MSc in Theoretical Chemistry

Applications of Statistical Mechanics: Molecular Simulation Methods in Chemistry

Jonathan Doye

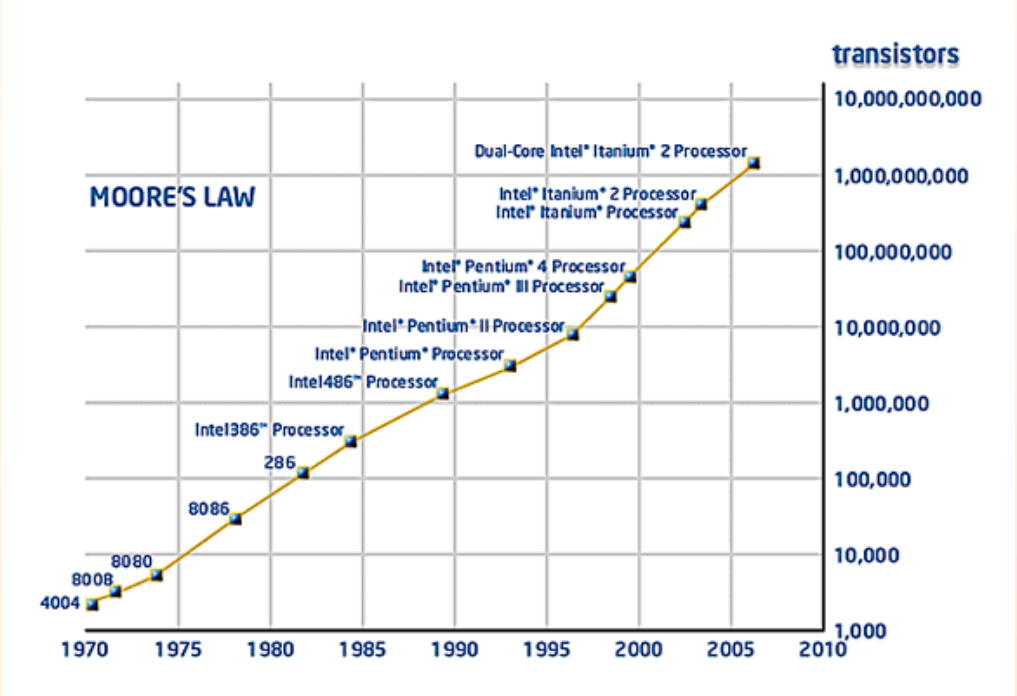
jonathan.doye@chem.ox.ac.uk

`http://physchem.ox.ac.uk/~doye/jon/teach/simulation.html`

Introduction

Why computer simulations?

- The inevitability of approximations in analytical work
  e.g. the motion of three interacting bodies.
- Moore's law and the increasing power of computers



- The increasing sophistication of computer simulation techniques.
- Wide-spread availability of user-friendly simulation packages.

What can molecular simulations achieve?

- Dynamics: $\{\mathbf{r}^N(t), \mathbf{p}^N(t)\}$. I.e. a movie
  n.b. $\mathbf{r}^N = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$
- Thermodynamics and equilibrium properties
  $\{\mathbf{r}_i^N\} \quad \Rightarrow \quad \langle Q \rangle$
  ensembles of configurations and averages

Roles of computer simulations

- Model systems
  - ▷ *As a test for theory.*
  - ▷ *Exploring complex systems.*
- Realistic models:
  - ▷ *Prediction of properties*
  - ▷ *Direct visualization of molecular configurations*

Limits of computer simulation

- Need the potential energy function $\mathcal{V}(\mathbf{r}^N)$ (GIGO principle)
- Size of system limited (max $O(10^8)$)
- Time scales relatively short (max $O(\mu s)$)

Computer simulation: a new science

1953: Metroplis *et al.*: First Monte Carlo simulation of a molecular system

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

1956 Alder and Wainwright: First molecular dynamics (MD) simulations.

1957 Alder and Wainwright: Hard-sphere debate

Phase Transition for a Hard Sphere System

B. J. Alder and T. E. Wainwright
University of California Radiation Laboratory, Livermore, California
(Received August 12, 1957)

