \sum_{i=1}^{N} \frac{\partial H}{\partial \dot{\mathbf{r}}_i}\frac{\partial \dot{\mathbf{r}}_i}{\partial t} \\ &= 0 + \sum_{i=1}^{N} \frac{\partial V}{\partial \mathbf{r}_i} \cdot \dot{\mathbf{r}}_i + \sum_{i=1}^{N} m_i \dot{\mathbf{r}}_i \cdot \ddot{\mathbf{r}}_i \\ &= \sum_{i=1}^{N} \dot{\mathbf{r}}_i \cdot (m_i \ddot{\mathbf{r}}_i - \mathbf{f}_i) = 0 \end{aligned} \tag{1.15}$$

from Newton's second law. □

Since the Hamiltonian is not explicitly dependent on time, the system has time-translational symmetry, meaning that we start the system at time t and at time $t + \delta t$, the system will evolve in the same way. This result of a continuous symmetry of the system leading to a conserved quantity is an example of Noether's theorem, which is arguably the most beautiful and profound result in physics. We will briefly introduce this result in section A.

The conservation of energy is fundamental in the derivation of the equilibrium ensembles in statistical mechanics. It can also be applied as a very powerful test of the stability of a numerical scheme for the integration of the equations of motion. We will repeatedly return to this point.

Time Reversal Symmetry

Another feature of Newtonian dynamics that plays a role both in the theory of statistical mechanics and in the practice of the development of molecular dynamics algorithms is *time reversal symmetry.* This states that if we reverse all velocities at time t while keeping the positions the same, then the system will retrace its trajectory back into the past. We introduce the notation

$$\mathbf{r}^N(t \mid \mathbf{r}_0^N, \mathbf{p}_0^N) \tag{1.16}$$

to be $\mathbf{r}^N$ under the initial condition $\mathbf{r}^N(0) = \mathbf{r}_0^N$ and $\mathbf{p}^N(0) = \mathbf{p}_0^N$. Then time reversal symmetry implies the relations

$$\mathbf{r}^N(t \mid \mathbf{r}^N(0), -\mathbf{p}^N(0)) = \mathbf{r}^N(-t \mid \mathbf{r}^N(0), \mathbf{p}^N(0))\,, \tag{1.17}$$

$$\mathbf{p}^N(t \mid \mathbf{r}^N(0), -\mathbf{p}^N(0)) = -\mathbf{p}^N(-t \mid \mathbf{r}^N(0), \mathbf{p}^N(0))\,. \tag{1.18}$$

1.1.3 Euler's Algorithm

Molecular dynamics methods are iterative numerical schemes for solving the equations of motion of the system. The first step is to discretise the time into small intervals, and we assume each interval has equal length δt. Then the evolution of the system is then described by the series of coordinates and velocities

$$\mathbf{r}^N(t_0) \equiv \mathbf{r}^N(0), \ldots, \mathbf{r}^N(t_{m-1}) \equiv \mathbf{r}^N(t_m - \delta t), \mathbf{r}^N(t_m), \mathbf{r}^N(t_{m+1}) \equiv \mathbf{r}^N(t_m + \delta t), \ldots \tag{1.19}$$

$$\dot{\mathbf{r}}^N(t_0) \equiv \dot{\mathbf{r}}^N(0), \ldots, \dot{\mathbf{r}}^N(t_{m-1}) \equiv \dot{\mathbf{r}}^N(t_m - \delta t), \dot{\mathbf{r}}^N(t_m), \dot{\mathbf{r}}^N(t_{m+1}) \equiv \dot{\mathbf{r}}^N(t_m + \delta t), \ldots \tag{1.20}$$

The schemes we will discuss all use the Cartesian coordinates.

The most fundamental integrator in molecular dynamics is *Euler's algorithm.* It approximates the position of the molecule at time $t + \delta t$ by a Taylor series about t truncated at $O(\delta t^2)$

$$\begin{aligned} \mathbf{r}_i(t+\delta t) &= \mathbf{r}_i(t) + \dot{\mathbf{r}}(t)\delta t + \frac{1}{2}\ddot{\mathbf{r}}(t)\delta t^2 + O(\delta t^3) \\ &= \mathbf{r}_i(t) + \delta t \mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) + O(\delta t^3)\,. \end{aligned} \tag{1.21}$$

Similarly one can obtain an expansion for $\mathbf{v}_i(t + \delta t)$

$$\mathbf{v}_i(t+\delta t) = \mathbf{v}_i(t) + \frac{\delta t}{m_i}\mathbf{f}_i(t) + O(\delta t^2)\,. \tag{1.22}$$

Having an expression for $\mathbf{f}_i$, which can be derived from the potential V, we can perform this process iteratively.

Algorithm 1.2 (Euler's Algorithm).

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \delta t \mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) \tag{1.23}$$

$$\mathbf{v}_i(t+\delta t) = \mathbf{v}_i(t) + \frac{\delta t}{m_i}\mathbf{f}_i(t)\,. \tag{1.24}$$

The error of displacement is $O(\delta t^3)$ and that of velocity is $O(\delta t^2)$.

Euler's algorithm is simple, but it has a huge drawback — it is simply not accurate enough.

To illustrate this, we calculated the equation of motion of a standard Harmonic oscillator, with Hamiltonian

$$H = \frac{1}{2}v^2 + \frac{1}{2}x^2 \tag{1.25}$$

and initial conditions $x_0 = 0$ and $v_0 = 1$. The results are plotted in figure 1. The exact solution is a sine wave

$$x(t) = \sin t\,. \tag{1.26}$$

We can see that Euler's solution quickly diverges from the exact solution as amplitude of oscillation quickly grows larger and larger. Euler's algorithm has several problems, making it very limited in current practical use:

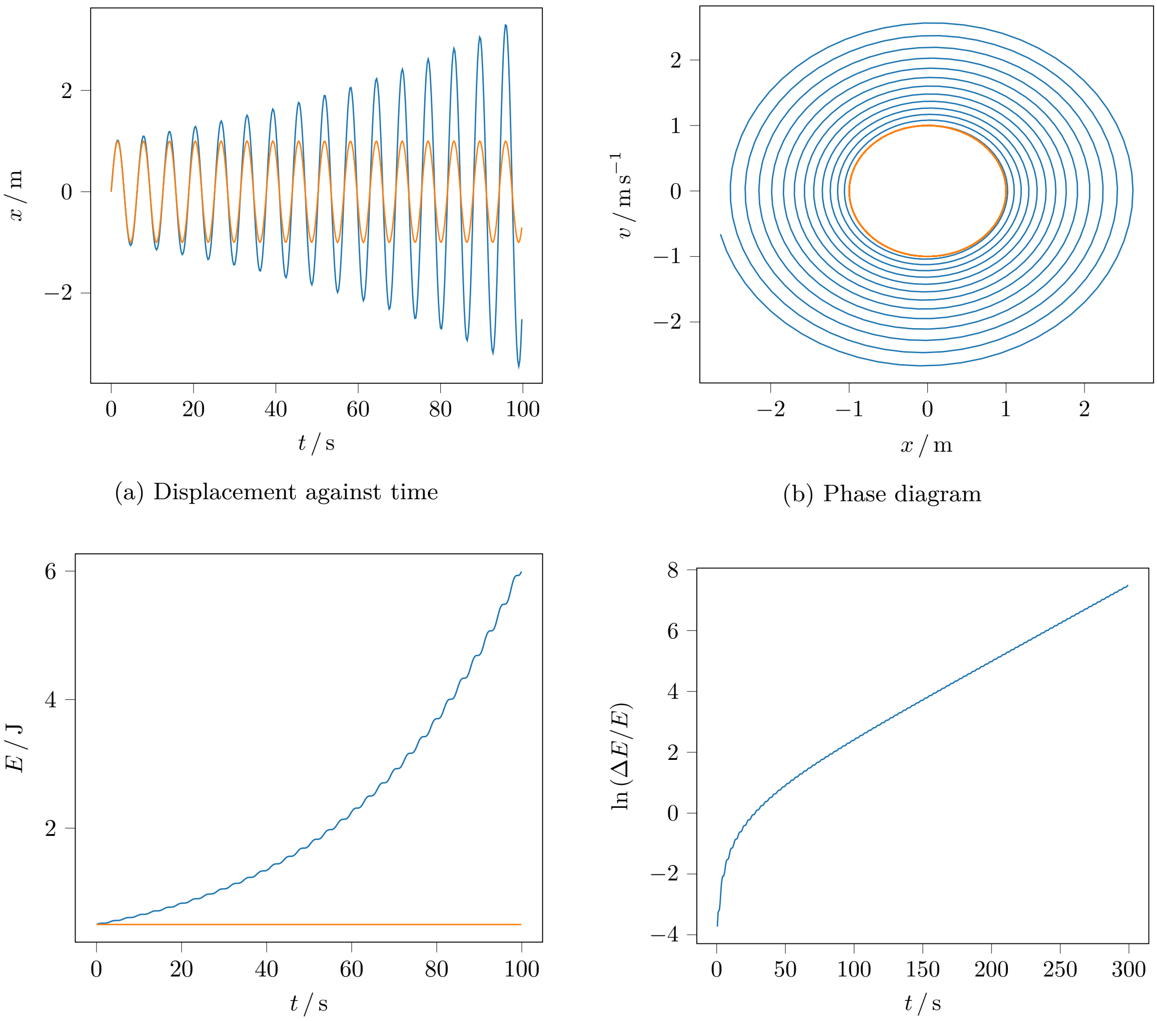
- The solution is not time-reversible.
- *Liouville's theorem* states that the volume of a set in the phase space is conserved as it evolves in time. In Euler's algorithm, the volume in the phase space is not conserved.
- The system is susceptible to energy drift. As we can see in the plot, the energy drifts exponentially fast.

This suggests that we need a more accurate algorithm.

1.1.4 Verlet Algorithm

An obvious thing to do to improve Euler's algorithm is to include more terms in the Taylor expansions. If we truncate the expansion of $\mathbf{x}(t + \delta t)$ at $O(\delta t^3)$, we get

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \delta t \mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) + \frac{\delta t^3}{6}\mathbf{b}_i(t) + O(\delta t^4)\,, \tag{1.27}$$



(a) Displacement against time

(b) Phase diagram

(c) Energy against time

(d) Logarithmic error of energy

Figure 1: The motion of a harmonic oscillator solved by Euler's method (blue) and the exact solution (orange). The step length in Euler's method is set to $\delta t = 0.05$ s.

where $\mathbf{b}_i \coloneqq \dddot{\mathbf{r}}_i$ is the third derivative of position. But this leads to a problem — it is not easy to evaluate this third derivative. But the trick is we don't have to evaluate it. If we expand the position backward in time, we get

$$\mathbf{r}_i(t-\delta t) = \mathbf{r}_i(t) - \delta t\mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) - \frac{\delta t^3}{6}\mathbf{b}_i(t) + O(\delta t^4)\,. \tag{1.28}$$

Now if we add these two expressions together, the third derivatives nicely cancel out and we are left with

$$\mathbf{r}_i(t+\delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\delta t) + \frac{\delta t^2}{m_i}\mathbf{f}_i(t) + O(\delta t^4)\,. \tag{1.29}$$

Now this expression has error $O(\delta t^4)$, and we don't even need to evaluate the velocities $\mathbf{v}(t)$ if we are only interested in the particles' coordinates.

However, in most cases, the velocities still tell us valuable information. To calculate it, we can subtract the two expansions forward and backward in time and get

$$\mathbf{v}_i(t) = \frac{1}{2\delta t}[\mathbf{r}_i(t+\delta t) - \mathbf{r}_i(t-\delta t)] + O(\delta t^3)\,. \tag{1.30}$$

Algorithm 1.3 (Verlet Algorithm).

$$\mathbf{r}_i(t+\delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\delta t) + \frac{\delta t^2}{m_i}\mathbf{f}_i(t) \tag{1.31}$$

$$\mathbf{v}_i(t) = \frac{1}{2\delta t}[\mathbf{r}_i(t+\delta t) - \mathbf{r}_i(t-\delta t)]\,. \tag{1.32}$$

The error of displacement is $O(\delta t^4)$ and that of velocity is $O(\delta t^3)$.

However, to calculate $\mathbf{v}(t)$, you need the knowledge of position at $t+\delta t$, i.e. the velocity update in Verlet algorithm is one step behind the position update. This is not a problem for propagating position because, assuming that the forces are not dependent on velocity, information on $\mathbf{v}_i(t)$ is not needed in (equation (1.31)). Still, this may be inconvenient for the determination of velocity-dependent quantities or for the algorithms which manipulate the velocity during dynamics. The position and velocity update can be brought in the same step by a reformulation of the Verlet scheme, called *velocity Verlet*. The prediction for position is now simply obtained from the Taylor expansion, again keeping to the second order

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \delta t\mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) + O(\delta t^3)\,. \tag{1.33}$$

For the advanced position obtained this way, we compute the force at time $t+\delta t$

$$\mathbf{f}_i(t+\delta t) = \mathbf{f}_i(\{\mathbf{r}_j(t+\delta t)\}) = \mathbf{f}_i\left(\left\{\mathbf{r}_i(t) + \delta t\mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t)\right\}\right)\,, \tag{1.34}$$

in which all particles have proceeded to their positions at $t+\delta t$. Substituting this expression into the Taylor expansion of $\mathbf{r}_i(t)$ about $t+\delta t$ backward in time, we obtain

$$\mathbf{r}_i(\mathbf{t}) = \mathbf{r}_i(t+\delta t) - \delta t\mathbf{v}_i(t+\delta t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t+\delta t) + O(\delta t^3)\,. \tag{1.35}$$

Adding this to the forward expansion (1.33) gives the prediction of velocity

$$\mathbf{v}_i(t+\delta t) = \mathbf{v}_i(t) + \frac{\delta t}{2m_i}[\mathbf{f}_i(t) + \mathbf{f}_i(t+\delta t)] + O(\delta t^3)\,. \tag{1.36}$$

Algorithm 1.4 (Velocity Verlet Algorithm).

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \delta t\mathbf{v}_i(t) + \frac{\delta t^2}{2m_i}\mathbf{f}_i(t) \tag{1.37}$$

$$\mathbf{v}_i(t+\delta t) = \mathbf{v}_i(t) + \frac{\delta t}{2m_i}[\mathbf{f}_i(t) + \mathbf{f}_i(t+\delta t)]\,. \tag{1.38}$$

The velocity Verlet looks rather different than the Verlet algorithm, especially the $O(\delta t^3)$ error terms when we derive it look concerning. However, we can show that these two algorithms are equivalent. We can show this by rewriting the velocity Verlet prediction of the position.

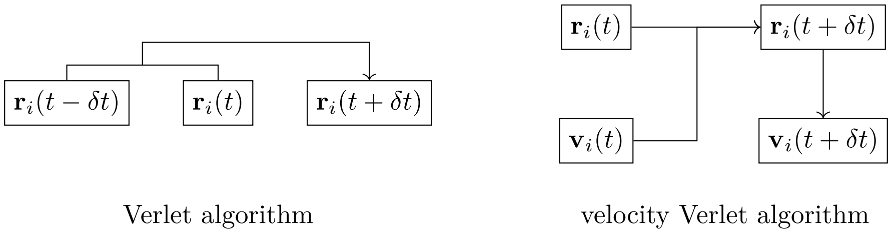
Proposition 1.5. The velocity Verlet algorithm is equivalent to the Verlet algorithm.

Proof. If we subtract the $t-\delta t \to t$ prediction for the position from the $t \to t+\delta t$ prediction in the velocity Verlet algorithm, we find

$$\mathbf{r}_i(t+\delta t) - \mathbf{r}_i(t) = \mathbf{r}_i(t) - \mathbf{r}_i(t-\delta t) + \delta t[\mathbf{v}_i(t) - \mathbf{v}_i(t-\delta t)] + \frac{\delta t^2}{2m_i}[\mathbf{f}_i(t) - \mathbf{f}_i(t-\delta t)]\,. \tag{1.39}$$

The $t-\delta t \to t$ update for the velocity is

$$\mathbf{v}_i(t) = \mathbf{v}_i(t-\delta t) + \frac{\delta t}{2m_i}[\mathbf{f}_i(t-\delta t) + \mathbf{f}_i(t)]\,. \tag{1.40}$$


$\mathbf{r}_i(t-\delta t)$
$\mathbf{r}_i(t)$
$\mathbf{r}_i(t+\delta t)$
$\mathbf{r}_i(t)$
$\mathbf{r}_i(t+\delta t)$
$\mathbf{v}_i(t)$
$\mathbf{v}_i(t+\delta t)$

Verlet algorithm

velocity Verlet algorithm

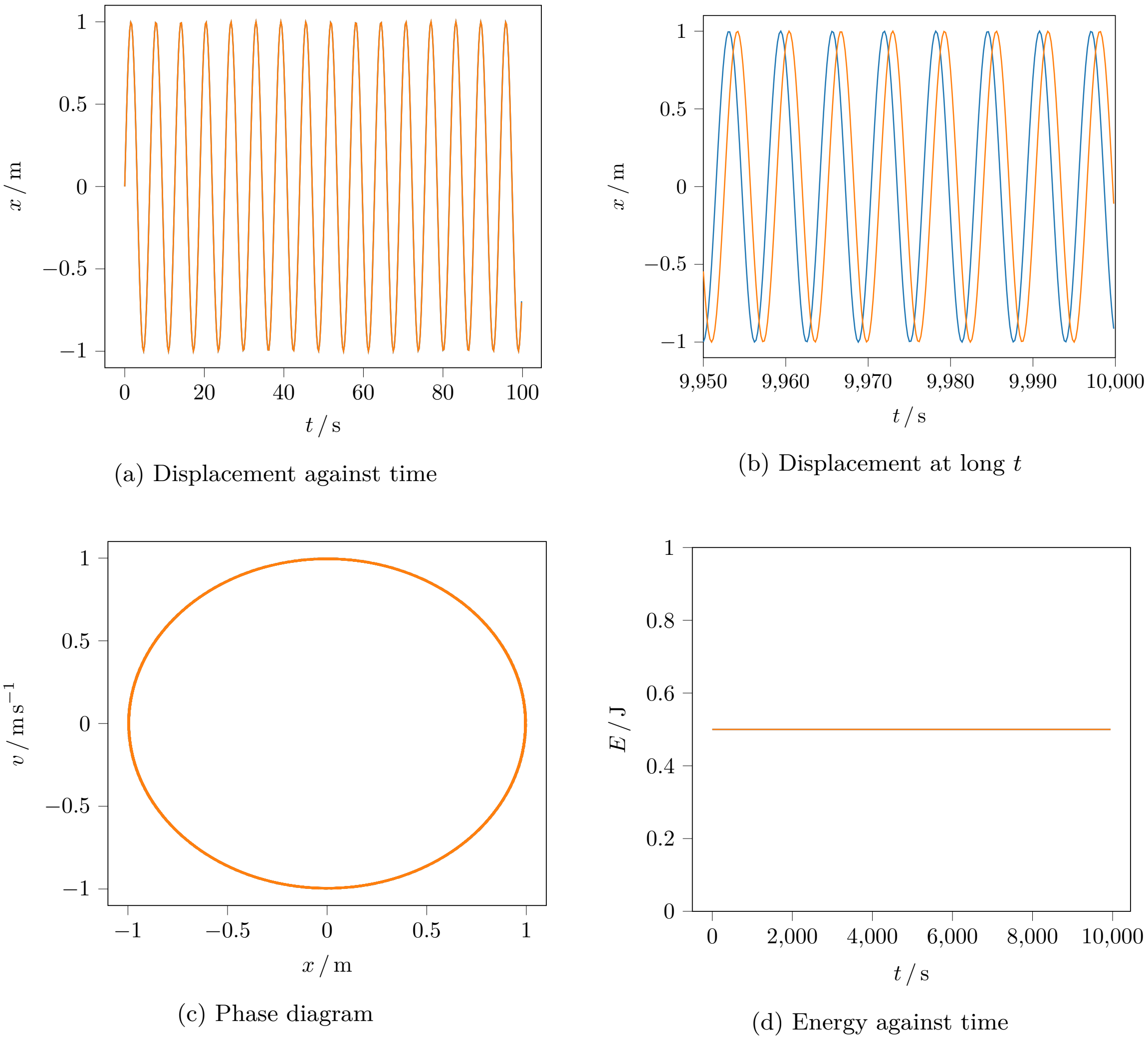
Figure 2: Schemes of Verlet and velocity Verlet algorithms.

If we substitute this into (1.39), we get

$$\mathbf{r}_i(t+\delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\delta t) + \frac{\delta t^2}{m_i}\mathbf{f}_i(t)\,. \tag{1.41}$$

The velocity Verlet algorithm gives the same prediction as the Verlet algorithm. □

Now let's examine the accuracy of Verlet's algorithm using harmonic oscillator.



(a) Displacement against time

(b) Displacement at long t

(c) Phase diagram

(d) Energy against time

Figure 3: The motion of a harmonic oscillator solved by Verlet algorithm (blue) and the exact solution (orange). The step length in Verlet's method is also $\delta t = 0.05$ s.

We can see that the Verlet algorithm is very accurate. Although the displacements deviate from the exact displacement over time, the phase-space volume is conserved exactly and the total energy remains nearly constant (with only small bounded oscillations) over very long times.