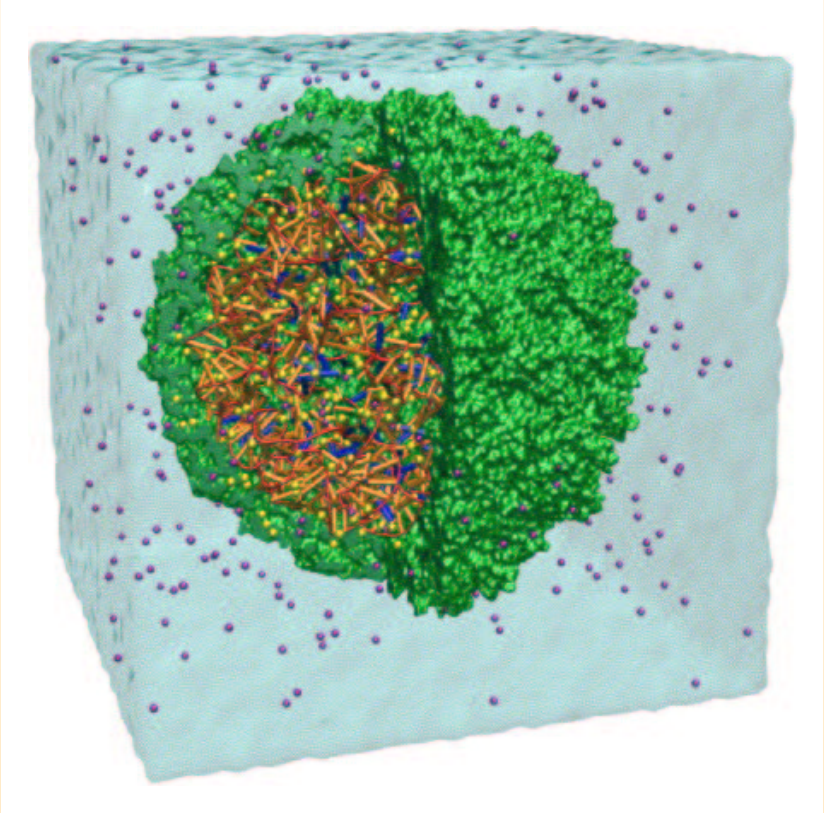
A CALCULATION of molecular dynamic motion has been designed principally to study the relaxations accompanying various nonequilibrium phenomena. The method consists of solving exactly (to the number of significant figures carried) the simultaneous classical equations of motion of several hundred particles by means of fast electronic computors. Some of the details as they relate to hard spheres and to particles having square well potentials of attraction have been described.[1,2] The method has been used also to calculate equilibrium properties, particularly the equation of state of hard spheres where differences with previous Monte Carlo[3] results appeared.

1964 Rahman: First MD with realistic interatomic potential (Ar)

1974 Rahman and Stillinger: First MD with realistic intermolecular potential (H_2O)

1977 McCammon *et al.*: First MD of a protein (BPTI) (500 atoms, 9.2 ps)

2006 Schulten group: MD of complete virus (STMV) (10^6 atoms, 50 ns)



Aims of this course

To introduce you to the principles behind computer simulation (not all the algorithmic details).

To put you in a position so that you know what you are doing when running someone else's code/a simulation package.

To put you in a position to start writing your own simulation codes.

This course won't cover:

- statistical mechanics basis
- how to program
- Where $\mathcal{V}(\mathbf{r}^N)$ comes from.

Recommended texts

Frenkel and Smit: Understanding Molecular Simulation: From Algorithms to Applications (Academic Press, 2002)

Allen and Tildesley: Computer Simulation of Liquids (Oxford, 1987)

Krauth: Algorithms and Computations (Oxford, 2006)

Newman and Barkema: Monte Carlo Methods in Statistical Physics (Oxford,1999)

Rapaport: The Art of Molecular Dynamics (Cambridge, 1995)

Press $et\ al.$: Numerical Recipes in ?: The Art of Scientific Computing, Cambridge

Molecular Dynamics

Molecular dynamics simulations are a method for computing the equilibrium and transport properties of a *classical* many-body system.

Classical: the nuclear motion of the constituent particles obeys the laws of classical mechanics, i.e. Newton's laws.

Classical mechanics is a good approximation for many materials. Quantum effects most relevant for light atoms and low temperatures.

Quantum nature of electrons enters through the potential, $\mathcal{V}(\mathbf{r}^N)$

Involves numerically solving Newton's equation by integrating them forward in time to generate a trajectory.

Basic scheme

1. Initialize the system: choose $\mathbf{r}^N(t=0)$ and $\mathbf{p}^N(t=0)$
2. Compute forces
3. Integrate equations of motion: from $t \longrightarrow t + \delta t$.
4. Repeat 3 and 4 until trajectory of desired length.

What ensemble?

For an isolated mechanical system the total energy is conserved during the motion. Proof:

$$\frac{dE}{dt} = \frac{d}{dt}\left[\sum_i^N \frac{1}{2} m_i \dot{\mathbf{r}}_i^2 + \mathcal{V}(\mathbf{r}^N)\right] = \sum_i^N m_i \dot{\mathbf{r}}_i \cdot \ddot{\mathbf{r}}_i + \sum_i^N \frac{\partial \mathcal{V}}{\partial \mathbf{r}_i} \cdot \dot{\mathbf{r}}_i \tag{1}$$

$$= \sum_i^N \dot{\mathbf{r}}_i \cdot \mathbf{f}_i - \sum_i^N \dot{\mathbf{f}}_i \cdot \mathbf{r}_i = 0 \tag{2}$$

Therefore, standard molecular dynamics simulates the microcanonical ensemble (NVE).

Newtonian dynamics also obeys time reversal symmetry. I.e. If we reverse the velocities of all the particles, keeping the positions the same, the system will retrace its trajectory back into the past. More formally,

$$\mathbf{r}^N\left(t, \mathbf{r}^N(0), -\mathbf{p}^N(0)\right) = \mathbf{r}^N\left(-t, \mathbf{r}^N(0), \mathbf{p}^N(0)\right) \tag{3}$$

$$\mathbf{p}^N\left(t, \mathbf{r}^N(0), -\mathbf{p}^N(0)\right) = -\mathbf{p}^N\left(-t, \mathbf{r}^N(0), \mathbf{p}^N(0)\right) \tag{4}$$

Integration Scheme

All based on Taylor expansions of position in time.

A simple example. Expand position to third order in time.

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \dot{\mathbf{r}}_i(t)\delta t + \frac{\ddot{\mathbf{r}}_i(t)}{2}\delta t^2 + \frac{\dddot{\mathbf{r}}_i(t)}{6}\delta t^3 + \mathcal{O}(\delta t^4) \tag{5}$$

As $\dot{\mathbf{r}}_i = \mathbf{v}_i$ and using Newton's 2nd law ($\mathbf{f}_i = m_i\ddot{\mathbf{r}}_i$), rewrite as

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

Similarly,

$$\mathbf{r}_i(t-\delta t) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\delta t + \frac{\mathbf{f}_i(t)}{2m_i}\delta t^2 - \frac{\dddot{\mathbf{r}}_i(t)}{6}\delta t^3 + \mathcal{O}(\delta t^4) \tag{7}$$

Adding these two equations gives

$$\mathbf{r}_i(t+\delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\delta t) + \frac{\mathbf{f}_i(t)}{m_i}\delta t^2 + \mathcal{O}(\delta t^4) \tag{8}$$

Note that due to the cancellation of odd time derivatives of $\mathbf{r}_i$:

- this expression is accurate to third order in time, even though only forces are used.
- velocities do not appear in the expression.

Get velocity, by subtracting Eq. 7 from Eq. 6.

$$\mathbf{v}_i(t) = \frac{1}{2\delta t}\left[\mathbf{r}_i(t+\delta t) - \mathbf{r}_i(t-\delta t)\right] + \mathcal{O}(\delta t^2) \tag{9}$$

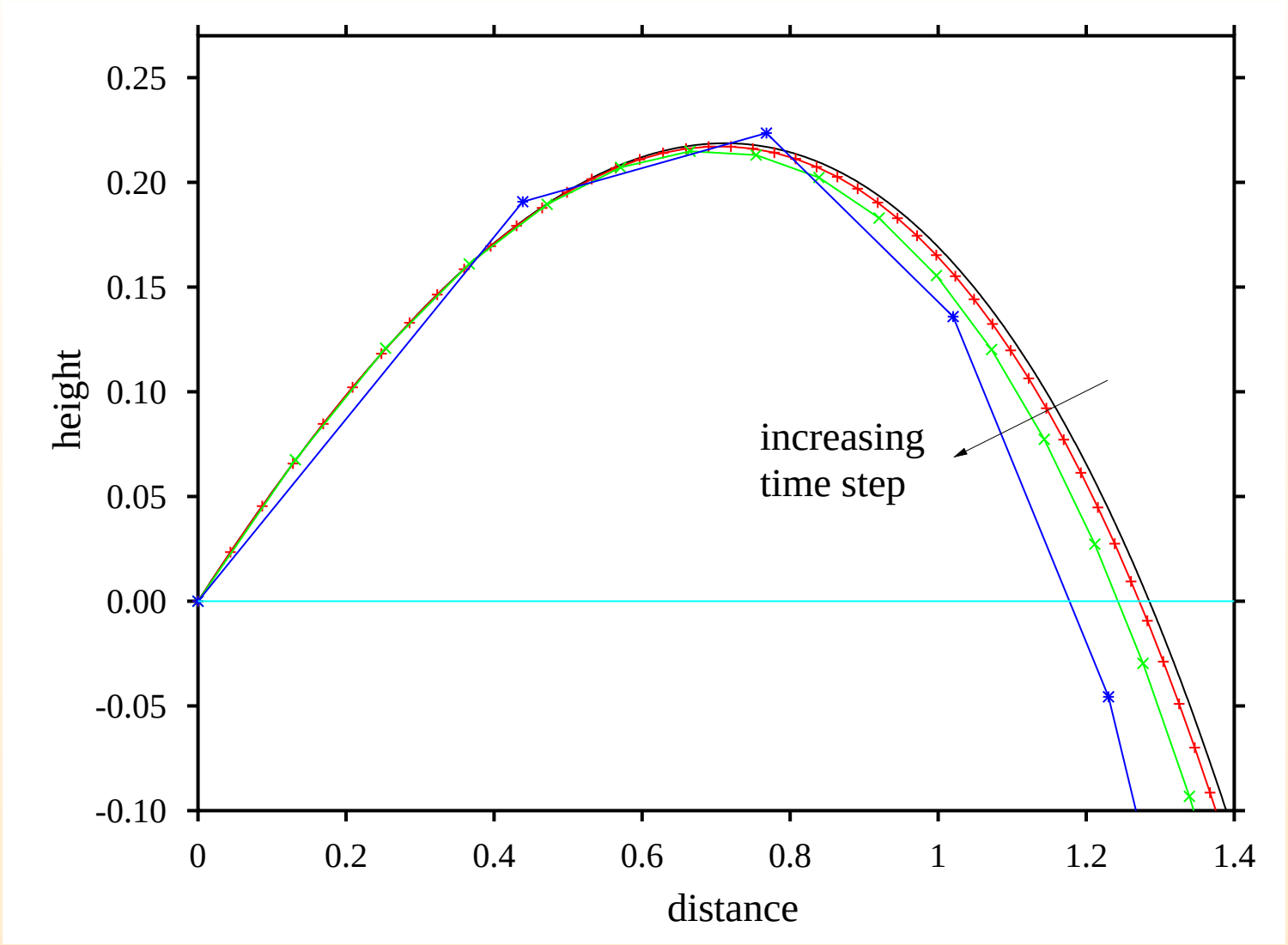
This is the Verlet scheme.