1.1.5 Why use the Verlet Algorithm? (Non-examinable)

While there are algorithms with better short time accuracy than the Verlet algorithm, the overwhelming majority of condensed matter molecular dynamics simulations is based on just the Verlet algorithm. There are a number of reasons for its popularity.

(i) The Verlet algorithm is simple and only depends on forces. No higher derivatives of the energy are needed. This is important because the force evaluation is the most CPU time consuming in MD simulations of interacting many-particle systems. Computation of higher derivatives of energy will increase computational costs substantially. Although the algorithms using force derivatives are more accurate, this gain is actually relatively minor. Because of the chaotic nature of the motion in many-particle systems, the particles rapidly deviate from the "true" trajectories. This is known as *Lyapunov instability*: trajectories that differ slightly in initial conditions will diverge exponentially in time. If we denote $\mathbf{r}(t) = \mathbf{r}(t \mid \mathbf{r}_0, \mathbf{p}_0)$ and $\mathbf{r}'(t) = \mathbf{r}(t \mid \mathbf{r}_0, \mathbf{p}_0 + \epsilon)$, then

$$|\mathbf{r}(t) - \mathbf{r}'(t)| \sim \epsilon \exp(\lambda t)\,, \tag{1.42}$$

where λ is the *Lyapunov exponent.* This can be seen if we plot the logarithmic error of displacement against time, where the upper bound of the plot is a straight line with gradient λ.

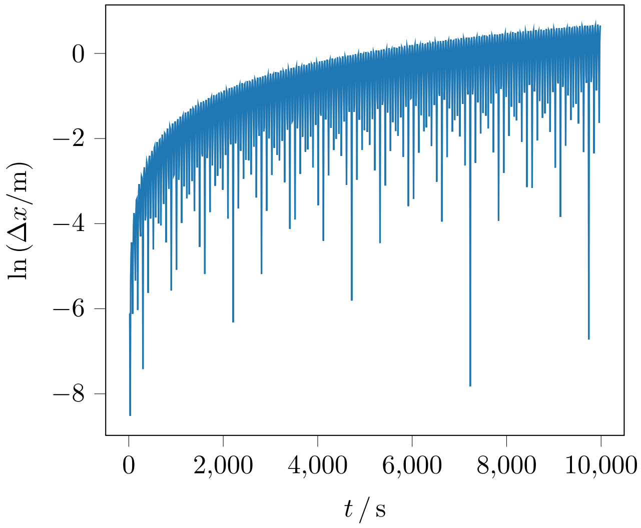


Figure 4: Logarithmic error of displacement of a harmonic oscillator using Verlet algorithm.

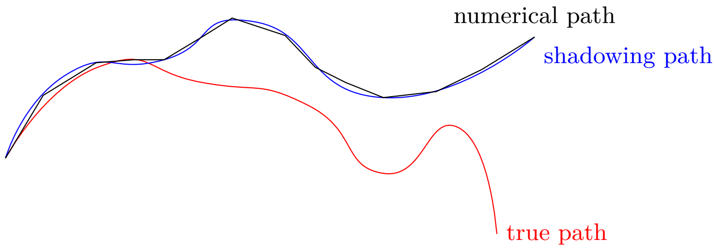
Having an exponentially growing error may seems horrible, but it is actually not as big a problem as it seems!

Theorem 1.6 (Shadowing theorem). Every numerical trajectory will be uniformly close to some true trajectory with slightly altered initial position. In other words, a numerical trajectory is "shadowed" by a true one.

This strong molecular chaos is ultimately the justification of methods of statistical mechanics.

(ii) Even though it only uses forces, the Verlet algorithm is correct up to and including $O(\delta t^3)$.

(iii) The Verlet algorithm is explicitly time reversible and, even though the trajectory relatively quickly diverges substantially from the true trajectory, the energy is conserved over an extremely



long period of time. Moreover, the Verlet algorithm rigorously conserves the normalisation of an ensemble probability distribution of points in phase space. In more advanced language, the Verlet integrator is said to be *symplectic*. These formal properties contribute to the superior long time stability of the Verlet algorithm.

For example, as we will discuss later, energy is the defining quantity for the microcanonical ensemble, since, for chaotic systems, there are no constraints on the regions trajectories can reach in phase space other than that they are confined to the hypersurface of constant energy. Energy conservation, together with norm conservation are therefore necessary conditions for thermodynamic stability, and ultimately for a proper definition of temperature. Long time stability is particularly important for the simulation of liquids which are stabilised by finite temperature dynamical fluctuations.

1.2 Connection to Equilibrium Statistical Mechanics

1.2.1 The Microcanonical Ensemble

So far, we have studied systems using Newtonian dynamics, in which the energy is naturally conserved if the system is closed. This corresponds to a microcanonical ensemble. For each observable A of the system, there is a corresponding *phase function* $\mathcal{A}(\mathbf{r}^N, \mathbf{p}^N)$ telling us the value of A given a state $(\mathbf{r}^N, \mathbf{p}^N)$ of the system. Then the ensemble average of the system is given by

$$\langle A \rangle_{NVE} = \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N)\mathcal{A}(\mathbf{r}^N, \mathbf{p}^N)\,, \tag{1.43}$$

where $\rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N)$ is the *microcanonical phase-space distribution function* restricting the manifold of accessible phase points $(\mathbf{r}^N, \mathbf{p}^N)$ to a hypersurface of constant energy E only, given by

$$\rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N) = \frac{f(N)}{\Omega_N}\delta(\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) - E)\,. \tag{1.44}$$

The phase function $\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)$ is the Hamiltonian, $f(N)$ is some function of the number of particles accounting for their indistinguishability, and Ω_N is the microcanonical partition function given by

$$\Omega_N = f(N) \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \delta(\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) - E)\,. \tag{1.45}$$

The factors $f(N)$ in the above expression can be omitted if we are only interested in mechanical observable averages over the ensemble distributions using ρ_{NVE}, but it becomes crucial if we want to give the normalisation factor Ω_N a thermodynamical interpretation when calculating entropy, free energy *etc.*

1.2.2 The Ergodic Principle and Time Averages

The above thermodynamic average of a quantity require us to evaluate it over the whole hypersurface of constant energy $\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)$ in the phase space — it is difficult to do this in a computer simulation.

However, we know that the state of a deterministic NVE system also evolves on this hypersurface of constant energy with time, with trajectory $(\mathbf{r}^N(t), \mathbf{p}^N(t))$. This allows us to determine how the physical observable A evolve along a certain trajectory as a function of time

$$A(t) \equiv \mathcal{A}(\mathbf{r}^N(t), \mathbf{p}^N(t))\,. \tag{1.46}$$

Now we are going to invoke some hypothesis.

Hypothesis 1.7 (Ergodic Hypothesis). Over a long enough period of time, the time spent by a system in some region of the phase space of microstates with the same energy is proportional to the volume of this region, i.e., that all accessible microstates are equiprobable over a long period of time.

This means that if we let our system evolves for a long period of time, then it will go over the whole subset of the phase space that it is allowed to go to — it is not saying that it will travel to every single point in the hypersurface of constant energy, which is impossible for a finite amount of time. What we are saying is that the system is sampling through a large enough portion of the phase space, so that the time average of the quantity $A(t)$ is essentially the ensemble average $\langle A \rangle_{NVE}$. We are replacing the ensemble average with a time average from a very long trajectory.

$$\langle A \rangle_{NVE} = \lim_{\tau\to\infty} \overline{A}_\tau = \lim_{\tau\to\infty} \frac{1}{\tau}\int_0^\tau \mathrm{d}t\, \mathcal{A}(\mathbf{r}^N(t), \mathbf{p}^N(t))\,. \tag{1.47}$$

In molecular dynamics computer simulation, we can only approximate a discrete path of time interval δt over a finite amount of time $\tau = M\delta t$. We then need to replace the above integral by a sum.

$$\langle A \rangle_\tau \approx \frac{1}{M}\sum_{m=1}^{M} \mathcal{A}(\mathbf{r}^N(t_m), \mathbf{p}^N(t_m))\,. \tag{1.48}$$

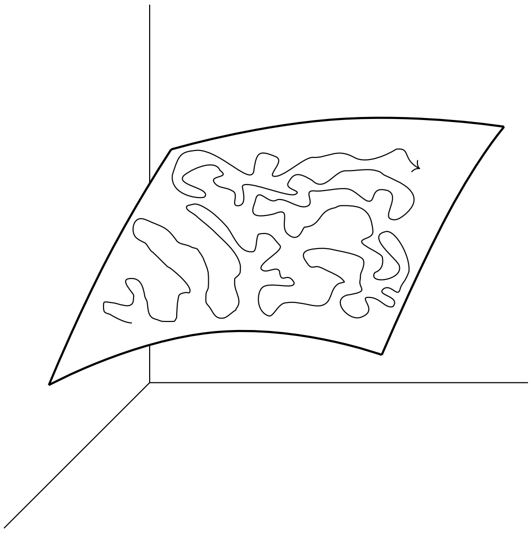


Figure 5: By the ergodic hypothesis, a trajectory will sample enough points on the whole hypersurface of constant energy such that the time average is a very good approximation to the ensemble average.

1.2.3 The Canonical Ensemble

Real world systems are hardly ever isolated. The least they do is to exchange energy with the environment. The states of such a system in equilibrium with a thermal reservoir of temperature T are distributed according to the canonical ensemble

$$\rho_{NVT}(\mathbf{r}^N, \mathbf{p}^N) = \frac{f(N)}{Q_N}\exp\left(-\frac{\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)}{k_B T}\right)\,, \tag{1.49}$$

where Q is the canonical partition function

$$Q_N = f(N) \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\, \mathbf{p}^N \exp\left(-\frac{\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)}{k_B T}\right) . \tag{1.50}$$

Canonical expectation values of observables are exponentially weighted averages over all points in phase space

$$\begin{aligned}\langle A \rangle_{NVT} &= \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \rho_{NVT}(\mathbf{r}^N, \mathbf{p}^N)\mathcal{A}(\mathbf{r}^N, \mathbf{p}^N) \\ &= \frac{f(N)}{Q_N} \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \mathcal{A}(\mathbf{r}^N, \mathbf{p}^N) \exp(-\beta\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)) . \end{aligned} \tag{1.51}$$

By taking the classical limit of the quantum canonical ensemble, the expression for the factor $f(N)$ can evaluated. If all N particles are identical, then it is

$$f(N) = \frac{1}{h^{3N} N!} . \tag{1.52}$$

The canonical partition function Q_N and microcanonical partition function Ω_N have all taken this into account. Their interpretation is suggested by considering the dimension of h, which is equal to that of position times momentum. $f(N)$ is therefore a very small reciprocal space volume which makes the canonical/microcanonical partition function dimensionless. Planck's constant therefore acts as a measure of the phase space metric and Q_N is interpreted as the effective number of accessible states at temperature T. The $N!$ factor takes account of the indistinguishability of the particles. It can be viewed as correcting for over-counting in the classical ensemble where permuting the position and momentum of a pair of particles would lead to a different but equivalent state (point) $(\mathbf{r}^N, \mathbf{p}^N)$ in the phase space. Similarly Ω_N is also the number of accessible states, except that in microcanonical ensemble it is restricted to the hypersurface of constant energy in the phase space (a manifold of dimension $6N - 1$). A mathematically more correct way of thinking the microcanonical partition function is that for a given infinitesimal change in energy $\mathrm{d}E$, the quantity $\Omega_N \mathrm{d}E$ gives the effective number of states contained in the volume between hypersurfaces with energy E and $E + \mathrm{d}E$.

Ω_N and Q_N are related to two very important thermodynamic quantities, namely the Boltzmann entropy

$$S = k_B \ln \Omega_N \tag{1.53}$$

and the Helmholtz free energy

$$A = -k_B T \ln Q_N . \tag{1.54}$$

They are the central relations linking statistical mechanics to thermodynamics. The factor $f(N)$ plays a crucial role in this identification. The founding fathers of statistical mechanics arrived at these results without the help of quantum mechanics — arguments concerning the additivity of entropy of mixing and similar considerations led them to postulate the form of the N dependence.

1.2.4 The Configuration Integral

It turns out that the kinetic energy is a rather trivial quantity in classical statistical thermodynamics.

Theorem 1.8 (Equipartition theorem). The average kinetic energy per particle is

$$\langle K \rangle / N = \frac{d}{2} k_B T , \tag{1.55}$$

where d is the dimension of the system.

This result is independent of the interaction potential or the mass. The origin of this is because the kinetic energy always takes the same form

$$\mathcal{K}(\mathbf{p}^N) = \sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m} . \tag{1.56}$$

When we evaluate the partition function, we can separate the Hamiltonian into a kinetic and a potential part, and the integral over the kinetic part will always be the same.

$$\begin{aligned}
Q_N &= f(N) \int \mathrm{d}^{3N}\mathbf{p}^N\, \mathrm{d}^{3N}\mathbf{r}^N\, \exp(-\beta\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)) \\
&= f(N) \int \mathrm{d}^{3N}\mathbf{p}^N\, \exp(-\beta\mathcal{K}(\mathbf{p}^N)) \int \mathrm{d}^{3N}\mathbf{r}^N\, \exp(-\beta\mathcal{V}(\mathbf{r}^N)) \\
&= f(N) \int \prod_{i=1}^{N} \mathrm{d}^3\mathbf{p}_i\, \exp\left(-\frac{\beta\mathbf{p}_i^2}{2m}\right) \int \mathrm{d}^{3N}\mathbf{r}^N\, \exp(-\beta\mathcal{V}(\mathbf{r}^N)) \\
&=: \frac{1}{N!\Lambda^{3N}} Z_N \,, \qquad (1.57)
\end{aligned}$$

where $\Lambda = h/\sqrt{2\pi m k_B T}$ is the thermal wavelength and we have defined the *configuration integral* to be

$$Z_N = \int \mathrm{d}^{3N}\mathbf{r}^N\, \exp(-\beta\mathcal{V}(\mathbf{r}^N)) \,. \qquad (1.58)$$

This is the more interesting quantity. For example, if we want to know the probability distribution $P_N(\mathbf{r}^N)$ for the configuration of the system, then we need to evaluate

$$P_N(\mathbf{r}^N) = \frac{\exp(-\beta\mathcal{V}(\mathbf{r}^N))}{Z_N} \,. \qquad (1.59)$$

The factor Λ^{3N} in the partition function, absorbing the h^{3N}, can be seen as a temperature dependent version of the volume element in the configuration space. The deeper significance of Λ is that it provides a criterion for the approach from the quantum to the classical limit. Quantum effects can be ignored in equilibrium statistics if Λ is smaller than any characteristic length in the system.

1.3 Temperature in Molecular Dynamics

Temperature was introduced in (1.49) as a parameter in the canonical ensemble, and via the fundamental equation (1.54), this statistical mechanical temperature is identified with the empirical temperature in classical thermodynamics. It is not immediately obvious, however, how to define and measure temperature in an MD simulation. To do this, we have to return to the microcanonical ensemble and find an observable (and correspondingly a phase function) whose microcanonical expectation value is a simple function of temperature, preferably linear. This would then allow us to measure the temperature of the ensemble by tracking the time average of the phase function over a sufficiently long period by the ergodic hypothesis. Such phase function is the kinetic energy, whose canonical average is

$$K = \left\langle \sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m_i} \right\rangle_{NVT} = \frac{3}{2} N k_B T \qquad (1.60)$$

in three dimensional system, as given by the equipartition theorem. The microcanonical average $\langle - \rangle_{NVE}$ (1.43) and the canonical average $\langle - \rangle_{NVT}$ (1.51) are not identical in general. But in Part II Statistical Mechanics we have shown that such fractional difference is vanishing as $N \to \infty$ — all ensembles are equivalent in the thermodynamic limit. Therefore the microcanonical average of the kinetic energy of a many particle system will also approach $\frac{3}{2} N k_B T$. Hence we can define an instantaneous or kinetic temperature function $\mathcal{T}$ in terms of the instantaneous kinetic energy $\mathcal{K}$ via[2]

$$\mathcal{T} = \frac{1}{3k_B N} \sum_{i=1}^{N} m_i \mathbf{v}_i^2 = \frac{2}{3k_B N} \mathcal{K} \,, \qquad (1.61)$$

[2] Technically, we have $\mathcal{T} = 2\mathcal{K}/k_B N_{\mathrm{dof}}$, where $N_{\mathrm{dof}} = 3N - 3$ is the degree of freedom if the centre of mass momentum of the system is removed.

which, averaged over an MD run over a long time will give us the temperature of a system

$$T = \frac{1}{M} \sum_{m=1}^{M} \mathcal{T}(t_m) \,. \tag{1.62}$$

1.3.1 Velocity Rescaling

Having found a method of measuring temperature in an MD run, the next problem is how to impose a specified temperature on the system and control it during a simulation. Several approaches has been developed, and the most simple one of them is just to scale all particle velocities by a factor determined from the current instantaneous temperature and desired temperature. Suppose the current instantaneous temperature $T(t)$ is considerably different from our desired target temperature, and we want to adjust it to T_0. Then we only need to rescale all current velocities $\mathbf{v}_i$ to

$$\mathbf{v}_i' = \sqrt{\frac{T_0}{T}}\mathbf{v}_i = \sqrt{\frac{K_0}{\mathcal{K}(\mathbf{x}^N(t), \mathbf{p}^N(t))}}\mathbf{v}_i \,. \tag{1.63}$$

In the canonical ensemble, velocities are distributed according to a Gaussian, leading to the famous Maxwell–Boltzmann distributions. The probability functions for each of the three Cartesian components of velocity of every particle i is strictly a Gaussian

$$P(v_{x,i}) = \sqrt{\frac{m_i}{2\pi k_B T}} \exp\left(-\frac{m_i v_{x,i}^2}{2 k_B T}\right) , \tag{1.64}$$

and the same for $v_{y,i}$ and $v_{z,i}$. Temperature rescaling only alters the width of the velocity distribution — it will not change a non-equilibrium distribution (non-Gaussian) into a Gaussian. Due to the chaotic motion of particles, the velocity distribution should eventually converge to a Gaussian, although it may take a while for this to establish.

We can accelerate this equilibration process by interfering with the dynamics more strongly and randomise the velocities by a sampling from a Gaussian distribution — the *thermostats* described in the next section are one approach to do this.

1.3.2 Thermostats

The simple velocity scaling has an apparent advantage of interfering the dynamics minimally by only scaling the velocities and not changing anything else. However, in principle this action is not what a canonical ensemble (or any standard ensemble) does to keep the temperature of a system constant. In addition, this algorithm can produce artifacts when frequently applied because energy will be transferred from other modes to the translational and rotational degrees of freedom — the system acquires high linear momentum and experiences extremely damped internal motions, being frozen into a single conformation, reminiscent of an ice cube flying through space, leading to a so called *flying ice cube* effect. This is wholly unphysical, since it violates the principle of equipartition of energy, which states that the energy should be equally partitioned into every degree of freedom of the molecule.

This problem is solved by using a *thermostat*, which simulates the effect of placing our small simulation system in contact with an infinite heat bath. From the statistical mechanical point of view, this exactly produces a canonical ensemble. There are multiple ways of achieving this, and the approaches can be broadly classified as being stochastic or deterministic. We will focus on stochastic thermostats.

Stochastic thermostats act by adding random noise to the system, which mimic the effect coupling with the heat bath. This will ensure that all accessible constant-energy surfaces are each visited according to their Boltzmann weight. Although this produces an exact canonical distribution, it comes at the expense of interfering with the dynamics, and so transport properties like diffusion will be affected. If those are of interest then the deterministic thermostat will be better.

Andersen Thermostat

The Andersen method mimics the effect of a heat bath by selecting a certain fraction of the particles/atoms at a regular interval to undergo "collisions" with a heat bath. These collisions are characterised by a collision frequency ν. For discrete time steps of length δt, the probability of a particle undergoing a collision is therefore $\nu\delta t$.

To implement this, we just need to randomly select particles for collision with probability $\nu\delta t$ at each time step, and reassign the velocity of the selected particles from a Maxwell–Boltzmann distribution with desired temperature. By doing so, the velocities after collisions are clearly completely uncorrelated with those before, so this procedure will strongly affect the dynamics if the collision frequency is high.

The Andersen thermostat is useful for sampling conformational space, but not so much for the computation of time-dependent properties.