An alternative scheme that has identical trajectories to the Verlet algorithm, but a more convenient expression for the velocity is velocity Verlet.

$$\mathbf{r}_i(t+\delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\delta t + \frac{\mathbf{f}_i(t)}{2m_i}\delta t^2 \tag{10}$$

$$\mathbf{v}_i(t+\delta t) = \mathbf{v}_i(t) + \frac{1}{2m_i}\left[\mathbf{f}_i(t) + \mathbf{f}_i(t+\delta t)\right]\delta t \tag{11}$$

Example: Motion of a projectile with air resistance



Ergodic hypothesis

Q: Can we use molecular dynamics to obtain equilibrium properties for the microcanonical ensemble?

Ergodic hypothesis: Time averages of ergodic systems in the limit of trajectories of infinite length are equivalent to ensemble averages.

I.e. $\bar{A} = \langle A \rangle_{NVE}$, where

$$\bar{A} = \lim_{\Delta t \to \infty} \int_0^{\Delta t} A\left(\mathbf{r}^N(t), \mathbf{p}^N(t)\right) dt \tag{12}$$

$$\langle A \rangle_{NVE} = \frac{\int A\left(\mathbf{r}^N, \mathbf{p}^N\right) \delta\left[\mathcal{H}\left(\mathbf{r}^N, \mathbf{p}^N\right) - E\right] d\mathbf{r}^N d\mathbf{p}^N}{\int \delta\left[\mathcal{H}\left(\mathbf{r}^N, \mathbf{p}^N\right) - E\right] d\mathbf{r}^N d\mathbf{p}^N} \tag{13}$$

This is equivalent to saying that each point on the constant NVE manifold is equally likely to be sampled in the MD simulation.

On a harmonic potential energy surface this would not be true. Anharmonicities that couple different degrees of freedom allow energy to be redistributed between different modes and prevents the system getting stuck in periodic orbits. The chaotic motion causes the system to lose memory of its initial conditions.

However, if there are large barriers relative to the kinetic energy per degree of freedom, this can prevent ergodicity being achieved on the simulation time scales.

In practice, as we have evaluated A at a series of discrete points along the trajectory, we use

$$\bar{A} \cong \frac{1}{M} \sum_{m=1}^{M} A(t_m) \tag{14}$$

What do we want from an integrator

Accuracy? I.e. an integrator that most closely follows the true trajectory.

Important for *satellite* dynamics, but not for *molecular* dynamics.

Molecular chaos means that the simulated trajectory quickly diverges from the true trajectory. I.e. the trajectories show extreme sensitivity to small difference in initial conditions.

For two trajectories that at $t = 0$ only differed by a small perturbation ϵ of the initial conditions

$$|\Delta\mathbf{r}(t)| = |\mathbf{r}'(t) - \mathbf{r}(t)| \sim \epsilon \exp(\lambda t) \tag{15}$$

where λ is the Lyapunov exponent.

$\Rightarrow$ Trying to achieve accuracy in MD is pointless.

New aim: representative trajectory and good statistical properties.

In particular, important that the integration scheme obeys time reversal symmetry, and conserves the energy over long times.

The latter is achieved by area-preserving or symplectic algorithms. The action of such an integrator on a volume in phase space leaves the volume unchanged.

Why is this important? If $\Omega(E)$ (the density of states) is mapped onto Ω' with different volume, it will no longer simulate a microcanonical ensemble.

Verlet algorithms are symplectic.

nb

$$\Omega(E) \propto \int \delta\left[\mathcal{H}\left(\mathbf{r}^N, \mathbf{p}^N\right) - E\right] d\mathbf{r}^N d\mathbf{p}^N \tag{16}$$

Force evaluation

Calculating the forces is the most computationally time-consuming part of any MD calculation.

For a pair potential

$$\mathcal{V}\left(\mathbf{r}^N\right) = \sum_i^N \sum_{j>i}^N v(r_{ij}) = \frac{1}{2}\sum_i^N \sum_{j\neq i}^N v(r_{ij}) \tag{17}$$

$$\mathbf{f}_i = -\frac{\partial \mathcal{V}}{\partial \mathbf{r}_i} = -\frac{1}{2}\sum_{j\neq i}\left(\frac{\partial v(r_{ij})}{\partial \mathbf{r}_i} + \frac{\partial v(r_{ji})}{\partial \mathbf{r}_i}\right) = -\sum_{j\neq i}\left.\frac{dv(r)}{dr}\right|_{r=r_{ij}} \frac{\mathbf{r_{ij}}}{r_{ij}} \tag{18}$$

The number of operations required to calculate the forces on all the atoms scales as N^2.

Initialization

Need to provide initial $\mathbf{r}^N$ and $\mathbf{p}^N$.

$\mathbf{r}^N(0)$ easy if simulating an ordered state with known structure, e.g. crystal, native state of protein.

If simulating a disordered state such as a liquid, E.g.

- Create random configuration and then minimize the potential energy (so that the potential energy is less that total energy).
- Or perform an initial simulation, where a crystal is rapidly heated to melt it, and then cooled to the desired total energy.

The velocities are assigned from a random distribution (ideally a Gaussian one, so that they satisfy the Maxwell-Boltzmann distribution).

The velocities are then scaled so that the system has the desired total energy E.

Note: Temperature can be calculated through the equipartition theorem

$$\left\langle \sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m_i} \right\rangle = \frac{3}{2} NkT \tag{19}$$

Monte Carlo

Monte Carlo is a stochastic method to calculate integrals.

Why integrals?

Common in classical statistical mechanics

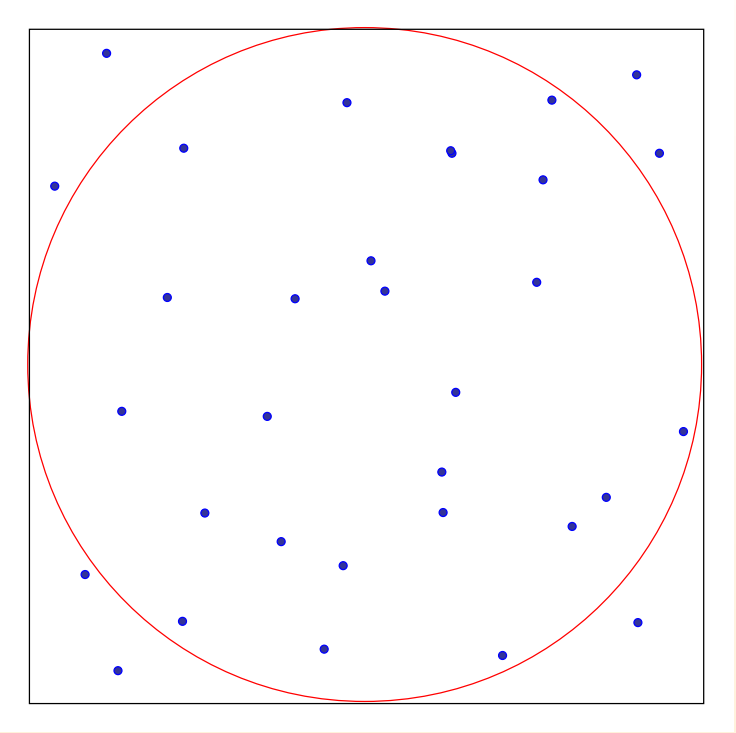
$$\langle A \rangle_{NVT} = \frac{\int A\left(\mathbf{r}^N, \mathbf{p}^N\right) \exp(-\beta E) d\mathbf{r}^N d\mathbf{p}^N}{\int \exp(-\beta E) d\mathbf{r}^N d\mathbf{p}^N} \tag{20}$$

1947: von Neumann, Metropolis and Ulam: First computational Monte Carlo application. Diffusion of neutrons in fissile material.

1953: Metroplis $et\ al.$: First Monte Carlo simulation of a molecular system

Called Monte Carlo because of its use of random numbers.

Calculating π by MC



Throwing darts randomly at the target

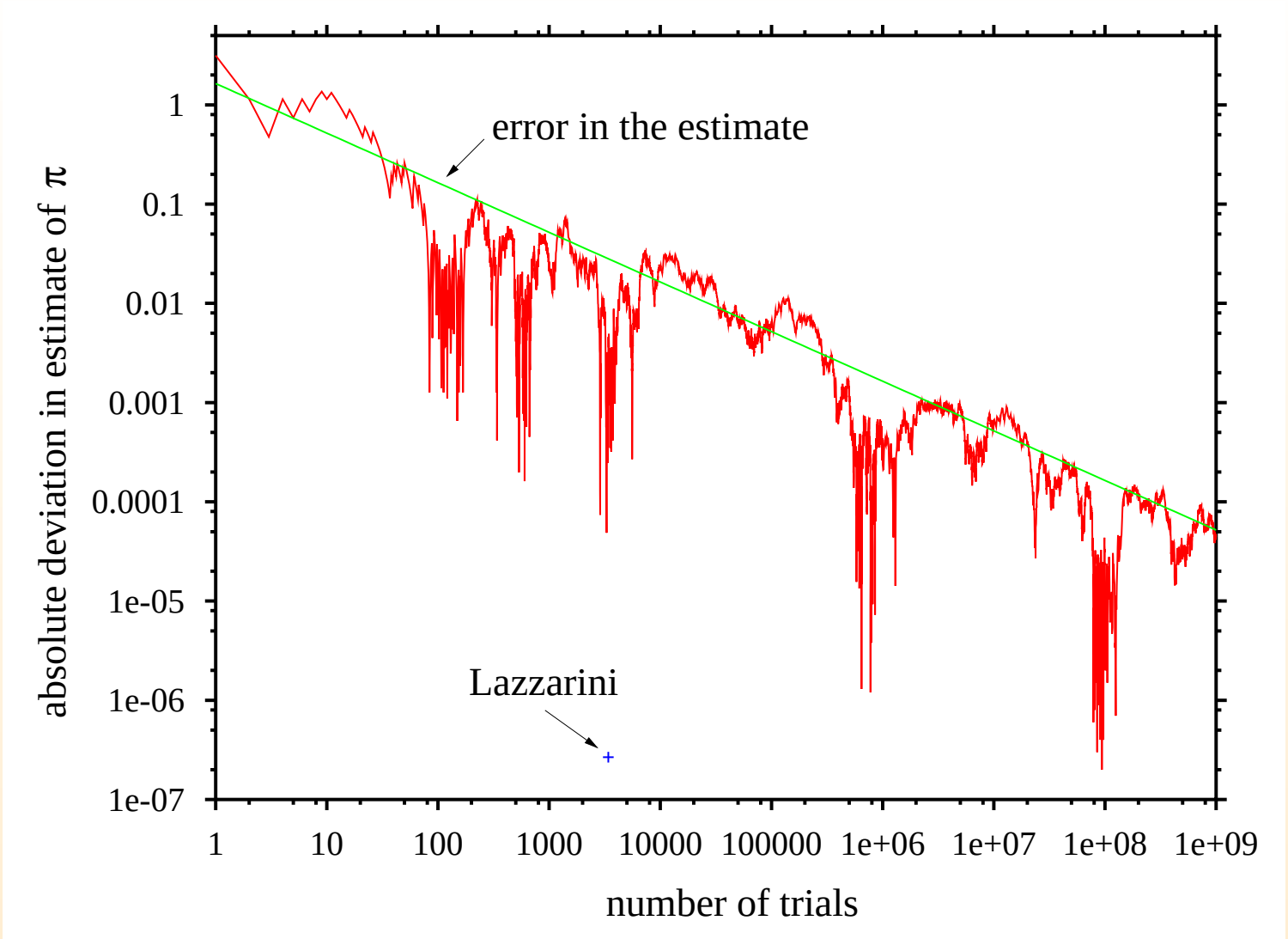
$$\lim_{n_{throws}\to\infty} \frac{n_{hits}}{n_{throws}} = \frac{\pi}{4} \tag{21}$$