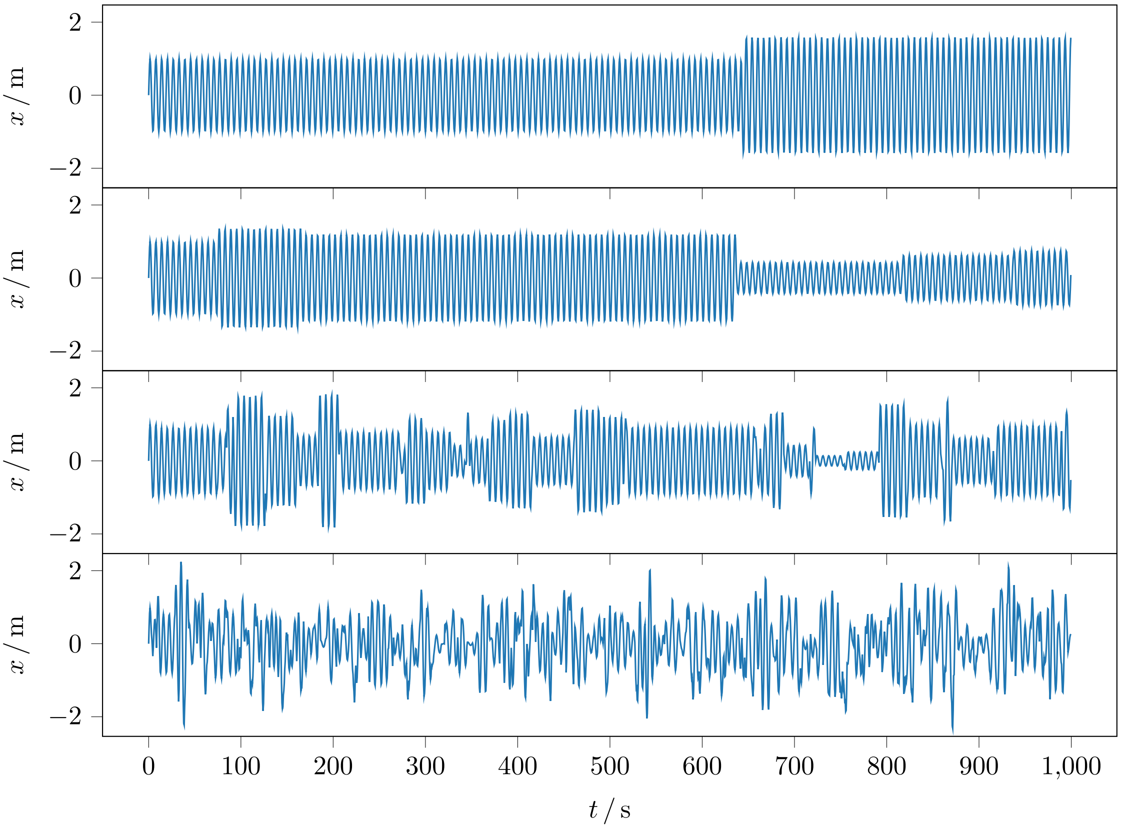


Figure 6: Displacements and velocities of a harmonic oscillator with Andersen thermostats of $\nu = 0.0005, 0.005, 0.05$ and 0.5 respectively.

Canonical Velocity Rescaling (Non-examinable)

Is it possible to somehow combine the advantages of the fairly continuous trajectory afforded by the velocity rescaling with the canonical sampling we desire in Andersen thermostat? The problem

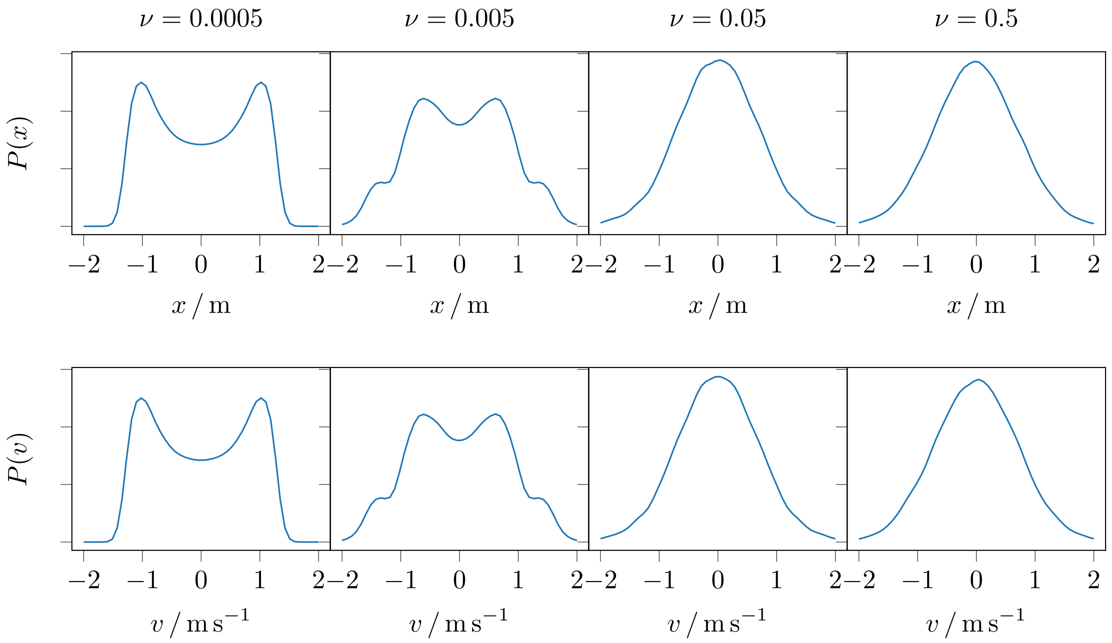


Figure 7: Distributions of displacements and velocities of a harmonic oscillator with Andersen thermostats of different ν.

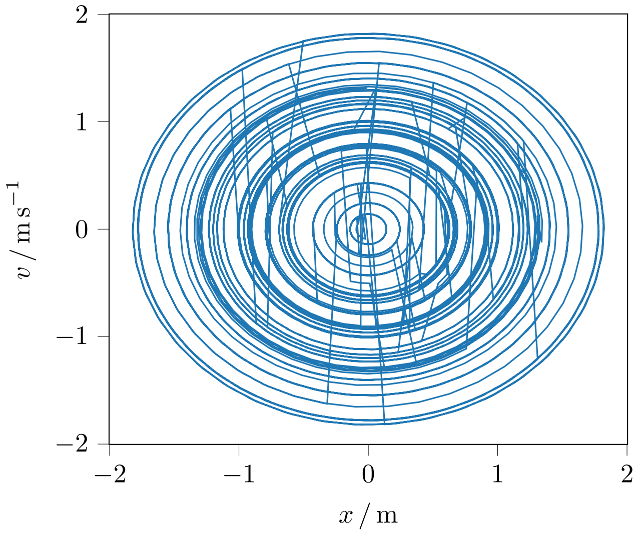


Figure 8: Phase diagram of a harmonic oscillator with Andersen thermostat of $\nu = 0.05$. At this thermalisation frequency, the dynamical nature of the harmonic oscillator is not lost while maintaining an efficient thermalisation such that a large area of phase space is explored according to their Boltzmann weight.

with velocity rescaling is that it will give the correct average kinetic energy by construction, but not necessarily the canonical distribution of kinetic energies. A simple way to fix this is to rescale to match not the average kinetic energy, but a kinetic energy chosen at random from the canonical distribution, given by

$$p(K)\mathrm{d}K = AK^{3N/2-1}\exp(-\beta K)\mathrm{d}K\,, \tag{1.65}$$

where the factor $K^{3N/2-1}$ comes from the volume element in the velocity hyperspace corresponding to the kinetic energy K, and A is a normalisation constant given in this case by

$$A = \frac{\beta^{3N/2}}{\Gamma(\frac{3N}{2})} \,. \tag{1.66}$$

In one dimension, this is more or less the same as the Andersen thermostat. However, in higher dimensions, the width of the kinetic energy distribution is relatively narrow, so the rescaling will generally be close to unity.

Berendsen Thermostat (Non-examinable)

Instead of abruptly rescaling the temperature at a single time step, Berendsen thermostat rescales the temperature by an exponential relaxation

$$\frac{\mathrm{d}T(t)}{\mathrm{d}t} = \frac{T_0 - T(t)}{\tau} \,, \tag{1.67}$$

where T_0 is the desired temperature and τ is the characteristic time scale of relaxation. This leads to a temperature change of

$$\frac{\delta T}{T_0} = \left[1 - \frac{T(t)}{T_0}\right] \frac{\delta t}{\tau} \tag{1.68}$$

per step. This, however, still does not generate the correct kinetic energy distribution, and can lead to serious artifacts. The stochastic terms introduced by Bussi *et al.* fix this problem. It submits the target temperature to a stochastic differential equation

$$\frac{\delta T}{T_0} = \left[1 - \frac{T(t)}{T_0}\right] \frac{\delta t}{\tau} - 2\sqrt{\frac{T(t)}{3T_0 N\tau}}\xi(t) \,, \tag{1.69}$$

where an extra stochastic term is added.

Nosé–Hoover Thermostat (Non-examinable)

Finally, let's briefly introduce a deterministic thermostat. It couples the system to a heat-bath "particle" with mass Q, generalised coordinate s and conjugate momentum p_s, giving what is known as the *Nosé Hamiltonian*

$$\mathcal{H}_N = \sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m_i s^2} + U(\mathbf{r}^N) + \frac{p_s^2}{2Q} + g k_B T f(s) \,, \tag{1.70}$$

where g and $f(s)$ are chosen such that the NVE distribution function for the super-system (including the heat bath) corresponds to a canonical distribution function for the physical subsystem. To do this, we start by calculating the phase-space volume for a given total energy E:

$$\Omega \propto \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \mathrm{d}s \, \mathrm{d}p_s \, \delta\left(\sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m_i s^2} + U(\mathbf{r}^N) + \frac{p_s^2}{2Q} + g k_B T f(s) - E\right) \,. \tag{1.71}$$

We can recover the physical Hamiltonian

$$\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = \sum_{i=1}^{N} \frac{\mathbf{p}_i}{2m_i} + U(\mathbf{r}^N) \tag{1.72}$$

by scaling the momentum by $\mathbf{p}_i \to \mathbf{p}_i/s$, giving

$$\Omega \propto \int \mathrm{d}^{3N}\mathbf{r}^N \, \mathrm{d}^{3N}\mathbf{p}^N \, \mathrm{d}s \, \mathrm{d}p_s \, s^{dN} \delta\left(\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) + \frac{p_s^2}{2Q} + g k_B T f(s) - E\right) \,, \tag{1.73}$$

where d is the dimension of the system. We need to find a way to integrate over the heat-bath coordinates s and p_s. Assuming that the argument of the delta function has a single root at $s = s_0$, then we can rewrite

$$\delta(h(s)) = \frac{\delta(s - s_0)}{|h'(s_0)|}\,, \tag{1.74}$$

and so

$$\Omega \propto \int \mathrm{d}^{3N}\mathbf{r}^N\,\mathrm{d}^{3N}\mathbf{p}^N\,\mathrm{d}p_s\,\frac{s_0^{dN}}{|h'(s_0)|}\,. \tag{1.75}$$

We would like to choose $f(s)$ and g such that

$$\frac{s_0^{dN}}{|h'(s_0)|} \sim \exp\left(-\frac{\mathcal{H}(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)\,. \tag{1.76}$$

It turns out that this can be achieved via $f(s) = \ln(s)$ and $g = dN + 1$, giving

$$s_0 = \exp\left(\frac{E - \mathcal{H}(\mathbf{r}^N,\mathbf{p}^N) - \frac{p_s^2}{2Q}}{gk_BT}\right) \tag{1.77}$$

$$\frac{1}{|h'(s_0)|} = \frac{1}{gk_BT}\exp\left(\frac{E - \mathcal{H}(\mathbf{r}^N,\mathbf{p}^N) - \frac{p_s^2}{2Q}}{gk_BT}\right)\,. \tag{1.78}$$

By doing so, an integration over p_s finally leads to

$$\Omega \propto \frac{\exp\left(\frac{E}{k_BT}\right)\sqrt{2\pi Qk_BT}}{(dN+1)k_BT}\int \mathrm{d}^{3N}\mathbf{p}^N\,\mathrm{d}^{3N}\mathbf{r}^N\,\exp\left(-\frac{\mathcal{H}(\mathbf{r}^N,\mathbf{p}^N)}{k_BT}\right)\,, \tag{1.79}$$

proportional to the canonical distribution. A molecular dynamics NVE simulation with the Hamiltonian $\mathcal{H}_N$ should therefore produce a canonical distribution of $(\mathbf{r}^N, \mathbf{p}^N)$ with Hamiltonian $\mathcal{H}$! The equations of motion can be obtained using Hamilton's equation from classical mechanics.

$$\dot{\mathbf{p}}_i = -\frac{\partial H_N}{\partial \mathbf{r}_i} = \mathbf{f}_i \qquad \dot{\mathbf{r}}_i = \frac{\partial H_N}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i s^2} \tag{1.80}$$

$$\dot{p}_s = -\frac{\partial H_N}{\partial s} = \frac{1}{s}\left[\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i s^2} - gk_BT\right] \qquad \dot{s} = \frac{\partial H_N}{\partial p_s} = \frac{p_s}{Q}\,. \tag{1.81}$$

The equations are, however, not straightforward to implement numerically. Hoover applied a non-canonical transformation to recast these equations into a more amenable form, known as the *Nosé–Hoover equations.*

Algorithm 1.9 (Nosé–Hoover equations).

$$\dot{\mathbf{p}}_i = \mathbf{f}_i - \frac{p_\eta}{Q}\mathbf{p}_i \qquad \dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} \tag{1.82}$$

$$\dot{p}_\eta = \sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{m_i} - dNk_BT \qquad \dot{\eta} = \frac{p_\eta}{Q}\,. \tag{1.83}$$

1.4 Pressure in Molecular Dynamics

Experiments are most often performed at constant pressure, rendering the NPT (isothermal-isobaric) ensemble the most realistic choice. Especially during the initial equilibration, pressure coupling is essential in order to let the volume of the simulation cell adjust.

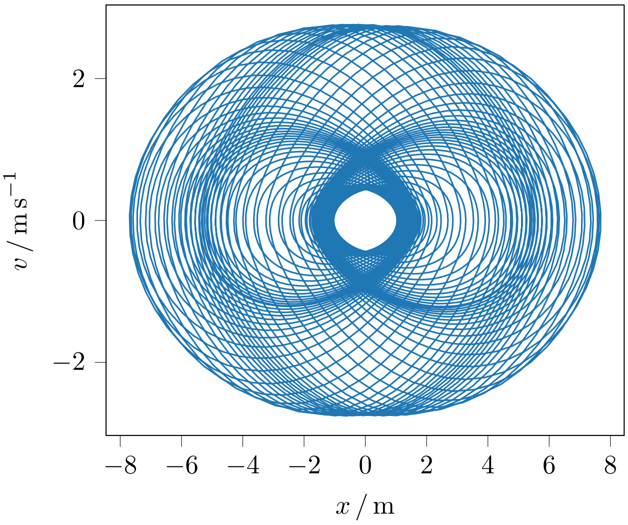


Figure 9: Phase diagram of a harmonic oscillator with Nosé–Hoover thermostat.

We first need an expression for the pressure. The macroscopic pressure can be derived from the free energy, and hence from the bridge relation (1.54), we get

$$\begin{aligned} P &= -\left(\frac{\partial A}{\partial V}\right)_{N,T} = k_B T \left(\frac{\partial \ln Q}{\partial V}\right)_{N,T} \\ &= \frac{k_B T}{Z}\left(\frac{\partial Z}{\partial V}\right)_{N,T} , \end{aligned} \tag{1.84}$$

where

$$Z(N,V,T) = \int \mathrm{d}^{3N}\mathbf{r}^N \, \exp(-\beta \mathcal{U}(\mathbf{r}^N)) \tag{1.85}$$

is the configuration integral. To differentiate with respect to V, we switch to the scaled coordinate defined as

$$\mathbf{s}^N = \frac{1}{L}\mathbf{r}^N = V^{-1/3}\mathbf{r}^N , \tag{1.86}$$

and so

$$Z(N,V,T) = V^N \int \mathrm{d}^{3N}\mathbf{s}^N \, \exp\left[-\beta\mathcal{U}(V^{1/3}\mathbf{s}_1, \ldots, V^{1/3}\mathbf{s}_N)\right] . \tag{1.87}$$

Differentiate with respect to V gives

$$\frac{\partial Z}{\partial V} = \frac{N}{V} Z(N,V,T) - \beta \int \mathrm{d}^{3N}\mathbf{s}^N \left[\frac{1}{3V}\sum_{i=1}^{N} \mathbf{r}_i \cdot \frac{\partial \mathcal{U}}{\partial \mathbf{r}_i} + \frac{\partial \mathcal{U}}{\partial V}\right] \exp(-\beta\mathcal{U}(\mathbf{r}^N)) \tag{1.88}$$

and so

$$\begin{aligned} P &= \frac{k_B T N}{V} + \frac{1}{3V}\left\langle \sum_{i=1}^{N} \mathbf{r}_i \cdot \mathbf{f}_i \right\rangle - \left\langle \frac{\partial \mathcal{U}}{\partial V} \right\rangle \\ &= \frac{1}{3V}\left\langle \sum_{i=1}^{N}\left[\frac{\mathbf{p}_i^2}{m_i} + \mathbf{r}_i \cdot \mathbf{f}_i\right]\right\rangle - \left\langle \frac{\partial \mathcal{U}}{\partial V} \right\rangle . \end{aligned} \tag{1.89}$$

The last term vanishes if $\mathcal{U}$ has no explicit dependence on the volume V. From this expression of the ensemble average of pressure, we get an expression of the *isotropic instantaneous pressure*

$$\mathcal{P} = \frac{1}{3V}\sum_{i=1}^{N}\left[\frac{\mathbf{p}_i^2}{m_i} + \mathbf{r}_i \cdot \mathbf{f}_i\right] \tag{1.90}$$

The quantity

$$\mathcal{W} := \sum_{i=1}^{N} \mathbf{r}_i \cdot \mathbf{f}_i \tag{1.91}$$

is also known as the *virial* of the system, and hence the pressure can also be compactly denoted as

$$\mathcal{P} = \frac{1}{3V} \left(2\mathcal{K} + \mathcal{W}\right) . \tag{1.92}$$

1.4.1 Barostats

As for simulating a canonical ensemble using thermostats, we can simulate an isothermal-isobaric ensemble using a *barostat*. If the pressure coupling is only important during the initial equilibrium phase, it can be acceptable to use the simple Berendsen barostat, which imposes the correct external pressure P_0, but violates the thermodynamic ensemble as it has no correct fluctuations. Similar to Berendsen thermostat, it is also an exponential relation

$$\frac{\mathrm{d}P(t)}{\mathrm{d}t} = \frac{P_0 - P(t)}{\tau} . \tag{1.93}$$

This is achieved via an isotropic scaling factor μ applied to all particle coordinates as well as simulation-cell dimensions

$$\mu = 1 - \frac{\kappa_T \delta t}{3\tau}(P_0 - P) , \tag{1.94}$$

where the κ_T here is the isothermal compressibility.

There are also Nosé–Hoover type barostat, in which coordinates, momenta and volume are coupled to a barostat coordinate ϵ via

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} + \underbrace{\frac{p_\epsilon}{W}\mathbf{r}_i}_{\text{barostat}} \qquad \dot{\mathbf{p}}_i = \mathbf{f}_i \underbrace{- \left(1 + \frac{d}{N}\right) \frac{p_\epsilon}{W}\mathbf{p}_i}_{\text{barostat}} \underbrace{- \frac{p_{\eta,1}}{Q_1}\mathbf{p}_i}_{\substack{\text{thermostat}\\\text{chain}}} . \tag{1.95}$$

In addition, grand canonical ensemble can also be realised using an extended Lagrangian formalism

$$\begin{aligned} L_{\mu VT} = &\sum_{i=1}^{N} \sum_{\alpha} \frac{1}{2} m s^2 \dot{x}_{i,\alpha}^2 - \sum_{i} \sum_{j \neq i, j \neq f} U_{ij} \\ &\underbrace{+ (\nu - N) \left[\frac{1}{2} m s^2 \sum_{\alpha} \dot{x}_{f,\alpha}^2 - \sum_{i \neq f} U_{if} \right] + \frac{1}{2} W \dot{\nu}^2 + U_\nu}_{\text{particle bath}} \\ &\underbrace{+ \frac{1}{2} Q \dot{s}^2 - U_s}_{\text{heat bath}} , \end{aligned} \tag{1.96}$$

in which

$$U_s = g k_B T \ln(s) \tag{1.97}$$

$$U_\nu = \nu \mu = \nu(\mu_{\text{id}} + \mu_{\text{ex}}) . \tag{1.98}$$

However, this scheme is rarely used in practice due to its complexity. Monte Carlo (see later) is much more convenient for this purpose.

1.5 Condensed Phase Simulations in Practice

In this section, we will discuss some practical aspects of molecular dynamics simulations using condensed phase fluid (e.g. liquid argon) as an example. What is so special about liquids? Unlike solids the atoms in liquid have no fixed equilibrium position, so there is no *long-range order* in the form of a lattice. This is true for gases as well, but unlike gases, the particle density in liquid is high, often comparable with the corresponding solid. The atoms in liquids are strongly interacting, establishing a local environment very similar to solids. Liquids exhibit solid-like *short-range order.*

Under Born–Oppenheimer approximation, we can write the potential energy as a function of the nuclear positions only

$$U = U(\mathbf{r}_1, \ldots, \mathbf{r}_N) = U(\mathbf{r}^N)\,, \tag{1.99}$$

and then the force acting on particle i given a nuclear configuration $\mathbf{r}^N$ is

$$\mathbf{f}_i(\mathbf{r}^N) = -\frac{\partial U(\mathbf{r}^N)}{\partial \mathbf{r}_i} = m_i \ddot{\mathbf{r}}_i\,. \tag{1.100}$$

We can then try to partition the Born–Oppenheimer potential energy surface onto many-body terms

$$U(\mathbf{r}^N) = \sum_i U_1(\mathbf{r}_i) + \sum_{i,j>i} U_2(\mathbf{r}_i, \mathbf{r}_j) + \sum_{i,j>i,k>j} U_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \ldots \tag{1.101}$$

We can ignore the one-body term because it will be constant if the space is homogeneous, as this term will be a constant. Three-body interactions and above are usually due to polarisations, and accounts for $\sim 10\%$ to the total interactions in liquid argons. This percentage may be higher for more polarisable systems. If we are assuming interactions are pairwise additive, then we are also ignoring three-body interactions and higher, so we are only left with two-body (pairwise) interactions.

1.5.1 Lennard-Jones Potential

A first step is to choose what model of interaction do we use. A simple example is the Lennard-Jones potential

$$V(\mathbf{r}) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]\,. \tag{1.102}$$

It has a long-range r^{-6} attractive force to account for the van der Waals dispersion, and a short-range r^{-12} repulsive force due to electron density overlap. For argon, $\sigma \approx 3.4\,\text{Å}$ and $\epsilon \approx 1.65 \times 10^{-21}$ J. This also allows us to express everything in reduced units, in which energies, lengths and masses are scaled by $E^* = E/\epsilon$, $r^* = r/\sigma$ and $m^* = m/m_0$, where m_0 is the mass of the particle. Under these definitions, all Lennard-Jones potentials are the same in reduced units

$$V^*(\mathbf{r}) = 4\left[\left(\frac{1}{r^*}\right)^{12} - \left(\frac{1}{r^*}\right)^{6}\right]\,. \tag{1.103}$$

This also allows us to define other scaled quantities, like scales density $\rho^* = \rho\sigma^3$, scales temperature $T^* = k_B T/\epsilon$, scaled pressure $P^* = P\sigma^3/\epsilon$ and scaled time $t^* = \sqrt{\epsilon/m\sigma^2}t$.

1.5.2 System Size and Periodic Boundary Conditions

Having chosen the force field, we can set up our simulation cells. The thermodynamic state of a liquid is specified by only two parameters, the temperature T and the pressure P, or alternatively the temperature T and the particle number density $\rho = N/V$. This means that if you want to simulate a specific state of the liquid, choosing the system size is equivalent to choosing how many particles you want to put into the cell. Due to the limited computational power, we cannot make the cell as large as we want. In the early days of molecular dynamics, the number of atoms was typically on the order of 100. Due to the rapid progress in the performance of computer hardware, this number is continuously increasing. A simulation of systems consisting of 10^5 atoms are common nowadays.

System size in MD is in practice a compromise between the length scale of the problem of interest and the minimum duration of a run required for proper statistical sampling. If we are interested in, for example, the onset of freezing of the system, then the size of the system must be much larger than if we want to study the distribution of atoms in a stable liquid. Similarly, computations of transport properties, such as diffusion coefficients, will require much longer runs than the estimation of internal energy.

However, a real world system rarely consists of $\sim 10^5$ atoms. There will be significant surface/boundary and some finite size effect if we directly use a small system for simulation. We want the system to mimic a bulk, homogeneous liquid or solid. To do this, we can either take a very big cluster and hope that in the interior of the cluster the surface effect can be neglected, or more cleverly, we can use the periodic boundary conditions. To do this, we make our simulation cell into a parallelepiped (e.g. simply a cube), and repeat its contents over the whole space, mimicking the homogeneous state of a liquid or solid. If the MD box is spanned by three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and the cells are displaced by $\ell\mathbf{a} + m\mathbf{b} + n\mathbf{c}$, where $\ell, m, n \in \mathbb{Z}$. This means that if we have a particle in $\mathbf{r}_i$ in the central cell, then there will also be particles at $\mathbf{r}_i + \ell\mathbf{a} + m\mathbf{b} + n\mathbf{c}$ for all $\ell, m, n \in \mathbb{Z}$. This means that we can simulate an infinite system with infinite particles by only calculating the motion of N particles in the central cell.

Still, due to the periodic nature, applying periodic boundary conditions to a small cell will still introduce certain errors, called finite size effect. This can be small or rather serious depending on the nature of the system. Moreover, note that the linear momentum is still a constant of motion in such a set of infinitely replicated systems. The conservation of angular momentum, however, is lost as a result of the reduction of rotational symmetry from spherical to cubic.

1.5.3 The Minimum Image Convention and Truncation of Interaction

Periodic boundary conditions create some other difficulties. Since the size of the simulation cell is now effectively infinite, a particle will now interact with an infinite number of particles, making the force evaluation difficult. The potential energy of the particles in the central cell, corresponding to $(\ell, m, n) = (0, 0, 0)$, is a sum of the interactions over all cells:

$$\mathcal{V}(\mathbf{r}^N) = \frac{1}{2}\sum_{i}^{N} V_i(\mathbf{r}^N) \tag{1.104}$$

$$V_i(\mathbf{r}^N) = \sum_{\ell,m,n=-\infty}^{\infty} \sum_{j}^{N}{}' V(\|\mathbf{r}_j + \ell\mathbf{a} + m\mathbf{b} + n\mathbf{c} - \mathbf{r}_i\|)\,, \tag{1.105}$$

where the $'$ indicates that we are excluding $j = i$ for $\ell, m, n = 0$ (self interaction in the central cell).

For short-range interactions like the van der Waals interaction, it is possible to make the simplification that if the system is sufficiently large, then the contributions of interactions with all images of the same atom, except the nearest, can be disregarded because they are too far away. Notice that the nearest image can be in the same (i.e. central) cell, but it can also be in one of the neighbouring cells if the two particles are more than half a box away in one direction in the central cell. This approximation is known as the *minimum image approximation.* The distance of particle i to the nearest image of particle j can be easily calculated from their positions in the central cell, using

$$\mathbf{r}_{ij}^{\text{MIC}} = \mathbf{r}_j - \mathbf{r}_i - L\,\text{round}\left(\frac{\mathbf{r}_j - \mathbf{r}_i}{L}\right)\,. \tag{1.106}$$

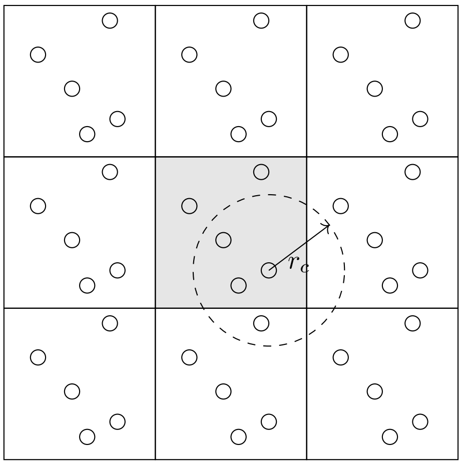


Figure 10: A simulation cell with cyclic boundary condition and cutoff radius $r = r_c$.

One further thing we can do is to truncate the Lennard-Jones potential at some *cutoff* radius. This is because such an interaction decays sufficiently quickly: if $r = 3\sigma$, then $V(r) = -0.005\epsilon$. Therefore, we can choose to neglect the contribution to energy and forces beyond this distances. This is done by letting $V_c(r) = 0$ for $r > r_c$, where r_c is the cutoff radius. However, we need to maintain the continuity of the potential, so we need to shift up the potential for $r \leq r_c$ by $|V(r_c)|$. The truncated potential therefore looks like

$$V_c(r) = \begin{cases} V(r) - V(r_c) & r \leq r_c \\ 0 & r > r_c \end{cases} . \tag{1.107}$$

We would often set the cutoff radius $r_c < \frac{L}{2}$ to avoid self-interaction artefact.

1.5.4 Pair Lists

When the cutoff radius r_c is much smaller than the cell dimensions, a lot of time will be spent checking whether a given pair of atoms is within the cutoff, when in fact most lie outside of it. In this case a useful time-saving measure is to use a "pair list" of atoms which are within $r_c + \delta r$ of each other. Provided that this list is updated sufficiently often that no atom will have moved more than the buffer radius δr between updates, only this list of atom pairs needs to be checked at each time step, rather than all possible pairs.

1.5.5 Initialising Positions and Momenta

For most purposes, we want our simulation to get to equilibrium as fast as possible. You can either pre-arrange the particles on a regular lattice, or place them randomly within the simulation cell. However, putting particles randomly can sometimes lead to extreme coordinates (e.g. two molecules are too close to each other / overlapping), and so sometimes a short steepest-descent is required to remove those situations before the simulation has started.

To set up the initial momenta, we can sample randomly from a Gaussian distribution

$$P(v) \propto \exp\left(-\frac{mv^2}{2k_BT}\right) \tag{1.108}$$

for each component of a particle's velocity. However, these randomly generated velocities sometimes give a non-zero total momentum of the system, so we want to correct non-zero velocities of the centre

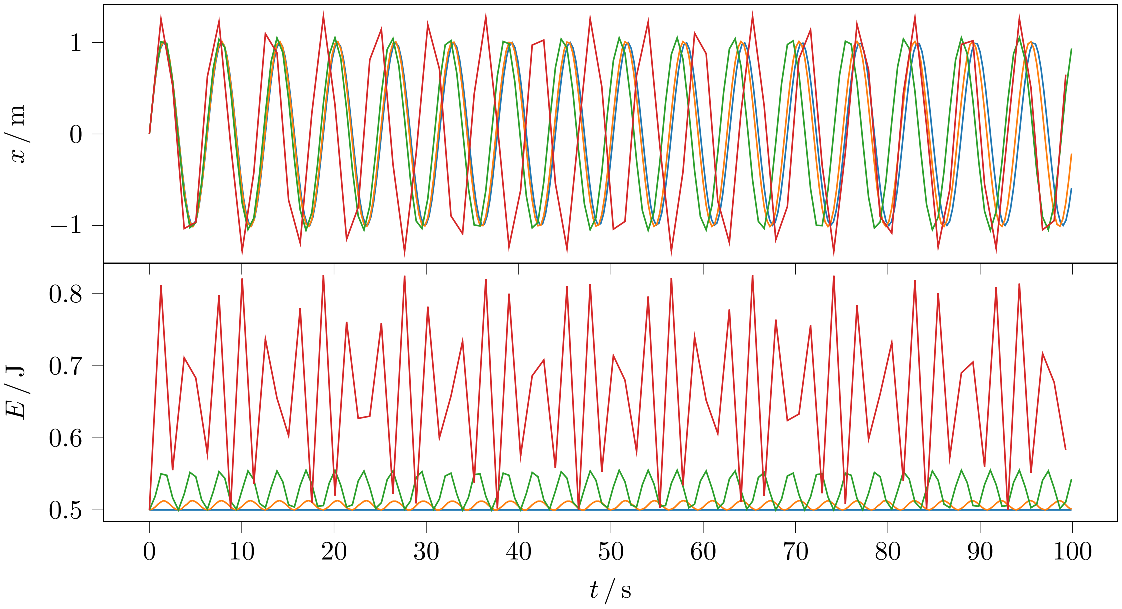


Figure 11: Displacement and energy of a harmonic oscillator using Verlet algorithm with time steps $\delta t = \tau/20$ (orange), $\tau/10$ (green) and $\tau/5$ (red) with exact results (blue).

of mass:

$$\mathbf{v}_{\mathrm{CoM}} = \frac{1}{M} \sum_i m_i \mathbf{v}_i \tag{1.109}$$

$$\mathbf{v}'_i = \mathbf{v}_i - \mathbf{v}_{\mathrm{CoM}}\,. \tag{1.110}$$

Moreover, the randomly generated velocity may deviate from our desired temperature (recall the equipartition theorem), so we need to rescale the velocity by a factor of $\sqrt{T_{\mathrm{target}}/T_{\mathrm{sample}}}$ to reach the desired temperature.

1.5.6 Time Steps

A good choice of time step is a balance between accuracy and computational cost. Too large a time step will lead to large errors in numerical integration. The time step should be significantly smaller than the time scale τ associated with the fastest-frequency oscillation in the system: a step of size $\tau/20$ is usually safe for Verlet-based schemes. The best indication for the breakdown of accuracy in the Verlet scheme due to too large time steps is the drift of total energy, which should be rigorously conserved according to Newton's equation of motion.

1.6 Radial Distribution Function

As we commented above, due to the high number density of atoms and strong interactions, liquid often exhibits short-range order. This can be captured by the *spatial autocorrelation function*

$$g(\mathbf{r}_1, \mathbf{r}_2) := \frac{1}{n_0^2} \langle n(\mathbf{r}_1) n(\mathbf{r}_2) \rangle \tag{1.111}$$

where $n_0 = N/V$ is the average particle density, just like the velocity autocorrelation function you met in Part II Statistical Mechanics. In an isotropic, homogeneous medium, the autocorrelation function

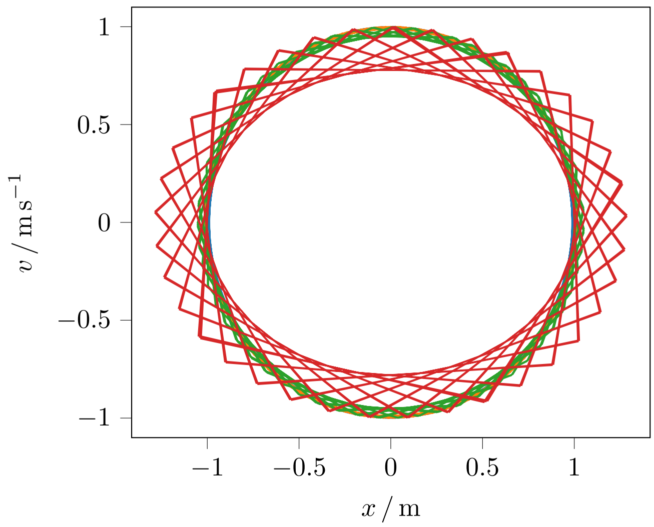
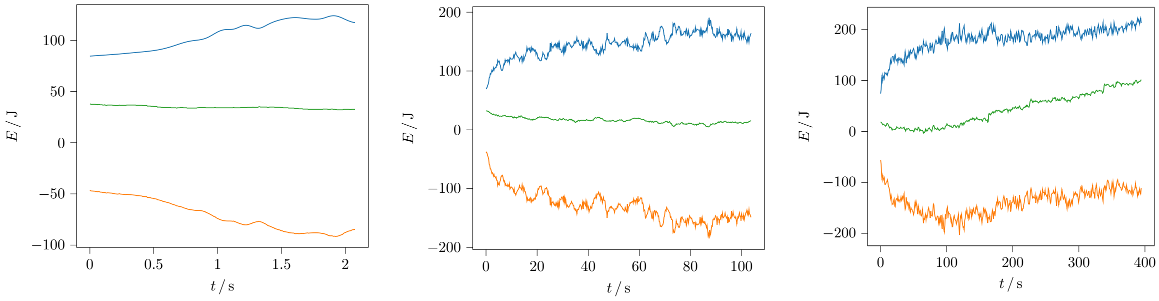


Figure 12: Phase diagram of a harmonic oscillator using Verlet algorithm with time steps $\delta t = \tau/20$ (orange), $\tau/10$ (green) and $\tau/5$ (red) with exact results (blue).



(a) $\delta t^* = 0.001$

(b) $\delta t^* = 0.05$

(c) $\delta t^* = 0.19$

Figure 13: A plot of the computed kinetic energy (blue), potential energy (orange) and total energy (green) of a Lennard-Jones fluid with 100 particles using times steps 0.001, 0.05 and 0.19 in reduced time units. 5000 steps are used for each run. A significant energy drift occurred for $\delta t^* = 0.19$, suggesting that this time step is too large.

should be only dependent on the particle distance $r = \|\mathbf{r}_1 - \mathbf{r}_2\|$, and so it allows us to define the *radial distribution function*

$$g(r) = \frac{\langle n(r) \rangle}{n_0} \,. \tag{1.112}$$

This can be understood as a histogram of two-particle distances, which can be constructed by going through the following steps:

1. Pick a reference particle i at position $\mathbf{r}_i$.
2. Draw a spherical shell of radius r and thickness Δr centred at $\mathbf{r}_i$. Determine the number of particles in the shell, which we denote as $n_i(r, \Delta r)$. A particle j in this shell would satisfy

   $$r \leq \|\mathbf{r}_i - \mathbf{r}_j\| < r + \Delta r \,. \tag{1.113}$$

   A small Δr will give us a high resolution, but it will introduce noise. A large r will smooth the distribution function, but we may lose some of the structural details.
3. Divide $n_i(r, \Delta r)$ by the volume of the shell to convert it into a number density, and average over reference particles.

   $$\bar{\rho}(r, \Delta r) = \frac{1}{N} \sum_{i=1}^{N} \frac{n_i(r, \Delta r)}{4\pi r^2 \Delta r} \,, \tag{1.114}$$

   where we assumed $\Delta r \ll r$.
4. Normalise this quantity by the particle number density of $\rho = (N-1)/V$

   $$g(r) = \frac{\bar{\rho}(r, \Delta r)}{\rho} = \frac{V}{4\pi r^2 \Delta r N(N-1)} \sum_{i}^{N} n_i(r, \Delta r) \,. \tag{1.115}$$

   The resulting $g(r)$ is a dimensionless quantity. Note that we use $N - 1$ because we do not consider a particle correlating with itself, although for large N, we can replace $N - 1$ by N.

A $g(r) > 1$ at some r would mean that we have an enhanced probability of meeting another particle at distance r away from one particle, and $g(r) < 1$ means that there is a depletion region.

A schematic construction of the radial distribution function for a 2D Lennard-Jones fluid is shown in figure 14. For distances $r \ll \sigma$, where σ is the repulsive core diameter, the radial distribution $\to 0$ because particles are excluded from this region. The maximum at a distance little over σ reflects the well defined coordination shell of nearest neighbours around a particle in a liquid. This peak in $g(r)$ is characteristic for the high density prevalent in liquids and is absent in the vapour phase. In most liquids there is also a broader second nearest neighbour peak. As a result of the disorder in liquids, this structure is considerably less pronounced compared to solids. For distances larger than second neighbours, fluctuations gradually takes over and the distribution of atoms becomes homogeneous, with $g(r) \to 1$.

As for a comparison, a crystalline solid has long-range order extending to infinity, so the radial distribution function should have well defined peaks extending to infinity (ideally) except being broadened by thermal vibrations at non-zero temperatures.

1.6.1 Coordination Numbers

As suggested by figure 14 the integral of the first peak of RDF is related to the average number of particles n_c in the first coordination shell, which is also known as the *coordination number*. In order to measure n_c we must specify how close an atom must approach the central particle in order to be counted as a first neighbour. The position r_{min} of the minimum between the first and second

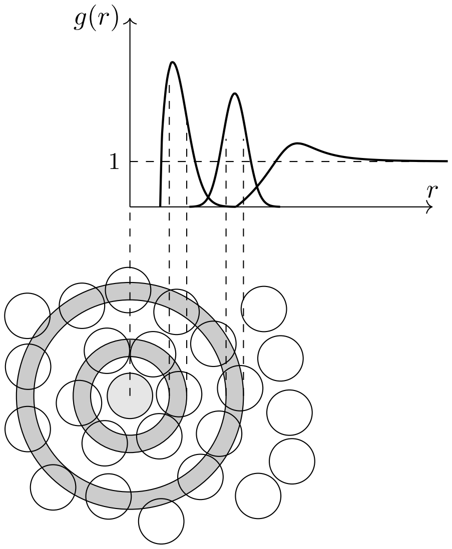


Figure 14: A typical radial distribution function of liquid. $g(r) = 0$ for small r due to hard-core repulsion. There are several peaks with $g(r) > 1$ at intermediate range, corresponding to the first, second and higher coordination shells. At large r, $g(r) \to 1$ as the distribution of particles become decorrelated.

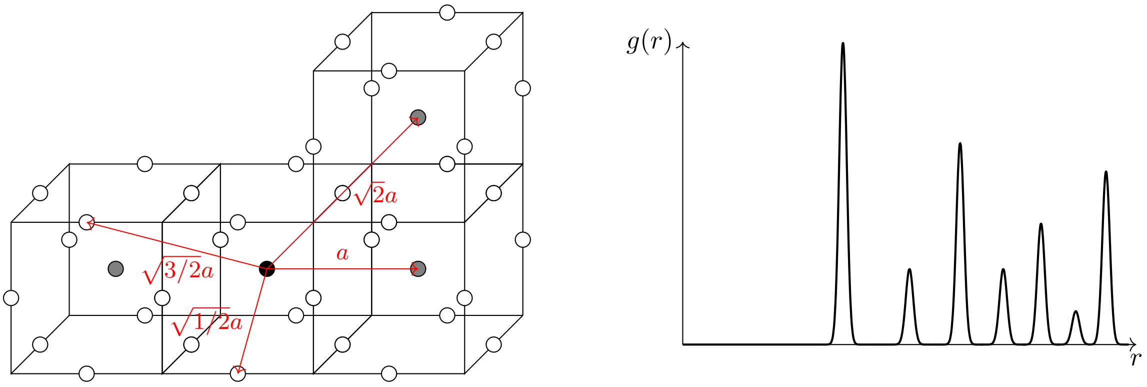


Figure 15: Radial distribution function in a solid fcc crystal.

maximum is used as a common (but not unique) criterion to define the first coordination shell. n_c is then found from the integral (for three dimensional cases)

$$n_c = 4\pi\rho \int_0^{r_c} \mathrm{d}r\, r^2 g(r)\,, \tag{1.116}$$

with $r_c = r_{\min}$ in our cases.