As an integral

$$\frac{1}{n_{throws}} \sum_i \mathcal{O}_i = \frac{\int_{-1}^{1} \mathcal{O}(x,y)dxdy}{\int_{-1}^{1} dxdy} \tag{22}$$

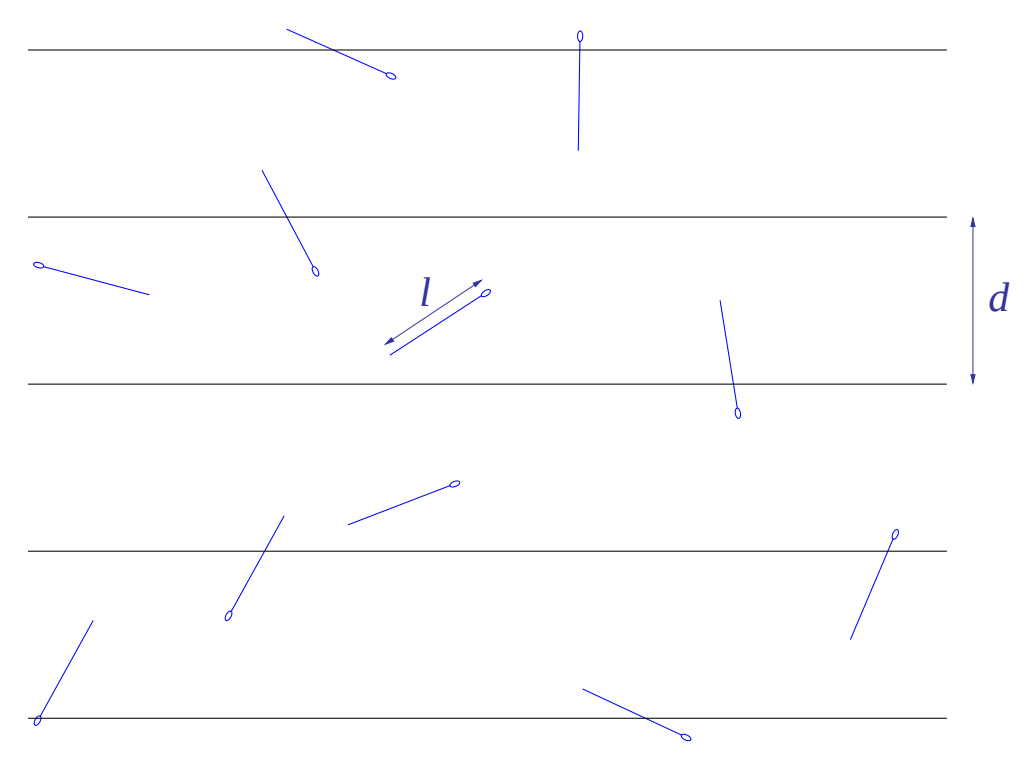
where

$$\mathcal{O}(x,y) = \begin{cases} 1 & r \leq 1 \\ 0 & r > 1 \end{cases} \tag{23}$$



Results of MC estimation of π.

Buffon's Needle



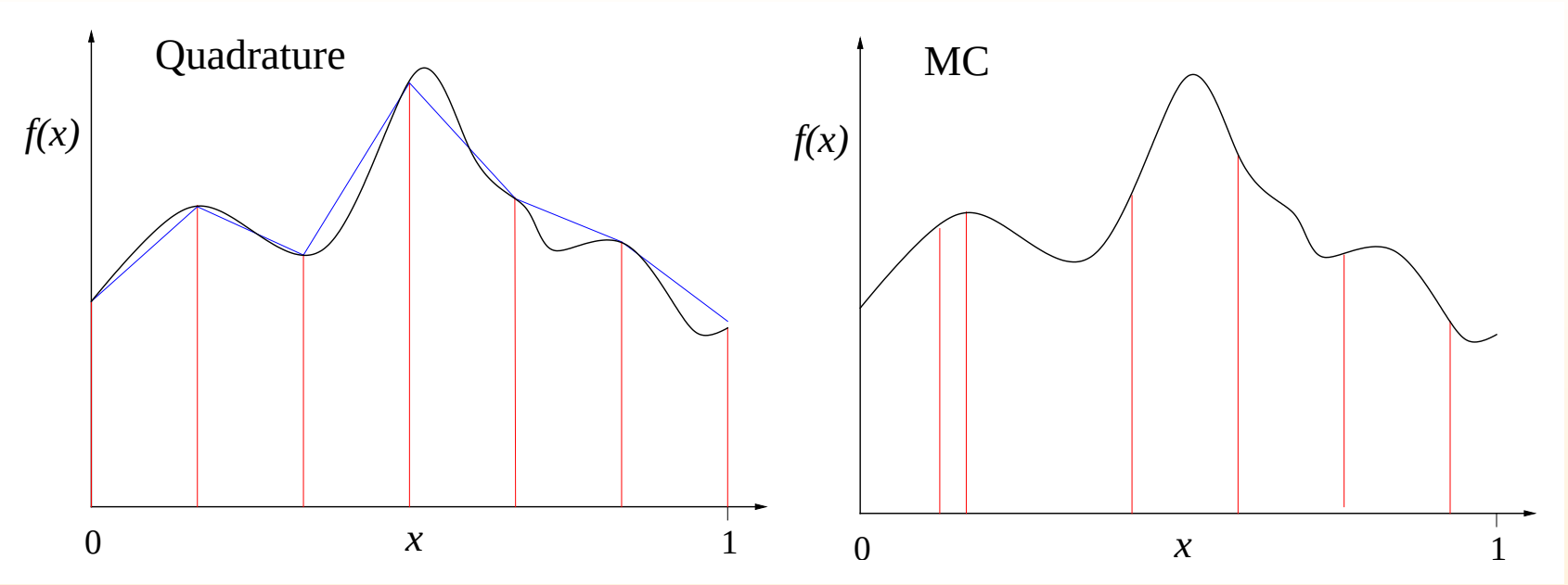
The probability of a needle overlapping a crack between floorboards can also be used to obtain π.

$$p_{overlap} = \frac{2l}{\pi d} \qquad (24)$$

First suggested by Comte de Buffon in 1777.

In 1901 Lazzarini claimed an estimate of $\pi = 3.1415929$ $(355/113)$, i.e. an accuracy of 7SF, on the basis of 3408 trials.

Estimating 1D integrals



MC estimate for the integral

$$I = \int_0^1 f(x)dx \approx \frac{1}{M}\sum_{i=1}^{M} f(x_i) \tag{25}$$

where M random points x_i in the interval $[0, 1]$ have been selected.

Error analysis: MC

The error in the estimate of I, σ_I, obeys

$$\sigma_I^2 = \left\langle \left(I - \langle I \rangle \right)^2 \right\rangle \tag{26}$$

$$= \left\langle \left(\frac{1}{M} \sum_{i=1}^{M} f(x_i) - \left\langle \frac{1}{M} \sum_{i=1}^{M} f(x_i) \right\rangle \right)^2 \right\rangle \tag{27}$$

$$= \frac{1}{M^2} \left\langle \left(\sum_{i=1}^{M} \left(f(x_i) - \langle f(x) \rangle \right) \right) \left(\sum_{j=1}^{M} \left(f(x_j) - \langle f(x) \rangle \right) \right) \right\rangle \tag{28}$$

$$= \frac{1}{M^2} \left\langle \sum_{i=1}^{M} \left(f(x_i) - \langle f(x) \rangle \right)^2 \right\rangle = \frac{1}{M} \sigma_f^2 \tag{29}$$

as $f(x_i)$ and x_i are uncorrelated.

Error analysis: Quadrature

Integration using the simple trapezium rule gives

$$I = \frac{1}{M}\left(\frac{1}{2}f(0) + \sum_{i}^{N-2} f(x_i) + \frac{1}{2}f(1)\right) + \mathcal{O}\left(\frac{1}{M^3}\right) \tag{30}$$

where the $x_i = i/N$ are equally spaced.

Thus, for 1D integrals quadrature methods converge much faster than the MC approach.

For higher dimensional integrals using the trapezium rule, the error has the same dependence on the number of points m in each dimension, i.e. $\mathcal{O}(1/m^3)$. However, as the total number of points $M = m^d$ the error is $\mathcal{O}(1/M^{3/d})$.

The scaling for the MC error is independent of dimension, i.e. $\sigma_I \sim 1/\sqrt{M}$.

$\Rightarrow$ MC scales better than trapezium rule for $d > 6$.

Importance sampling

As the Boltzmann weights in the statistical mechanics integrals that we want to evaluate (Eq. 20) are non-negligible for only a small fraction of configuration space, random uniform sampling will be of no use.

e.g. for 100 hard spheres at the freezing transition, only one in 10^{260} configurations have a non-zero Boltzmann weight.

Therefore, it would be useful to preferentially sample regions where an integral has the most weight.