1.6.2 Radial Distribution Function in Statistical Mechanics

The way we introduced the radial distribution function was operational in the sense that it is based on how this quantity is determined in a simulation. We will now give a proper statistical mechanical definition of the radial distribution function making use of the Dirac delta function. We first consider a more general pair correlation function which not only probes the distance r between particles but also the orientation of the displacement vector $\mathbf{r}$.

$$\gamma(\mathbf{r}) = \frac{V}{N(N-1)} \left\langle \sum_{i,j\neq i}^{N} \delta(\mathbf{r}_{ij} - \mathbf{r}) \right\rangle\,, \tag{1.117}$$

where the angular bracket denote an integral over the configurational probability distribution function (1.58). In a more explicit form,

$$\gamma(\mathbf{r}) = \frac{V}{N(N-1)} \sum_{i,j\neq i}^{N} \int \mathrm{d}^{3N}\mathbf{r}^N\, P_N(\mathbf{r}^N)\delta(\mathbf{r}_{ij} - \mathbf{r})\,. \tag{1.118}$$

It is proportional to the probability of observing two particles separated by a vector $\mathbf{r}$. This correlation function with the information on the orientation of $\mathbf{r}$ included is suitable for crystalline solids, in which the lattice defines special directions in space. Liquids are isotropic systems with no preference for a direction, so we expect the function $\gamma(\mathbf{r})$ to depend only on the length $r = \|\mathbf{r}\|$. Using this property, we integrate over a spherical shell $V(r, \Delta r)$ with radius between r and $r + \Delta r$, and assuming that the shell is sufficiently thin, we then write

$$\int_{V(r,\Delta r)} \mathrm{d}^3\mathbf{r}\, \gamma(\mathbf{r}) \approx 4\pi r^2 \Delta r g(r)\,. \tag{1.119}$$

Here the $g(r)$ is the radial distribution function. Substituting the expression of $\gamma(\mathbf{r})$, we get

$$\begin{aligned} g(r) &\approx \frac{V}{4\pi r^2 \Delta r N(N-1)} \int_{V(r,\Delta r)} \mathrm{d}^3\mathbf{r} \sum_{i,j\neq i} \int \mathrm{d}^{3N}\mathbf{r}^N\, P_N(\mathbf{r}^N)\delta(\mathbf{r}_{ij} - \mathbf{r}) \\ &= \frac{V}{4\pi r^2 \Delta r N(N-1)} \int \mathrm{d}^{3N}\mathbf{r}^N\, P_N(\mathbf{r}^N) \sum_{i,j\neq i} \int_{V(r,\Delta r)} \mathrm{d}^3\mathbf{r}\, \delta(\mathbf{r}_{ij} - \mathbf{r})\,, \end{aligned} \tag{1.120}$$

where we have changed the order of integration. The inner integral gives unity when the vector $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ lies within volume $V(r, \Delta r)$ and zero otherwise. The delta function therefore counts the number of particle pairs with distances between r and $r + \Delta r$. Taking particle i as reference we recover the quantity $n_i(r, \Delta r)$ introduced before

$$n_i(r, \Delta r) = \sum_{j\neq i} \int_{V(r,\Delta r)} \mathrm{d}^3\mathbf{r}\, \delta(\mathbf{r}_{ij} - \mathbf{r})\,. \tag{1.121}$$

Therefore,

$$\begin{aligned} g(r) &\approx \frac{V}{4\pi r^2 \Delta r N(N-1)} \int \mathrm{d}^{3N}\mathbf{r}^N\, P_N(\mathbf{r}^N) \sum_i n_i(r, \Delta r) \\ &= \frac{V}{4\pi r^2 \Delta r N(N-1)} \left\langle \sum_i n_i(r, \Delta r) \right\rangle\,. \end{aligned} \tag{1.122}$$

This becomes an equality in the limit of an infinitely thin shell,

$$g(r) = \lim_{\Delta r \to 0} \frac{V}{4\pi r^2 \Delta r N(N-1)} \left\langle \sum_i n_i(r, \Delta r) \right\rangle . \tag{1.123}$$

Since all particles are equivalent, we must have

$$g(r) = \frac{V}{N-1} \lim_{\Delta r \to 0} \frac{1}{4\pi r^2 \Delta r} \langle n(r, \Delta r) \rangle , \tag{1.124}$$

where we removed the subscript i to indicate a generic particle. We identify this linked to our instantaneous, histogramic radial distribution function (1.115) via

$$g_{\mathrm{stat}}(r) = \frac{N-1}{N} g_{\mathrm{hist}}(r) , \tag{1.125}$$

where $(N-1)/N$ is a good approximation to 1 except at very low density. We could have avoided this complexity if we used N^2 as the normalisation factor in the definition of pair correlation function (1.117), but it turns out that $N(N-1)$ is the correct normalisation factor for a good treatment at very low density, i.e. in a gas phase limit. Consider a system with only two particles ($N = 2$) in a finite volume V,

$$\begin{aligned} \gamma(\mathbf{r}) &= \frac{V}{N(N-1)} \left\langle \sum_{i,j\neq i} \delta(\mathbf{r}_{ij} - \mathbf{r}) \right\rangle = V \langle \delta(\mathbf{r}_{ij} - \mathbf{r}) \rangle \\ &= V \int \mathrm{d}^3\mathbf{r}_1 \, \mathrm{d}^3\mathbf{r}_2 \, P_2(\mathbf{r}_1, \mathbf{r}_2) \delta(\mathbf{r}_{12} - \mathbf{r}) . \end{aligned} \tag{1.126}$$

P_2 is just the Boltzmann exponent of the pair potential, and so

$$\gamma(\mathbf{r}) = V \frac{\int \mathrm{d}^3\mathbf{r}_1 \, \mathrm{d}^3\mathbf{r}_2 \, \exp[-V(r_{12})/k_BT] \delta(\mathbf{r}_{12} - \mathbf{r})}{\int \mathrm{d}^3\mathbf{r}_1 \, \mathrm{d}^3\mathbf{r}_2 \, \exp[-V(r_{12})/k_BT]} . \tag{1.127}$$

Transforming to the CoM coordinate $\mathbf{r}$ and relative coordinate $\mathbf{r}' = \mathbf{r}_{12}$ gives

$$\gamma(\mathbf{r}) = V \frac{\int \mathrm{d}^3\mathbf{R} \, \mathrm{d}^3\mathbf{r}' \, \exp[-V(r'_{12})/k_BT] \delta(\mathbf{r}' - \mathbf{r})}{\int \mathrm{d}^3\mathbf{R} \, \mathrm{d}^3\mathbf{r}' \, \exp[-V(r'_{12})/k_BT]} . \tag{1.128}$$

Since the potential is independent of $\mathbf{R}$, the integral over $\mathbf{R}$ gives a factor V,

$$\gamma(\mathbf{r}) = V \frac{\int \mathrm{d}^3\mathbf{r}' \, \exp[-V(r')/k_BT] \delta(\mathbf{r}' - \mathbf{r})}{\int \mathrm{d}^3\mathbf{r} \, \exp[-V(r')/k_BT]} = V \frac{\exp[-V(r')/k_BT]}{\int \mathrm{d}^3\mathbf{r}' \, \exp[-V(r')/k_BT]} . \tag{1.129}$$

For $V^{1/3} \gg \sigma$, we can identify

$$\int \mathrm{d}^3\mathbf{r}' \, \exp[-V(r')/k_BT] \approx V , \tag{1.130}$$

and so

$$\gamma(\mathbf{r}) = \mathrm{e}^{-V(r)/k_BT} . \tag{1.131}$$

As expected, $\gamma(\mathbf{r})$ is isotropic and we can directly write

$$g(r) = \mathrm{e}^{-V(r)/k_BT} , \tag{1.132}$$

or taking the logarithms,

$$-k_BT \ln g(r) = V(r) . \tag{1.133}$$

In the low density limit of a many-particle gas, (1.132) is still the leading term in an expansion in terms of powers of density, and so it remains approximately valid. This suggests that there is a close relation between the radial distribution function and pair potential. The equation (1.133) can be generalised to the condensed phase, with the logarithm of $g(r)$ identified as an effective pair potential. This effective pair potential, which is really a free energy, is called the *potential of mean force* and plays a crucial role in the statistical mechanics of liquids and solutions.

1.6.3 Relation of Other Properties to the Radial Distribution Function

For simple single-component atomic liquids, with a radial potential $V(r)$, it turns out that many properties can be simply computed from the $g(r)$, which contains all the information necessary about the statistics of pair distances. For example, one can show

$$\frac{U}{N} = 2\pi\rho \int_0^\infty \mathrm{d}r\, V(r) g(r) r^2 \tag{1.134}$$

$$P = \rho k_B T - \frac{2}{3}\pi\rho^2 \int_0^\infty \mathrm{d}r\, \frac{\mathrm{d}V(r)}{\mathrm{d}r} g(r) r^3\,. \tag{1.135}$$

1.6.4 Experimental Determination of Radial Distribution Function

Radial distribution functions can be observed in experiment from the diffraction patterns of radiation with a wavelength comparable to the interatomic distances. This means that for normal liquids with interatomic distances in the order of angstroms, we can use neutrons or X-rays.[3] The quantity that is actually directly measured from a diffraction experiment is the intensity $I(\theta)$ scattered in a direction at angle θ to the incoming beam. If $\mathbf{k}_{\text{in}}$ and k_{out} are the wavevectors of the incoming and outgoing beam, then the momentum transferred is

$$\mathbf{k} = \mathbf{k}_{\text{in}} - \mathbf{k}_{\text{out}}\,. \tag{1.136}$$

In elastic scatterings, $\|\mathbf{k}_{\text{in}}\| = \|\mathbf{k}_{\text{out}}\|$, and therefore

$$k = \|\mathbf{k}\| = \frac{4\pi}{\lambda} \sin\left(\frac{\theta}{2}\right)\,. \tag{1.137}$$

To a very good approximation, the observed scattered intensity can be separated into an atomic form factor $f(k)$ and a structure factor $S(k)$

$$I(\theta) = |f(k)|^2\, S(k)\,. \tag{1.138}$$

The form factor is specific to the atomic species and also depends on instrumental corrections. The *structure factor* is given by

$$S(\mathbf{k}) = \frac{1}{N} \left\langle \sum_{\ell,m} \exp[\mathrm{i}\mathbf{k}\cdot(\mathbf{r}_\ell - \mathbf{r}_m)] \right\rangle \tag{1.139}$$

and contains all the information on the position of the particles. Similar to the RDF we have used a more general formulation allowing for possible dependence on the direction of the momentum transfer as is the case for example for Bragg scattering of crystals. For liquids, however, the structure factor is isotropic and only depends on the magnitude $k = \|\mathbf{k}\|$ of the scattering vector. To relate the structure factor to the radial distribution we use the formal definition of the RDF in terms of the delta function (1.117) and the identity

$$\delta(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}k\, \mathrm{e}^{\mathrm{i}kx}\,. \tag{1.140}$$

To proceed, we first separate the sum (1.139) in $\ell = m$ and $\ell \neq m$ term

$$\begin{aligned} S(\mathbf{k}) &= \frac{1}{N} \left\langle \sum_{\ell}^{N} \exp[\mathrm{i}\mathbf{k}\cdot\mathbf{0}] \right\rangle + \frac{1}{N} \left\langle \sum_{\ell,m\neq\ell} \exp[\mathrm{i}\mathbf{k}\cdot(\mathbf{r}_\ell - \mathbf{r}_m)] \right\rangle \\ &= 1 + \frac{1}{N} \left\langle \sum_{\ell,m\neq\ell} \exp[\mathrm{i}\mathbf{k}\cdot\mathbf{r}_{\ell m}] \right\rangle\,. \end{aligned} \tag{1.141}$$

[3] For more on scattering, see my notes on Part II C6: *Diffraction Methods in Chemistry.*

Its inverse Fourier transform is then given by

$$\frac{1}{(2\pi)^3}\int \mathrm{d}^3\mathbf{k}\,\mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{r}}S(\mathbf{k}) = \delta(\mathbf{r}) + \frac{1}{N}\left\langle \sum_{\ell,m\neq\ell}\delta(\mathbf{r}_{\ell m}-\mathbf{r})\right\rangle , \tag{1.142}$$

and now by (1.117) and setting $N-1\approx N$, we get

$$\frac{1}{(2\pi)^3}\int \mathrm{d}^3\mathbf{k}\,\mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{r}}S(\mathbf{k}) = \delta(\mathbf{r}) + \frac{N}{V}\gamma(\mathbf{r})\,. \tag{1.143}$$

Alternatively,

$$\frac{1}{(2\pi)^2}\int \mathrm{d}^3\mathbf{k}\,\mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{r}}[S(\mathbf{k})-1] = \frac{N}{V}\gamma(\mathbf{r})\,. \tag{1.144}$$

The $g(r)$ can therefore be obtained from experiment by the inverse Fourier transform of the measured structure factor after subtracting the “self-correlation”.

1.7 Static Versus Dynamical Averages

Under the ergodic hypothesis, we can use molecular dynamics to sample from microcanonical ensemble, and by weak-coupling approaches, some other standard statistical mechanical ensembles can be generated as well. All the quantities we introduced above do not incorporate explicit knowledge of the time evolution of the system. They are just static equilibrium averages that can equally well be obtained via the Monte-Carlo route. If we are only interested in these static properties, we don’t need to track how the system evolves in time. We just need a good equilibrium sampling.

However, some quantities are not like that. They are directly dependent on how the system evolves in time, and so we have to use time-dependent simulations instead of Monte-Carlo. They are called *dynamical properties*. Some examples include transport or diffusive properties (e.g. heat conductivity and diffusion coefficient), rheological properties (e.g. shear modulus) and spectroscopic properties.

However, Green–Kubo relations allow us to directly link those macroscopic transport coefficients to the microscopic fluctuations at equilibrium by relating the transport coefficients to the integral of the autocorrelation functions of certain quantities. For example, the diffusion coefficient is linked to the integral of the velocity autocorrelation function, and the viscosity is linked to the stress tensor. Taking advantage of these, we can extract transport coefficients directly from a equilibrium molecular dynamics simulation instead of observing how a system evolves to equilibrium.

The time autocorrelation functions capture the rate at which a system decorrelates from an initial state. Consider a signal (e.g. a phase function) $A(t)$ with domain $t\in[0,\Delta t]$. Without loss of generality, we can let $\langle A\rangle = 0$. The autocorrelation function of A is then defined as

$$C_{AA}(\tau) \coloneqq \frac{1}{\Delta t-\tau}\int_0^{\Delta t-\tau}\mathrm{d}t\,A(t+\tau)A(t)\,. \tag{1.145}$$

We have

$$C_{AA}(\tau\to 0) = \left\langle A^2\right\rangle \tag{1.146}$$

$$C_{AA}(\tau\to\Delta t) = \langle A\rangle^2\ . \tag{1.147}$$

Hence, one can extract the transport coefficients from an MD simulation at equilibrium using Green–Kubo relations from the following scheme:

1. Run a simulation at equilibrium.
2. Compute relevant microscopic quantity. For example, if we are interested in the diffusion coefficient, then we can calculate the particle’s velocity.

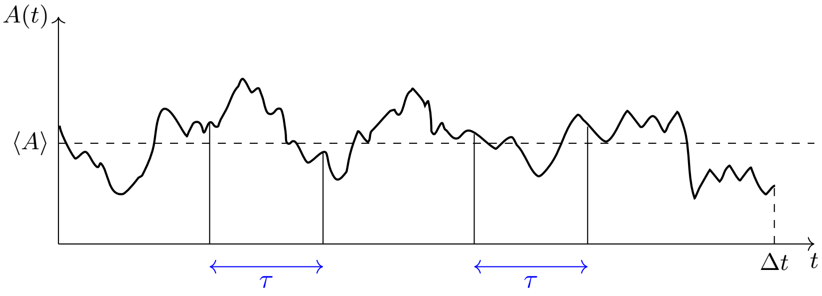


Figure 16: Constructing the autocorrelation function of a quantity.

3. Calculate the autocorrelation function.
4. Integrate to get the transport coefficient.

One can generalise the autocorrelation function to define the *cross-correlation function*[4]

$$C_{\mathrm{AB}}(\tau) := \frac{\langle A(t)B(t+\tau)\rangle}{\langle A\rangle\langle B\rangle}\,. \tag{1.149}$$

Where the normalisation factor may be there or not. This will be useful when studying coupled processes, like energy transfer or reaction rates. The loss of correlation with time is usually an exponential decay, and the decorrelation time can be estimated from a time integral over the correlation function

$$\tau_{AB} = \frac{1}{C_{AB}(0)}\int_0^\infty \mathrm{d}\tau\, C_{AB}(\tau)\,. \tag{1.150}$$

One may check that the correlation function has the time translation invariance and time inversion symmetry

$$\langle A(t+\tau)B(t)\rangle = \langle A(t'+\tau)B(t')\rangle \tag{1.151}$$
$$\langle A(\tau)A(0)\rangle = \langle A(0)A(-\tau)\rangle\ . \tag{1.152}$$

1.7.1 Fluctuation-Dissipation Theorem

In fact, the use of time correlation function at equilibrium in obtaining off-equilibrium properties like transport coefficient is a very deep and profound result related to the fluctuation-dissipation theorem.

An informal statement of this theorem is Onsager's *regression hypothesis*: the relaxation of a system after some macroscopic perturbation is governed by the same laws as the regressions of spontaneous microscopic fluctuations in an equilibrium system. A system's fluctuation at equilibrium is indistinguishable from its approach from off-equilibrium to equilibrium, provided that it is not too far away from equilibrium. The fluctuation-dissipation theorem formalises this hypothesis.

Theorem 1.10 (Fluctuation-dissipation theorem). Consider a system that has been prepared in a state sampled from some non-equilibrium distribution $F(\mathbf{r}^N,\mathbf{p}^N)$ by the means of a small perturbation ΔH. Write the average of the observable A in the perturbed state as

$$\overline{A}(t) = \int \mathrm{d}^{3N}\mathbf{r}^N\,\mathrm{d}^{3N}\mathbf{p}^N\, A(t\mid\mathbf{r}^N,\mathbf{p}^N)F(\mathbf{r}^N,\mathbf{p}^N)\,. \tag{1.153}$$

[4] More formally, it can also be defined using a phase-space average as an ergodic integral

$$C_{AB}(\tau) := \int \mathrm{d}^{3N}\mathbf{r}^N\,\mathrm{d}^{3N}\mathbf{p}^N\, A(\mathbf{r}^N,\mathbf{p}^N)B(\tau\mid\mathbf{r}^N,\mathbf{p}^N)\rho(\mathbf{r}^N,\mathbf{p}^N)\,. \tag{1.148}$$

If the initial state is not too far from equilibrium, the time evolution follows

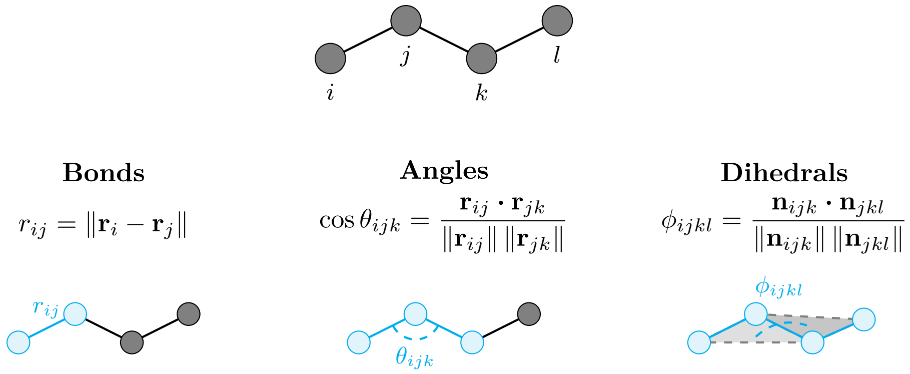
$$\frac{\overline{A}(t)}{\overline{A}(0)} = \frac{C_{AA}(t)}{C_{AA}(0)}\,. \tag{1.154}$$

Hence, instead of measuring how a system responds to an external force, we can directly measure how it fluctuates at equilibrium.

1.8 Force Fields

A force field is the classical parameterisation of the adiabatic potential energy surface $V(\mathbf{r}^N)$. It can be separated into bonded and non-bonded potentials to include both bonded and non-bonded interactions.

Bonded potentials capture covalent, intramolecular interactions. They are usually partitioned onto n-body terms up to $n = 4$.



$$r_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|$$

$$\cos\theta_{ijk} = \frac{\mathbf{r}_{ij}\cdot\mathbf{r}_{jk}}{\|\mathbf{r}_{ij}\|\,\|\mathbf{r}_{jk}\|}$$

$$\phi_{ijkl} = \frac{\mathbf{n}_{ijk}\cdot\mathbf{n}_{jkl}}{\|\mathbf{n}_{ijk}\|\,\|\mathbf{n}_{jkl}\|}$$

Some commonly used bonded potentials are

- Harmonic bond.
$$u(r_{ij}) = \frac{1}{2}k_r(r_{ij} - r_0)^2\,. \tag{1.155}$$
- Harmonic angle.
$$u(\theta_{ijk}) = \frac{1}{2}k_\theta(\theta_{ijk} - \theta_0)^2\,. \tag{1.156}$$
- Proper dihedral. It simulates the conditions where several dihedral angles are likely, and it has many probable forms. The Ryckaert–Bellemans form is
$$u_{rb}(\phi_{ijkl}) = \sum_{n=0}^{5} c_n \cos^n(\phi_{ijkl})\,. \tag{1.157}$$
It is a 2π-periodic function with several minima and maxima.
- Improper dihedral. It is used to capture the rigidity of certain systems like planar BF_3, where only one specific dihedral angle is probable.
$$u_{id}(\phi_{ijkl}) = \frac{1}{2}k_\phi(\phi_{ijkl} - \phi_0)^2\,. \tag{1.158}$$

Non-bonded potentials capture van der Waals, electrostatic and polarisation interactions at both intermolecular and intramolecular levels. These interactions are often approximated using only pairwise terms.

Common non-bonded potentials include:

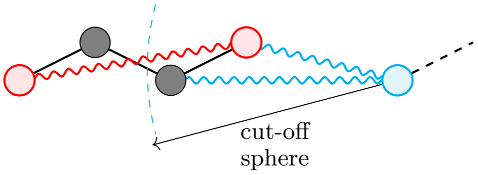


Figure 17: Both intermolecular and intramolecular non-bonded forces should be considered. However, 1-2 and 1-3 interactions are often excluded.

- Lennard-Jones potential

$$u_{\mathrm{LJ}}(r_{ij}) = 4\epsilon_{ij} f_{\mathrm{LJ}}(r_{ij}) \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} \right) , \tag{1.159}$$

  where $f_{ij}(r_{ij})$ is a cut-off function such that $f_{\mathrm{LJ}}(r_{ij}) = 0$ for r_{ij} above the cut-off radius and one otherwise. The most common way obtaining these coefficients is the Lorentz–Berthelot rules

$$\epsilon_{ij} = \sqrt{\epsilon_i \epsilon_j} \qquad \sigma_{ij} = \frac{\sigma_i + \sigma_j}{2} \tag{1.160}$$

$$A_{ij} = 4\epsilon_{ij}\sigma_{ij}^{12} \qquad B_{ij} = 4\epsilon_{ij}\sigma_{ij}^{6} . \tag{1.161}$$

- Coulomb potential

$$u_{ij} = f_c(r_{ij}) \frac{1}{4\pi\epsilon_0} \frac{q_i q_j}{r_{ij}} . \tag{1.162}$$

1.9 Errors in Simulations

Broadly speaking, the errors in an MD simulation can be classified into two categories. First, there are systematic errors that are inherent to the system. It is related to the model and method one uses to describe a given system (e.g. force field, quantum mechanical effect), and it is harder to gain control over. There are also *statistical errors* that can arise in either time domain due to finite sampling, or in spatial domain due to finite size effects. We will focus on statistical errors.

1.9.1 Block Average Method

If we have a trajectory of length τ, the average of a dynamical observable A over this trajectory can be calculated as

$$A_\tau = \frac{1}{\tau} \int_0^\tau \mathrm{d}t\, A(t) . \tag{1.163}$$

In the limit of infinite simulation time, the estimated average should converge to the true ensemble average

$$\lim_{\tau \to \infty} \langle A_\tau \rangle = \langle A \rangle . \tag{1.164}$$

But we can only simulate the system for a finite amount of time. Then how do we calculate the error of our estimate A_τ?

To estimate the error, we can consider finite blocks that partition the trajectory onto smaller chunks, and then evaluate the average of A over each block. For a given choice of τ, the variance of A_τ is given by

$$\sigma^2(A_\tau) = \sum_{b=1}^{n_b} \frac{(A_{\tau,b} - A_{\Delta t})^2}{n_b} . \tag{1.165}$$

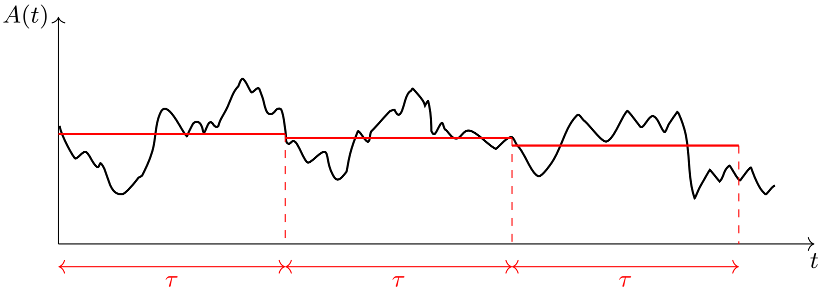


Figure 18: Estimating the statistical error using the block average method.

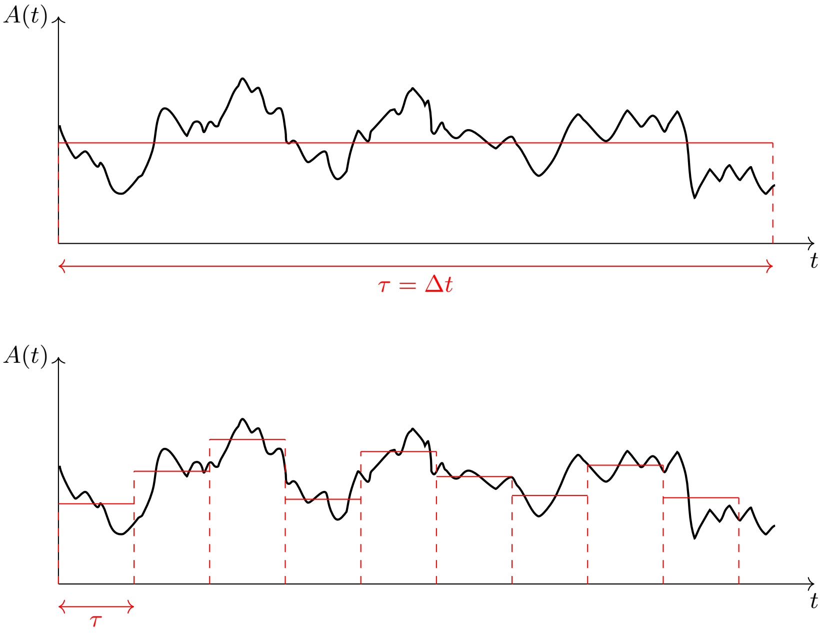


Figure 19: Using different block sizes to estimate the error. The first case clearly underestimates the error, and the second case overestimates it.

Clearly, the estimated variance will be strongly dependent on the choice of τ. If we use $\tau = \Delta t$, then we get $\sigma^2(A_\tau) = 0$ and this clearly underestimates the error. If we choose too small a τ, then we would overestimate the error. How do we choose the appropriate block size?

Recall that $\lim_{\tau\to\infty} \langle A_\tau \rangle = \langle A \rangle$. We can determine an appropriate block size by relating the variance of our time average to a time autocorrelation function

$$\sigma^2(A_\tau) = \left\langle A_\tau^2 \right\rangle - \langle A_\tau \rangle^2 = \frac{1}{\tau^2}\int_0^\tau \mathrm{d}t \int_0^\tau \mathrm{d}t' \, \underbrace{\langle [A(t) - \langle A \rangle][A(t') - \langle A \rangle] \rangle}_{=C_{AA}(t-t')} \,. \tag{1.166}$$

The important time scale for this autocorrelation function is the characteristic decay of C_{AA}

$$\tau_A = \frac{1}{2}\int_{-\infty}^{\infty} \mathrm{d}t \, \frac{C_{AA}(t)}{C_{AA}(0)} \,. \tag{1.167}$$

Therefore, if the block sampling time τ is much greater than the characteristic decay time τ_A, then

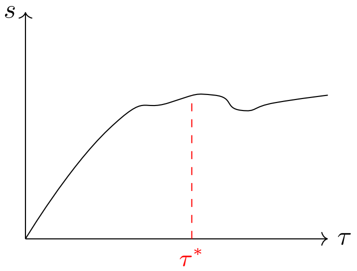


Figure 20: When τ is small s increases with τ, indicating that the block size is too small and contain correlated data. When $\tau - s$ curve has levelled off, the block size is long enough to ensure independent block average.

we can estimate the variance using

$$\begin{aligned}\sigma^2(A_\tau) &\simeq \frac{1}{\tau}\int_{-\infty}^{\infty} \mathrm{d}t\, C_{AA}(t) \\ &\simeq \frac{2\tau_A}{\tau} C_{AA}(0)\,. \end{aligned} \tag{1.168}$$

The relative variance in A_τ is therefore

$$\frac{\sigma^2(\tau_A)}{\langle A\rangle^2} = \frac{2\tau_A}{\tau}\frac{\langle A^2\rangle - \langle A\rangle^2}{\langle A\rangle^2}\,. \tag{1.169}$$

This indicates that the variance in a measured quantity is inversely proportional to the number of uncorrelated measurements $M \sim \tau/\tau_A$. From this, we can derive a recipe for choosing an appropriate block length τ. We choose it such that the *statistical inefficiency*

$$s = \frac{\tau\sigma^2(A_\tau)}{\langle A\rangle^2}\,, \tag{1.170}$$

which measures how much the variance of our time-averaged observable is “inflated” due to time correlation, becomes independent of τ.

1.9.2 System Size Dependence

Let us assume that an observable can be decomposed onto uncorrelated single-particle contributions:

$$\langle A\rangle = \sum_{i=1}^{N} \langle a_i\rangle = N\langle a\rangle\ , \tag{1.171}$$

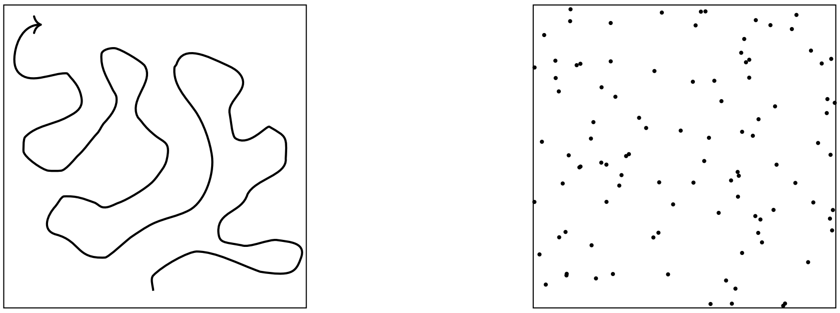
then

$$\langle A^2\rangle - \langle A\rangle^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} \langle [a_i - \langle a\rangle][a_j - \langle a\rangle]\rangle\ . \tag{1.172}$$

As the fluctuations in a_i and a_j are uncorrelated for $i \neq j$, we find that

$$\frac{\langle A^2\rangle - \langle A\rangle^2}{\langle A\rangle^2} = \frac{1}{N}\frac{\langle a^2\rangle - \langle a\rangle^2}{\langle a\rangle^2}\,. \tag{1.173}$$

The statistical error in such an additive property decreases as we increase the system size. However, no such scaling can be expected when computing truly collective (not decomposable into uncorrelated single-particle contributions) properties.



(a) Molecular dynamics

(b) Monte Carlo

Figure 21: The difference between molecular dynamics and Monte Carlo.

2 Monte Carlo

2.1 Monte Carlo and Inferential Statistics

Monte Carlo (MC) is a method of estimating the value of a quantity using the principles of inferential statistics. In contrast to molecular dynamics, which is a deterministic method that aims to follow the actual time evolution of a system as close as possible to evaluate some quantities, Monte Carlo is *stochastic*. It explores the states of the system by doing random sampling, and evaluates the desired quantity from the samples. In an MD simulation, the evolution of the system from one time step to the next has a solid physical meaning, but this does not have to be the case for an MC simulation.

2.1.1 Inferential Statistics

A simple example. If you toss a coin 100 times and they all land heads. What do you think you will get if you flip it next time? We have done a fairly large number of experiments, so based on our sample, we would suspect that this coin is unfair, and we would still get a head next time. Now if you toss another coin for another 100 times, and you get 52 heads and 48 tails. The best estimate of the probability of getting a head for our next toss would be 52% based on our samples.

Law 2.1 (Strong law of large numbers (Kolmogorov's law)). Given a collection of independent and identically distributed (iid) samples from a random variable with finite mean, the sample average converges almost surely to the expected value.

$$\text{Prob}\left(\lim_{n\to\infty} \overline{X}_n = \mu\right) = 1\,. \tag{2.1}$$

What this means is that, as the number of trials n goes to infinity, the probability that the average of the observations converges to the expected value, is equal to one.[5]

You might be worrying about some extreme events, like you are tossing a fair coin, but you get all heads for the first 100 tosses. But as you do more tosses, say 10,000 more tosses, you are likely to get around 5,000 heads with 5,000 tosses, and the extreme result of the first 100 tosses will be washed

[5] There is also a weak law of large numbers.

Law 2.2 (Weak law of large numbers (Khinchin's law)). The sample mean converges in probability to the expected value. That is, for any $\epsilon > 0$,

$$\lim_{n\to\infty} \text{Prob}(|\overline{X}_n - \mu| < \epsilon) = 1\,. \tag{2.2}$$

This is not a maths course so we don't care about the difference here.

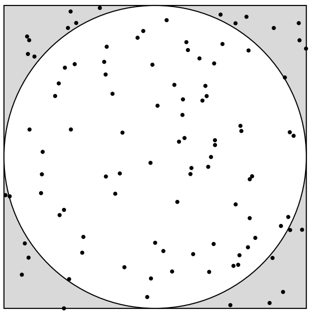


Figure 22: Randomly generating $N = 100$ points in a square.

away. This is the *regression to the mean.* Following an extreme event, the next random event will likely be less extreme.

2.1.2 Monte Carlo

Let's see an example of Monte Carlo in its simplest form. Consider a square of side length 2, and its inscribed circle. The area of the square is 4 and the area of the circle is π. This means that if I randomly choose a point in the square, it has a chance

$$p = \frac{A_{\text{circle}}}{A_{\text{square}}} = \frac{\pi}{4} \tag{2.3}$$

to fall inside the circle. Now if I am asking my computer to generate a random set of N points for some large enough N, then the fraction of points that will fall inside the circle will be

$$\lim_{N\to\infty} \frac{N_{\text{in}}}{N} = p = \frac{\pi}{4}\,. \tag{2.4}$$

This means that by counting how many points are inside a circle, we can estimate the value of π by[6]

$$\pi \simeq 4p = \frac{4N_{\text{in}}}{N}\,. \tag{2.5}$$

We are essentially evaluating a 2D integral by the stochastic method.

2.2 Importance Sampling

As you can see in the above example, randomly sampling points from a uniform distribution may not be the most efficient way to perform an MC calculation.

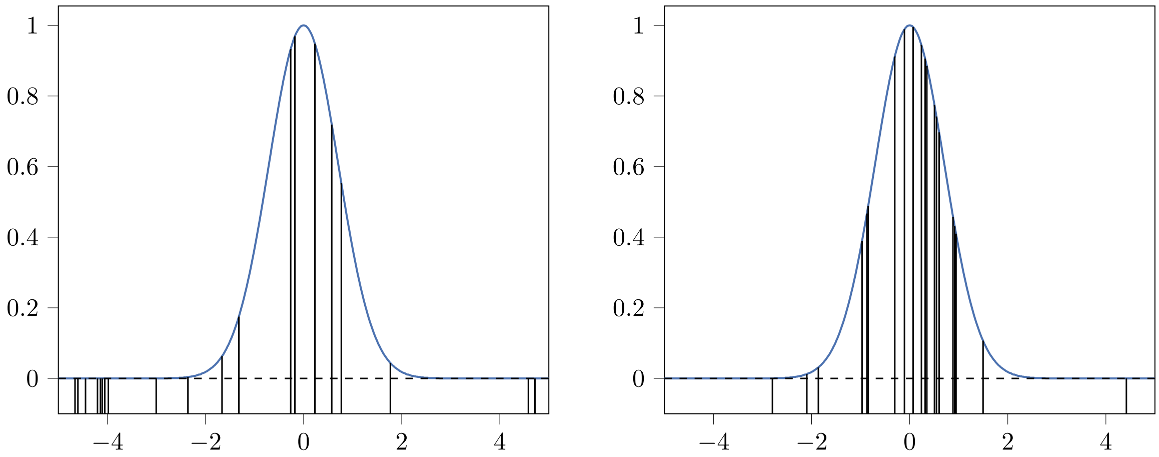
Suppose we are evaluating some integral

$$\int_a^b \mathrm{d}x\, f(x)\,. \tag{2.6}$$

If we want to do this using standard Monte Carlo, then we would sample points from a to b uniformly. However, if we are doing something like

$$\int_{-5}^{5} \mathrm{d}x\, \mathrm{e}^{-x^2}\,, \tag{2.7}$$

which we know that most of the weight of the integral actually comes from a small range of x where $f(x)$ is large. Sampling more often in that region should greatly increase the accuracy of the MC integration. However, this implies that we know something about $f(x)$ already.



(a) Sampling with uniform distribution.

(b) Sampling with biased distribution

Figure 23: Monte Carlo method to evaluate the integral $\int_{-5}^{5} \mathrm{d}x\, \mathrm{e}^{-x^2}$ using uniform and biased sampling.

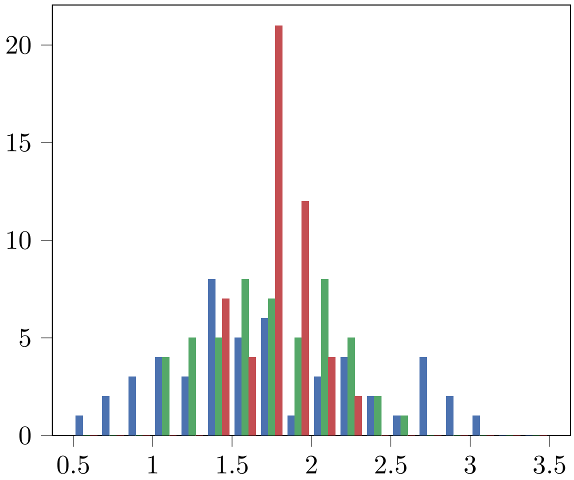


Figure 24: The histogram of 50 attempts to evaluate the integral $\int_{-5}^{5} \mathrm{d}x\, \mathrm{e}^{-x^2} = \sqrt{\pi}\,\mathrm{erf}(5) \approx 1.77245$ using uniform (blue), biased (green) and an even more biased (red) sampling. The uniform sampling gives 1.73 ± 0.64, the biased sampling gives 1.75 ± 0.39 and the more biased sampling gives 1.78 ± 0.19. All three sampling methods give correct results on average due to a fairly large number of attempts (law of large numbers), but a more biased sampling gives a more consistent result.

To do this, we sample from a biased probability distribution $w(x)$, which is derived from the best guess of what the function $f(x)$ will look like. An estimate of the integral can then be written as

$$I = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_{i|w})}{w(x_{i|w})}\,, \tag{2.8}$$

where $x_{i|w}$ means that x_i is drawn from the probability distribution $w(x)$, $x_i \sim w(x)$. Here the division by $w(x_{i|w})$ is to compensate for biasing the distribution. When $w(x_{i|w})$ is a uniform distribution, we recover the uniform sampling.

[6] I tried for $N = 1,000,000,000$ and it gives me an estimate of $\pi \approx 3.141633$ in 7 minutes. Apparently this is not the most efficient way of calculating π

The variance of our integral estimation is

$$\sigma^2_{I|w} = \frac{1}{N}\left\langle \frac{1}{N}\sum_{i=1}^{N}\left(\frac{f(x_{i|w})}{w(x_{i|w})} - \left\langle \frac{f(x_{i|w})}{w(x_{i|w})}\right\rangle\right)^2\right\rangle = \frac{1}{N}\left\langle \sigma^2_{f|w}\right\rangle . \tag{2.9}$$

The error still scales as $N^{-1/2}$ but the prefactor is different for different sampling. The best prefactor would result from using

$$w(x) = \frac{1}{\mathcal{I}} f(x) , \tag{2.10}$$

where $\mathcal{I} = \int_a^b \mathrm{d}x\, f(x)$ is for normalisation, for which $\sigma_{f|w} = 0$. In this case our estimation would always be exact, but this requires us to know the integral *a priori*. Hence, in practice, we can only have a good estimate of what a suitable $w(x)$ might be. Choosing a distribution that gives a good estimate for $w(x)$ is known as *importance sampling*. This is a crucial step in MC calculation. When sampling highly non-uniform functions, brute force MC with uniform sampling can result in a very large pre-factor for $\sigma_{I|w}$.

2.2.1 Importance Sampling in Statistical Mechanics

Importance sampling is extremely crucial in statistical mechanical systems since it involves high dimensional integrals with significant contributions from only a very small fraction of the total possible sampling spaces.

For example, suppose we want to evaluate the ensemble average of a physical quantity A

$$\langle A\rangle = \sum_{|n\rangle} A_n p_n = \frac{1}{Q}\sum_{|n\rangle} A_n \mathrm{e}^{-\beta E_n} , \tag{2.11}$$

where A_n is the value of the quantity A for the microstate $|n\rangle$, and the sum is taken over all possible microstates. To evaluate this using Monte Carlo, the simplest way is to choose M states completely randomly and take the average with the Boltzmann factor

$$A_M = \frac{\sum_{|n\rangle}^{M} A_n \mathrm{e}^{-\beta E_n}}{\sum_{|n\rangle}^{M} \mathrm{e}^{-\beta E_n}} , \tag{2.12}$$

which is analogous to the random uniform sampling we discussed in the context of integrals. In theory, as $M \to \infty$ (or to the number of all possible microstates if this is finite), then $A_M \to \langle A\rangle$. But in practice there are two major problems.

1. The number of states in a statistical mechanical system usually grows exponentially with system size.

   For example, consider an $N \times N$ 2D Ising lattice, where each lattice point can sit in $S_i = \pm 1$. The total number of states is 2^{N^2}. Even a 5×5 lattice will have $2^{25} = 33,554,432$ distinct states. Doubling each side to a 10×10 lattice, this number will grow to 10^{30}. The number will soon get astronomically large, and it will get extremely difficult to sample a proper fraction of it.

2. Averages are typically dominated by a very small fraction of the states, which random uniform sampling is very unlikely to find.

   Consider a hard-sphere liquid. If you locate the particles randomly, then you will easily get particles overlapping with each other, even at moderate particle density. Such situation gives infinite energy and so contribute nothing to the average $\langle A\rangle$. For example, if we simulate 100 hard-spheres near the freezing transition, then only one in about 10^{260} random configurations would count towards the average.

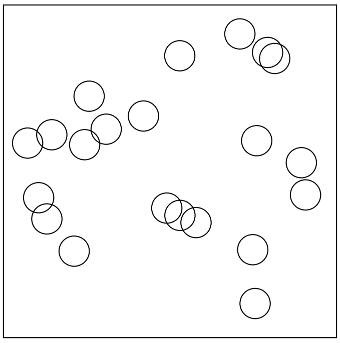


Figure 25: Randomly generating particles in a cell. Even at moderate density, there is an extremely high chance of resulting in some overlap.

Overcoming these challenges clearly would involve some form of importance sampling. Since Boltzmann distribution determines the weight of each state in the ensemble average, it is a natural choice of function from which we sample our points. Applying the Boltzmann weighting function, we get

$$A_m = \frac{\sum_{|n\rangle}^{M} A_{n|p} \mathrm{e}^{-\beta E_n} (\frac{1}{\mathrm{e}^{-\beta E_n}})}{\sum_{|n\rangle}^{M} \mathrm{e}^{-\beta E_n} (\frac{1}{\mathrm{e}^{-\beta E_n}})} = \frac{\sum_{|n\rangle}^{M} A_{n|p}}{M} \, , \tag{2.13}$$

where the subscript $n \mid p$ means that we are sampling microstates $|n\rangle$ with probability given by the Boltzmann distribution $p_n = \mathrm{e}^{-\beta E_n}/Q$.

However, this method also has two clear difficulties.

1. Computing the partition function is almost always impossible.
2. It is hard to know how to find the most probable states.

The Metropolis algorithm provides a clever way around these problems.

2.3 Metropolis Sampling and Detailed Balance

The Metropolis algorithm performs a biased random walk through the configuration space as follows

1. Start with a given configuration S_i.
2. In the next step, the system will have a probability $W_{ij} = W(S_i \to S_j)$ to transfer into a new configuration S_j. The transition probability will be derived later.
3. Repeat to create a trajectory through the phase space.
4. Evaluate the physical quantity of interest via

$$A_m = \frac{\sum_{i}^{M} A_{i|p}}{M} \, . \tag{2.14}$$

2.3.1 Markov Chain

Mathematically, the above random walk process is described by a *Markov process.*

Definition 2.3. A *Markov process* is a stochastic process that satisfies the *Markov property*: the future of the system is independent of the past, and is only dependent on the current state of the system.

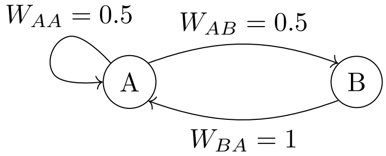


Figure 26: An example of a Markov chain. At each time step, the system moves from one state to another with the given transition probability.

A *Markov chain* is a type of Markov process that has discrete time.

Suppose we have a stochastic process with a set of possible states $S_1, S_2, S_3, \ldots$ (finite, countably infinite or continuum) and discrete time $t_1, t_2, \ldots$. We denote the state of the system at time t_n as X_{t_n}. Then for the process to be Markov, the conditional probability that the system will be in state $X_{t_{n+1}} = S_j$ is only dependent on the current state of the system $X_{t_n} = S_i$. It is independent of $X_{t_{n-1}}, X_{t_{n-2}}, \ldots$ or the value of n. Hence, we can write the transition probability from state S_i to S_j

$$W_{ij} = W(S_i \to S_j) = \mathrm{Prob}(X_{t_{n+1}} = S_j \mid X_{t_n} = S_i)\,. \tag{2.15}$$

As with usual probabilities, we require $W_{ij} \geq 0$ and $\sum_j W_{ij} = 1$. Since W can be considered as a matrix, it is also known as the *transition matrix*.

The total probability that at time t_n the system is in state S_j is

$$\mathrm{Prob}(X_{t_n} = S_j) = \sum_i W_{ij}\,\mathrm{Prob}(X_{t_{n-1}} = S_i)\,. \tag{2.16}$$

If we treat time as continuous, $\mathrm{Prob}(X_t = S_j) = P(S_j, t)$ and consider that the probability is conserved at all times: $\sum_j P(S_j, t) = 1$, then we get

$$\frac{\mathrm{d}P(S_j, t)}{\mathrm{d}t} = \sum_i [W_{ij}P(S_i, t) - W_{ji}P(S_j, t)]\,. \tag{2.17}$$

The change in the probability of the system being in state S_j is the probability of moving into state S_j minus the probability of moving away from state j.

2.3.2 Equilibrium and Detailed Balance

We say a system is in *equilibrium* if the average macroscopic properties of the system have stopped changing, except for small fluctuations away from equilibrium that soon dissipate back towards equilibrium. The properties of a equilibrium system are independent of its history. Thus we can ignore how a system has reached equilibrium and focus on the distribution of microstates at equilibrium.

Suppose the system is at equilibrium such that

$$\frac{\mathrm{d}P(S_j, t)}{\mathrm{d}t} = 0\,, \tag{2.18}$$

this means that

$$\sum_i [W_{ij}P_i^{\mathrm{eq}} - W_{ji}P_j^{\mathrm{eq}}] = 0\,. \tag{2.19}$$

The probability of moving out from S_j is equal to the probability of moving into S_j. A sufficient but not necessary condition for this to happen is

$$W_{ij}P_i^{\mathrm{eq}} = W_{ji}P_j^{\mathrm{eq}} \tag{2.20}$$

for all i, j. This is known as the *detailed balance.* In such case, the average number of moves from any state i to state j is equal to the number of moves from j to i.

In Metropolis Monte Carlo, we want the probability that the system sits in the microstate S_j is equal to its Boltzmann factor: $P_j \propto \mathrm{e}^{-\beta E_j}$. Then once the equilibrium has established, an ensemble of random walkers will populate the states S_j with the Boltzmann distribution that we desired. To achieve this, all we need to do is to choose the correct transition probabilities such that

$$\frac{W_{ij}}{W_{ji}} = \frac{P_j^{\mathrm{eq}}}{P_i^{\mathrm{eq}}} = \mathrm{e}^{-\beta(E_j - E_i)} . \tag{2.21}$$

The trick to determine the transition probability W_{ij} is to split it into two steps.

1. Choose a new configuration S_j with a selection probability α_{ij}.
2. Accept or reject this new configuration with an acceptance probability $P_{\mathrm{acc},ij}$. If accepted, move to S_j, and if not, stay in S_i.

Then the transition probability can be written as

$$W_{ij} = \alpha_{ij} P_{\mathrm{acc},ij} . \tag{2.22}$$

If we take α to be symmetric, $\alpha_{ij} = \alpha_{ji}$, then the condition of detailed balance becomes

$$\frac{P_{\mathrm{acc},ij}}{P_{\mathrm{acc},ji}} = \mathrm{e}^{-\beta \Delta E} , \tag{2.23}$$

where $\Delta E = E_j - E_i$. There are many choices of $P_{\mathrm{acc},ij}$ that satisfies this condition. The most popular choice is the *algorithm of Metropolis*

$$P_{\mathrm{acc},ij} = \begin{cases} \mathrm{e}^{-\beta \Delta E} & \text{if } \Delta E > 0 \\ 1 & \text{otherwise} \end{cases} . \tag{2.24}$$

This means that if the energy decreases, you always accept the trial move, whereas if the energy increases, you accept it with a probability proportional to the Boltzmann factor of the energy difference. The clear advantage is now you don't need to evaluate the partition function.

Note that for a Monte Carlo simulation to sample points in configurational space according to the correct Boltzmann weight, the detailed balance condition is sufficient but not necessary. There might exist correct sampling schemes that violate the detailed balance. However, unless we can prove that a non-detailed-balanced scheme yields the correct distribution, there is no point of doing so in practice.

Example. Metropolis Monte Carlo for Interacting Fluid.

Consider an interacting fluid of N particles with positions $\mathbf{r}^N = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ and momenta $\mathbf{p}^N = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N)$ interacting through the potential $V(\mathbf{r}^N)$. When considering the energy E_i of a state i, it includes both the kinetic energy and the potential energy, but as seen before, we can always integrate over the trivial momenta parts in the partition function to obtain

$$Q = \frac{1}{N!\Lambda^{3N}} Z_N . \tag{2.25}$$

We can always ignore the momenta and do our MC averages over the states determined by $\mathbf{r}^N$ alone, with the probability distribution

$$P(\mathbf{r}^N) = \frac{1}{Z_N} \exp[-\beta V(\mathbf{r}^N)] . \tag{2.26}$$

To move from step k to step $k+1$ in the Monte Carlo trajectory:

1. Select a fluid particle j at random from the configuration of step $\mathbf{k}$:
$$\mathbf{r}_k^N = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_j, \ldots, \mathbf{r}_N)\,. \tag{2.27}$$
2. Move it to a new position with a random displacement $\mathbf{r}'_j = \mathbf{r}_j + \boldsymbol{\Delta}$, with $\Delta_i \in [-\delta, +\delta]$ for some δ.
3. Calculate the potential energy $V(\mathbf{r}^N_{\text{trial}})$ for the new state $\mathbf{r}^N_{\text{trial}} = (\mathbf{r}_1, \ldots, \mathbf{r}'_j, \ldots, \mathbf{r}_N)$.
4. Accept the move $\mathbf{r}_k^N \to \mathbf{r}^N_{\text{trial}}$ with a probability
$$P_{\text{acc}}(\mathbf{r}_k^N \to \mathbf{r}^N_{\text{trial}}) = \min\{1, \exp[-\beta(V(\mathbf{r}^N_{\text{trial}}) - V(\mathbf{r}_k^N))]\}\,. \tag{2.28}$$
If the move is accepted, $\mathbf{r}^N_{k+1} = \mathbf{r}^N_{\text{trial}}$. If not, $\mathbf{r}^N_{k+1} = \mathbf{r}^N_k$.

We need to have a suitable choice of our move step $\boldsymbol{\Delta}$. If it is too large, most trial moves will be rejected, and if $\boldsymbol{\Delta}$ is too small, you will move very slowly in the phase space, and it will take a large number of steps to reach equilibrium. For most systems, an average acceptance rate of $20\% - 30\%$ is suitable.

In each MC step, we only attempt to move one particle, and so N separate MC steps are roughly equal to a single MD step, where all particles are moved together. Why don't we attempt to move N particles in a single MC step? There is no significant difference in computational cost between N single moves and one move of N particles in terms of evaluating $V(\mathbf{r}^N_{\text{trial}})$, but the rejection probability will be larger if we move all particles together. If the average probability of rejection for moving one atom is p_{rej}, then the probability of an accepted move of N particles at once is $(1 - p_{\text{rej}})^N$, which tends to zero as N increases. To get any acceptance at all in a collective move, we would require really small values of $\boldsymbol{\Delta}$. Hence, for the same computational cost, single particle moves advance particles much faster than collective moves.

Example. The Ising model The Ising model has Hamiltonian
$$\mathcal{H} = -J \sum_{\langle i,j \rangle} S_i S_j - B \sum_i S_i\,, \tag{2.29}$$
where $S_i = \pm 1$ and $\langle i, j \rangle$ denotes sum over all neighbouring pairs of particles. As you have investigated in Part II B7: *Statistical Mechanics*, there are three competing effects. The neighbouring interactions want to keep everything aligned, the external field wants to keep everything to a specific direction, and the entropy wants to screw everything up as there are more disordered microstates than aligned microstates. At low temperatures, the ground state keeps (almost) all spins aligned with the external field, since then both the neighbouring alignment and the external field terms in the Hamiltonian will be simultaneously minimised. Defining the magnetisation
$$M = \frac{1}{N^2} \sum_i S_i\,, \tag{2.30}$$
we can see that at low temperatures, $M \to 1$ for $B > 0$ and $M \to -1$ for $B < 0$.

To do this in Monte Carlo, we can use the following scheme:

1. Start From an arbitrary initial configuration of spins.
2. At step k, select a random spin $S_j(k)$ and flip it to $-S_j(k)$.
3. Evaluate the energy change
$$\Delta E = 2S_j(k) \left[J \sum_{\text{neighbouring } n} S_n(k) + B \right]\,. \tag{2.31}$$

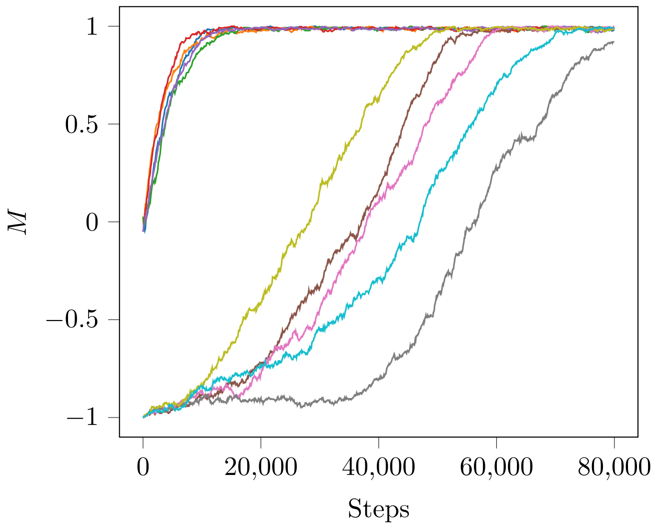


Figure 27: Metropolis Monte Carlo of a 30 × 30 2D square lattice Ising Model with $J = 1$, $\beta = 0.6$ and $B = 0.4$. Five runs are taken from random initial configurations, and five runs are taken from completely flipped configurations. It takes much longer for a flipped-spin system to get to equilibrium.

4. Accept the move $S_j(k) \to -S_j(k)$ with probability

$$P_{\text{acc}}(S_j(k) \to -S_j(k)) = \min\{1, \exp(-\beta\Delta E)\}\,. \tag{2.32}$$

   If accepted, $S_j(k+1) = -S_j(k)$, and if not, $S_j(k+1) = S_j(k)$.

2.4 Hard-to-Simulate Systems

Our above discussions are based on the assumption that our system is ergodic, so that it can visit all its states (or sufficiently close to all of its states if it is a continuum) in a finite number of steps. Due to the finite time scale and length scale nature of simulations, it is usually impossible to fully sample the available microstates — sometimes not even close to sampling a large proportion of its microstates. There are *quasi-ergodic systems* with frustrated or rugged energy landscapes (many deep minima separated by high energy barriers) such that if you are stuck in one of the minima, it is impossible to get out and sample other places. Examples of this type of system include

- Systems with strong electrostatic interactions.
- Fluids with strongly orientation-dependent interactions (e.g. dipole, hydrogen bonds).
- Self-organising systems.
- Supercooled liquids.

One apparent solution to this ergodicity problem is to use brute force. If we run a large enough number of simulations from many different random initial conformations, then we would ultimately sample all over the phase space. However, we will introduce two other clever solutions that help us solve the ergodicity problem.

1. Add a biasing potential and sample from a non-Boltzmann distribution so that the unfavorable states are sampled adequately.

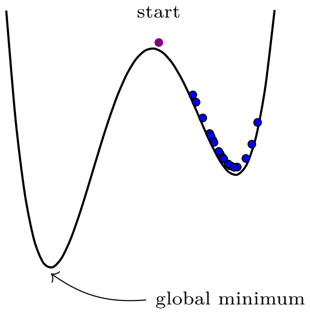


Figure 28: The system might appear to equilibrate inside the local well, even though our simulation is not sampling the phase space correctly. This is the *ergodicity problem*: we cannot get everywhere in the phase space within the finite amount of simulation time.

2. Enhance sampling by changing the kinetic energy of particles.

Both methods help us climb over the potential barriers between minima.

2.4.1 Biased Sampling Methods

The standard (unbiased) canonical distribution is

$$P(i) \propto \exp[-\beta V(\mathbf{r}_i^N)] \,. \tag{2.33}$$

In the biased ensemble, we introduce an additional weighting potential U for the microstates in the exponential

$$P_{\text{biased}}(i) \propto \exp[-\beta(V(\mathbf{r}_i^N) + U(\mathbf{r}_i^N))] \,. \tag{2.34}$$

It modifies the canonical distribution so that some configurations can have higher or lower probabilities of being visited than they would normally do. If we compare the weighted and the unweighted probabilities, we get

$$\frac{P(i)}{P_{\text{biased}}(i)} = \exp[\beta U(\mathbf{r}_i^N)] \,. \tag{2.35}$$

Using the biased property, we can follow our standard procedure in a Metropolis Monte Carlo simulation. We split the total transition probability into

$$W_{ij} = \alpha_{ij} P_{\text{acc},ij} \,, \tag{2.36}$$

and make the selection probability symmetric

$$\alpha_{ij} = \alpha_{ji} \,. \tag{2.37}$$

Then by the detailed balance, we have

$$P_{\text{acc},ij} = \min\{1, P_{\text{biased}}(j)/P_{\text{biased}}(i)\} = \min\{1, \exp[-\beta(\Delta V + \Delta U)]\} \,. \tag{2.38}$$

Then to recover the ensemble average of a physical quantity in the standard canonical ensemble, we reweight the sampled microstates and get

$$\langle A \rangle = \frac{\sum_i A_i P(i)/P_{\text{biased}}(i)}{\sum_i P(i)/P_{\text{biased}}(i)} = \frac{\sum_i A_i \exp[\beta U(\mathbf{r}_i^N)]}{\sum_i \exp[\beta U(\mathbf{r}_i^N)]} \,. \tag{2.39}$$

Since the introduction of biasing is to avoid ergodicity issues, we often choose $U(\mathbf{r}_i^N)$ such that the simulation explores a wide range of microstates to accumulate a good statistics. We can think of them as artificially reshaping the free-energy landscape to improve sampling in specific regions. We will briefly introduce two common biasing methods: metadynamics and umbrella sampling.

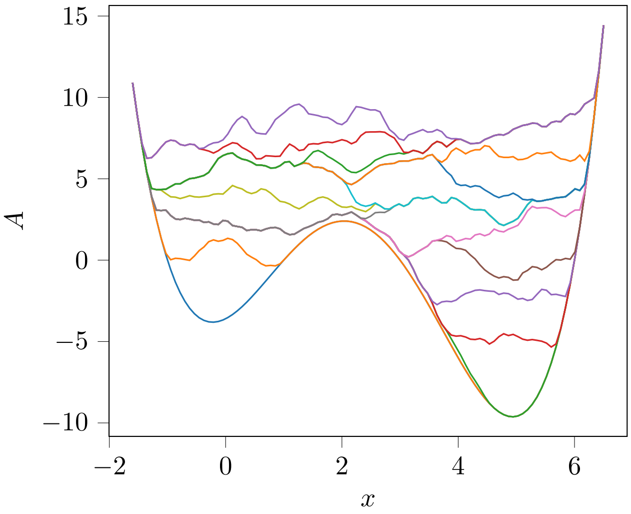


Figure 29: Energy landscapes of the system against Monte Carlo steps (1,000 steps between consecutive plots) in a metadynamics simulation.

Metadynamics

Metadynamics tackles difficult-to-simulate systems by applying a history-dependent biasing potential that discourages the system from revisiting previously explored states.

1. A few low-dimensional reaction coordinates are chosen as *collective variables* (CVs), ξ to describe the slow degrees of freedom which we plot our energy landscapes against.
2. During the simulation, a small Gaussian potential is added to the potential energy of the system at the current position of the system in the CV space. This discourages the system from returning to the same region, pushing it into new unexplored states.
3. Over time, the biased potential will "fill in" the local minima, flattening the energy landscape. The system is then driven over the energy barrier and is dropped into another minimum.
4. The accumulated bias potential is related to the negative of the free energy landscape, allowing one to reconstruct the free-energy profile of the system.

The biased potential at time t is given by

$$U(\xi, t) = \sum_{t'<t} w e^{-\frac{(\xi-\xi(t'))^2}{2\sigma^2}}\,, \tag{2.40}$$

where w and σ are the height and width of the Gaussian bias potential added at each time step.

The free energy of a state is related to its probability by

$$A(\xi) = -k_B T \ln P(\xi)\,. \tag{2.41}$$

In metadynamics,

$$A_{\text{biased}}(\xi, t) = A_{\text{unbiased}}(\xi) + U(\xi, t)\,. \tag{2.42}$$

In the limit of long simulation time, the landscape is completely filled, making all states equiprobable, and so

$$\lim_{t\to\infty} \ln P_{\text{biased}}(\xi, t) = -\beta \left[A_{\text{unbiased}}(\xi) + \lim_{t\to\infty} U(\xi, t)\right] = \text{const.} \tag{2.43}$$

The constant shift in the free energy is unimportant. Therefore we can take

$$A_{\text{unbiased}}(\xi) = -\lim_{t\to\infty} U(\xi, t)\,. \tag{2.44}$$

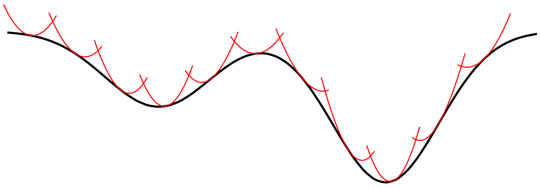


Figure 30: Umbrella sampling adds a local harmonic biasing potential in each run.

Various other quantities can then be derived, for example

$$S(\xi) = -\frac{\partial A(\xi)}{\partial T} \tag{2.45}$$

$$E(\xi) = A(\xi) + TS(\xi) \tag{2.46}$$

$$K = \exp\left(-\frac{\Delta A}{k_B T}\right) \tag{2.47}$$

$$k = A \exp\left(-\frac{\Delta A^{\ddagger}}{k_B T}\right) . \tag{2.48}$$

Since the unbiased probability distribution is

$$P_{\text{unbiased}} = P_{\text{biased}}(\xi) \mathrm{e}^{\beta U(\xi)} , \tag{2.49}$$

for any physical quantity A, the unbiased ensemble average can be computed as

$$\langle A \rangle = \frac{\sum_i A(\mathbf{r}_i) \mathrm{e}^{\beta U(\xi_i)}}{\sum_i \mathrm{e}^{\beta U(\xi_i)}} . \tag{2.50}$$

Umbrella Sampling

In contrast to metadynamics, in umbrella sampling, we need to know the states A and B we want to connect, and then we define a *reaction coordinate* ξ to connect these two states. We proceed as follows:

1. We perform J simulations of the same system.
2. In each simulation, we restrain the system to sample a small range of the reaction coordinate ξ centred around ξ_j. This is done by adding a bias potential $U_j(\xi)$, which can for example be a harmonic potential $U_j(\xi) = \frac{1}{2}k(\xi - \xi_j)^2$.
3. Use different target value ξ_j for each simulation such that it spans the entire range of interest.
4. Measure the weighted ensemble distribution $P_{\text{biased}}(\xi)$ for each simulation. Unweight and stitch together all $P_{\text{biased}}(\xi)$ to produce an unweighted underlying free function.

Each simulation is connected to the unweighted distribution by

$$P(\mathbf{r}^N) \propto P_{\text{biased},j}(\mathbf{r}^N) \exp[\beta U_j(\xi)] . \tag{2.51}$$

By integrating all other coordinates except the reaction coordinate of interest, one can write

$$P(\xi) \propto P_{\text{biased},j}(\xi) \exp[\beta U_j(\xi)] . \tag{2.52}$$

Taking the logarithm gives

$$A(\xi) = -k_B T \ln[P_{\text{biased},j}(\xi)] - U_j(\xi) + \text{const.} \tag{2.53}$$

We can adjust the constant to glue different segments of $A(\xi)$ together.

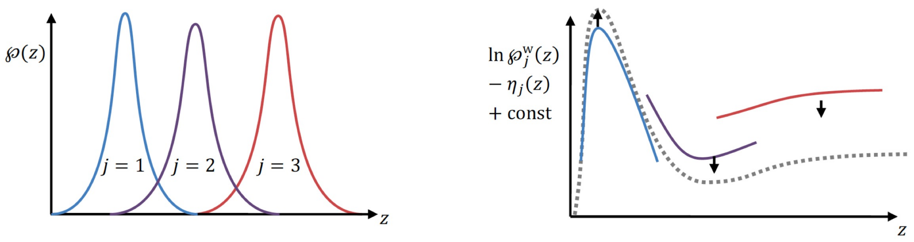


Figure 31: Using umbrella sampling to construct the free energy landscape.

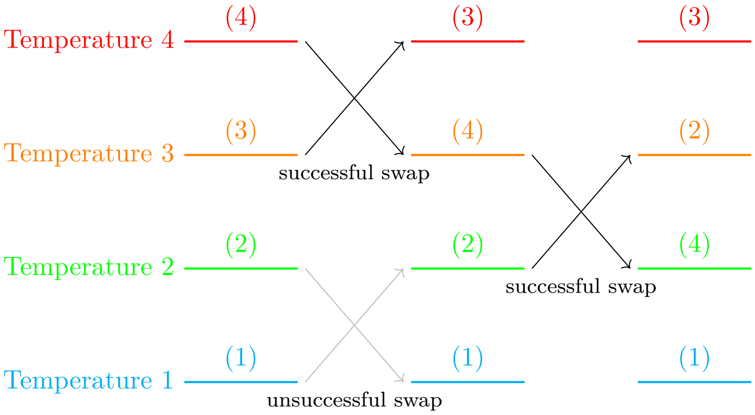


Figure 32: Temperature replica exchange molecular dynamics.

2.4.2 Temperature Replica Exchange Molecular Dynamics

The next method uses high temperature to avoid being trapped in a local minimum. It is known as *temperature replica exchange molecular dynamics* (T-REMD) (or the corresponding Monte Carlo). Suppose we want to investigate the system at temperature T_1.

1. We create J replicas of the same system, and perform them simultaneously at different temperature T_j with $T_1 < T_2 < \cdots < T_J$.
2. Each simulation is evolved independently, either through MD or MC.
3. At set intervals, *replica swap MC moves* are performed between adjacent replicas. In a swap move, the instantaneous configurations are exchanged between the two temperatures.

The replica exchanging process can also be modelled by a Markov chain in the entire J-system ensemble. Now a microstate of this ensemble is a list of all positions in all of the replicas

$$\mathbf{R} = (\mathbf{r}_1^N, \mathbf{r}_2^N, \ldots, \mathbf{r}_J^N)\,. \tag{2.54}$$

Since the replicas do not interact, we have

$$P(\mathbf{R}) = \prod_j P_j(\mathbf{r}_j^N)\,. \tag{2.55}$$

Using canonical probabilities in each replica,

$$P(\mathbf{R}) = \prod_j \frac{\exp[-\beta_j V(\mathbf{r}_j^N)]}{Z_j}\,. \tag{2.56}$$

Consider a swap move between configurations at two adjacent temperatures 1 and 2. This would exchange the coordinates in replica 1 $\mathbf{r}_1^N$ with the coordinate of replica 2 $\mathbf{r}_2^N$, so that the coordinates after exchange are

$$\mathbf{R}' = (\mathbf{r}_2^N, \mathbf{r}_1^N, \ldots, \mathbf{r}_J^N)\,. \tag{2.57}$$

We can again use Metropolis algorithm to decide whether or not to accept this swap, by

$$P_{\mathrm{acc}}(\mathbf{R} \to \mathbf{R}') = \min\{1, P(\mathbf{R}')/P(\mathbf{R})\}\,, \tag{2.58}$$

where

$$P(\mathbf{R}) = \frac{\exp[-\beta_1 V(\mathbf{r}_1^N)]}{Z_1} \times \frac{\exp[-\beta_2 V(\mathbf{r}_2^N)]}{Z_2} \prod_{j=3}^{J} \frac{\exp[-\beta_j V(\mathbf{r}_j^N)]}{Z_j} \tag{2.59}$$

$$P(\mathbf{R}') = \frac{\exp[-\beta_1 V(\mathbf{r}_2^N)]}{Z_1} \times \frac{\exp[-\beta_2 V(\mathbf{r}_1^N)]}{Z_2} \prod_{j=3}^{J} \frac{\exp[-\beta_j V(\mathbf{r}_j^N)]}{Z_j}\,. \tag{2.60}$$

Writing $\Delta\beta = \beta_2 - \beta_1$ and $\Delta V = V(\mathbf{r}_2^N) - V(\mathbf{r}_1^N)$, we have

$$P_{\mathrm{acc},ij}(\mathbf{R} \to \mathbf{R}') = \min\{1, \exp[\Delta\beta \Delta V]\}\,. \tag{2.61}$$

For $T_2 > T_1$, we would expect $V(\mathbf{r}_2^N) > V(\mathbf{r}_1^N)$, so the acceptance probability is usually small. Hence, we need to set the temperatures of the replica close enough with each other in order to achieve a good rate of accepted swaps.

2.5 Thermodynamic Integration

Suppose we want to compute the difference in binding free energy between two ligands L_1 and L_2 with the same protein

$$\Delta\Delta A = \Delta A_1 - \Delta A_2\,. \tag{2.62}$$

To compute this, we consider an *alchemical (unphysical) path* that describes the transformation of ligands L_1 into L_2, connected through intermediate states. We define this path by introducing a coupling parameter λ, such that $\lambda = 0$ corresponds to ligand L_1 (initial state) and $\lambda = 1$ corresponds to ligand L_2 (final state). The energy of the system can then be defined as

$$V(\lambda, \mathbf{r}^N) = (1-\lambda)V_1(\mathbf{r}^N) + \lambda V_2(\mathbf{r}^N)\,. \tag{2.63}$$

The partition function for an arbitrary λ is given by

$$Q(N,V,T;\lambda) = \frac{1}{\Lambda^{3N} N!} \int \mathrm{d}^{3N}\mathbf{r}^N \, \exp[-\beta V(\lambda, \mathbf{r}^N)]\,. \tag{2.64}$$

Taking the derivative of the free energy with respect to λ gives

$$\begin{aligned} \left(\frac{\partial A}{\partial \lambda}\right)_{N,V,T} &= -\beta^{-1} \frac{\partial \ln Q}{\partial \lambda} \\ &= \frac{\int \mathrm{d}^{3N}\mathbf{r}^N \, \left(\frac{\partial V(\lambda,\mathbf{r}^N)}{\partial \lambda}\right) \exp[-\beta V(\lambda, \mathbf{r}^N)]}{\int \mathrm{d}^{3N}\mathbf{r}^N \, \exp[-\beta V(\lambda, \mathbf{r}^N)]} \\ &= \left\langle \frac{\partial V(\lambda, \mathbf{r}^N)}{\partial \lambda} \right\rangle_\lambda\,. \end{aligned} \tag{2.65}$$

The average $\langle \ldots \rangle_\lambda$ can be viewed as an ensemble average over a system interacting with the potential $V(\lambda, \mathbf{r}^N)$. The free energy difference of the two states follows from a simple integration

$$\Delta A = A(\lambda = 1) - A(\lambda = 0) = \int_0^1 \mathrm{d}\lambda \, \left\langle \frac{\partial A}{\partial \lambda} \right\rangle_\lambda = \int_0^1 \mathrm{d}\lambda \, \left\langle \frac{\partial V(\lambda, \mathbf{r}^N)}{\partial \lambda} \right\rangle\,. \tag{2.66}$$

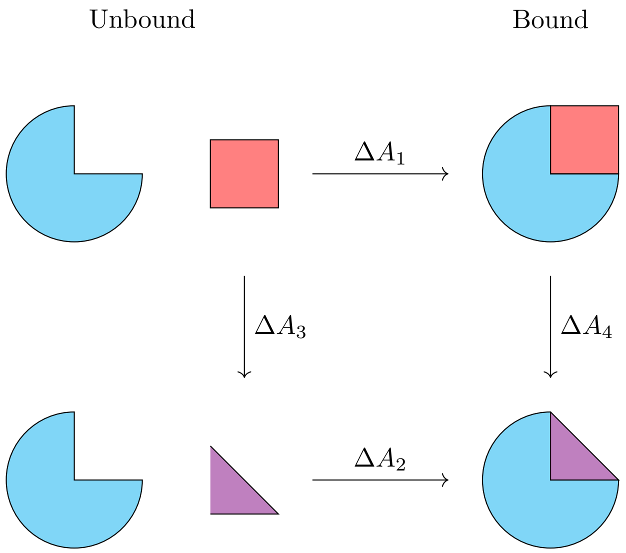


Figure 33: Thermodynamic cycle calculating the relative binding affinities of two ligands with the same protein substrate.

We can construct a thermodynamics cycle as shown in figure 33. We want to figure out

$$\Delta\Delta A = \Delta A_1 - \Delta A_2 \,, \tag{2.67}$$

but from the above alchemical paths, we can evaluate ΔA_3 and ΔA_4. It is easy to see that

$$\Delta A_1 - \Delta A_2 - \Delta A_3 + \Delta A_4 = 0 \,, \tag{2.68}$$

and so we can instead calculate

$$\Delta\Delta A = \Delta A_3 - \Delta A_4 \,. \tag{2.69}$$

Appendices

A Noether's Theorem

The simplest way of deriving Noether's theorem is to use the Lagrangian mechanics, which is another way of formulating classical mechanics. First let's be clear of our notations. For a system of N particles in d dimensions, we will rewrite the coordinates $\mathbf{r}_i$ as x^A, where $A = 1, \ldots, dN$. The Newton's equations are

$$\dot{p}_A = -\frac{\partial V}{\partial x^A}\,, \tag{A.1}$$

where $p_A = m_A \dot{x}^A$. To reduce the clutter in notations, when we write x^A in the argument of a function, we mean that it is a function of all x^A.

Lagrangian mechanics starts from defining the Lagrangian of a system.

Definition A.1. The *Lagrangian* for a system is defined by

$$L(x^A, \dot{x}^A) = T(\dot{x}^A) - V(x^A)\,, \tag{A.2}$$

where $T = \frac{1}{2}\sum_A m_A(\dot{x}^A)^2$ is the kinetic energy and $V(x^A)$ is the potential energy.

Note the weird minus sign between the kinetic and the potential energy. Despite this strange definition of the Lagrangian, it works really elegantly.

If we know that at $t = t_0$, the particles are at $x^A(t_0) = x_0^A$, and at $t = t_1$, the particles are at $x^A(t_1) = x_1^A$, there are infinite ways the systems can evolve with times between these two end points. How do we find the true paths $x^A(t)$ taken by the particles?

Theorem A.2 (Principle of Least Action). The actual path taken by the system is an extremum of the *action*, defined by

$$S[x^A(t)] = \int_{t_0}^{t_1} \mathrm{d}t\, L(x^A(t), \dot{x}^A(t))\,. \tag{A.3}$$

The S is an example of a *functional.* It maps functions to a number.

Proof. Consider varying a given path slightly, so

$$x^A(t) \longrightarrow x^A(t) + \delta x^A(t)\,, \tag{A.4}$$

where we fix the end points of the path by demanding $\delta x^A(t_0) = \delta x^A(t_1) = 0$. Then this results in a change in the action

$$\begin{aligned}\delta S &= \delta\left[\int_{t_0}^{t_1} \mathrm{d}t\, L\right] \\ &= \int_{t_0}^{t_1} \mathrm{d}t\, \delta L \\ &= \int_{t_0}^{t_1} \mathrm{d}t \sum_A \frac{\partial L}{\partial x^A}\delta x^A + \frac{\partial L}{\partial \dot{x}^A}\delta \dot{x}^A\,. \end{aligned} \tag{A.5}$$

We integrate the second term by parts to get

$$\delta S = \int_{t_0}^{t_1} \mathrm{d}t \sum_A \left[\frac{\partial L}{\partial x^A} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{x}^A}\right)\right]\delta x^A + \left[\frac{\partial L}{\partial \dot{x}^A}\delta x^A\right]_{t_0}^{t_1}\,. \tag{A.6}$$

The boundary term vanishes since we required $\delta x^A(t_0) = \delta x^A(t_1) = 0$. At an extremum of the action S, $\delta S = 0$ for all changes in the path $\delta x^A(t)$. This holds if and only if

$$\frac{\partial L}{\partial x^A} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{x}^A}\right) = 0\,. \tag{A.7}$$

for all A. These are known as the Euler–Lagrange equations. To finish the proof, we only need to show that Euler–Lagrange equations are equivalent to Newton's equations. From the definition of the Lagrangian, we have

$$\frac{\partial L}{\partial x^A} = -\frac{\partial V}{\partial x^A}\,, \tag{A.8}$$

while

$$\frac{\partial L}{\partial \dot{x}^A} = p_A\,. \tag{A.9}$$

Then it's easy to see that Newton's equations (A.1) are indeed equivalent to Euler–Lagrange equations (A.7). □

In fact Lagrangian mechanics is much more powerful than that. It turns out we can use any generalised coordinate we want (e.g. spherical, hyperbolic, or just some arbitrary parameters that uniquely define the configuration of the system), and we may add constraints to the coordinates, making it much more powerful than Newton's formulation of classical mechanics. Unfortunately, we can't go into too much detail here. If you are interested, see e.g. Prof. David Tong's notes on Classical Dynamics. But the important conclusion is that for any Lagrangian written in generalised coordinates $L(q_i, \dot{q}_i, t)$, the Euler–Lagrange equations still hold:

$$\frac{\partial L}{\partial q_i} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}_i}\right) = 0\,. \tag{A.10}$$

Definition A.3. Consider a one-parameter transformation of maps

$$q_i(t) \longrightarrow Q_i(s,t) \tag{A.11}$$

for $s \in \mathbb{R}$ such that $Q_i(0,t) = q_i(t)$. Then this transformation is said to be a *continuous symmetry* of the Lagrangian L if

$$\frac{\partial}{\partial s} L(Q_i(s,t), \dot{Q}_i(s,t), t) = 0\,. \tag{A.12}$$

Theorem A.4 (Noether's theorem). For each continuous symmetry, there is a conserved quantity.

Proof.

$$\frac{\partial L}{\partial s} = \sum_i \frac{\partial L}{\partial Q_i}\frac{\partial Q_i}{\partial s} + \frac{\partial L}{\partial \dot{Q}_i}\frac{\partial \dot{Q}_i}{\partial s}\,, \tag{A.13}$$

so we have

$$\begin{aligned} 0 = \left.\frac{\partial L}{\partial s}\right|_{s=0} &= \sum_i \frac{\partial L}{\partial q_i}\left.\frac{\partial Q_i}{\partial s}\right|_{s=0} + \frac{\partial L}{\partial \dot{q}_i}\left.\frac{\partial \dot{Q}_i}{\partial s}\right|_{s=0} \\ &= \sum_i \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}_i}\right)\left.\frac{\partial Q_i}{\partial s}\right|_{s=0} + \frac{\partial L}{\partial \dot{q}_i}\left.\frac{\partial \dot{Q}_i}{\partial s}\right|_{s=0} \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\sum_i \frac{\partial L}{\partial \dot{q}_i}\left.\frac{\partial Q_i}{\partial s}\right|_{s=0}\right)\,. \end{aligned} \tag{A.14}$$

The quantity

$$\sum_i \frac{\partial L}{\partial \dot{q}_i}\left.\frac{\partial Q_i}{\partial s}\right|_{s=0} \tag{A.15}$$

is constant for all time. □

Let's find some examples.

Example. Homogeneity of space.

Consider a system of N particles with Lagrangian

$$L = \frac{1}{2}\sum_i m_i \dot{\mathbf{r}}_i{}^2 - V(r_{ij})\,, \tag{A.16}$$

where $V(r_{ij})$ means that the potential is only dependent on the relative distances $r_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|$ between particles, not on their absolute positions. Then this Lagrangian has symmetry of translation: $\mathbf{r}_i \to \mathbf{r}_i + s\mathbf{n}$ for any vector $\mathbf{n}$ and real number s.

$$L(\mathbf{r}_i, \dot{\mathbf{r}}_i, t) = L(\mathbf{r}_i + s\mathbf{n}, \dot{\mathbf{r}}_i, t)\,. \tag{A.17}$$

Then by Noether's theorem, the conserved quantity is

$$\sum_i \frac{\partial L}{\partial \dot{\mathbf{r}}_i} \cdot \mathbf{n} = \sum_i \mathbf{p}_i \cdot \mathbf{n}\,. \tag{A.18}$$

The component of linear momentum in any direction is conserved, and hence

$$\sum_i \mathbf{p}_i \tag{A.19}$$

is also conserved.

Homogeneity in space $\implies$ translational invariance of L $\implies$ conservation of total linear momentum.

Example. Isotropy of Space.

The isotropy of space means that a closed system is invariant under rotations around an axis $\hat{\mathbf{n}}$, so all $\mathbf{r}_i$ are rotated to $\mathbf{r}'_i$ by the same amount. To work out the corresponding conserved quantity it suffices to work with the infinitesimal form of the rotations

$$\mathbf{r}_i \longrightarrow \mathbf{r}_i + \delta\mathbf{r}_i = \mathbf{r}_i + \alpha\hat{\mathbf{n}} \times \mathbf{r}_i\,, \tag{A.20}$$

where α is infinitesimal. To see that this is indeed a rotation, you can calculate the length of the vector and notice it is preserved to linear order in α. Then we have

$$L(\mathbf{r}_i, \dot{\mathbf{r}}_i) = L(\mathbf{r}_i + \alpha\hat{\mathbf{n}} \times \mathbf{r}_i, \dot{r}_i + \alpha\hat{\mathbf{n}} \times \dot{\mathbf{r}}_i)\,, \tag{A.21}$$

giving us the conserved quantity

$$\sum_i \frac{\partial L}{\partial \dot{\mathbf{r}}_i} \cdot (\hat{\mathbf{n}} \times \mathbf{r}_i) = \sum_i \hat{\mathbf{n}} \cdot (\mathbf{r}_i \times \mathbf{p}_i) = \hat{\mathbf{n}} \cdot \mathbf{L}\,. \tag{A.22}$$

This is the component of the total angular momentum in the direction $\hat{\mathbf{n}}$. Since $\hat{\mathbf{n}}$ is arbitrary, $\mathbf{L}$ is conserved.

Isotropy of space $\implies$ rotational invariance of L $\implies$ conservation of total angular momentum.