Back to 1D: Sample points from a probability distribution $w(x)$. Estimate of integral now:

$$I \approx \frac{1}{M} \sum_{i=1}^{M} \frac{f(x_{i|w})}{w(x_{i|w})} \tag{31}$$

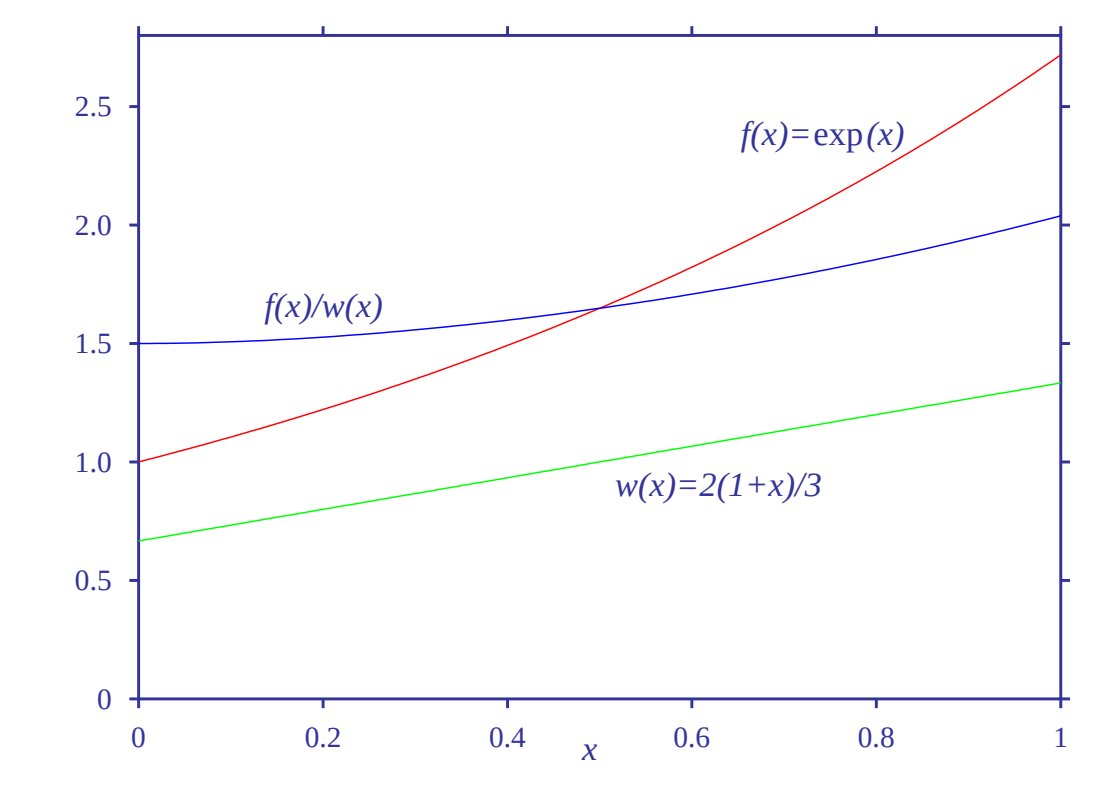
where $x_{i|w}$ means that the x_i have been drawn from the distribution $w(x)$. The error is now:

$$\sigma_I^2 = \frac{1}{M^2} \left\langle \sum_{i=1}^{M} \left(\frac{f(x_{i|w})}{w(x_{i|w})} - \left\langle \frac{f(x)}{w(x)} \right\rangle \right)^2 \right\rangle = \frac{1}{M} \sigma_{f/w}^2 \tag{32}$$

Even though this procedure has not changed the scaling, it's effect on the prefactor can be useful.

If $w(x) = f(x)/\langle f \rangle$, then $\sigma_{f/w} = 0$. However, this choice requires one to already know the integral we are trying to evaluate!

A good choice of $w(x)$ can lead to enhanced accuracy.



For uniform sampling

$$\sigma_I \approx \frac{0.5}{\sqrt{N}} \qquad (33)$$

For importance sampling, using the weight function $w(x) = 2(1+x)/3$

$$\sigma_I \approx \frac{0.16}{\sqrt{N}} \qquad (34)$$

To apply this importance sampling approach directly to our statistical mechanical integrals, the ideal weight function would be $\exp(-\beta E)/Q$. However, to generate such a distribution of configurations, one would first need to know the partition function Q.

Metropolis escaped from this impasse by introducing a Markov chain method, where one only needed to know the relative probabilities of different configurations. It uses a biased random walk.

Metropolis Monte Carlo

Basic approach

- Start with a configuration o
- Choose and accept a new configuration n with a transition probability $\pi(o \rightarrow n)$
- Calculate properties of interest for the current configuration and add to average.
- Repeat to create a trajectory through phase space

Note:

- The configuration does not have to change at each step, i.e.

$$\pi(o \rightarrow o) = 1 - \sum_{n \neq o} \pi(o \rightarrow n) \neq 0 \tag{35}$$

- The trajectories generated are stochastic (cf MD trajectories are deterministic).

Detailed balance

Aim: to sample configurations from the Boltzmann distribution.

Thought experiment:

- Have a large ensemble of MC trajectories for a given system.
- Probability that a configuration appears within this ensemble obeys the Boltzmann distribution, i.e. $P(o) = \exp(-\beta E_o)/Q$.

Key requirement: form of transition probabilities must maintain this distribution. i.e. it is stationary, and the net probability flow into and out of every state must balance.

$$P(o)\sum_i \pi(o \to i) = \sum_j P(j)\pi(j \to o) \tag{36}$$

This equation can be satisfied, if 'detailed balance' is obeyed:

$$P(o)\pi(o \to n) = P(n)\pi(n \to o) \tag{37}$$

This condition guarantees equilibrium is maintained. Does not say if or how equilibrium will be reached.

$\pi(o \to n)$ broken into two parts.

$$\pi(o \to n) = \alpha(o \to n) acc(o \to n) \tag{38}$$

- $\alpha(o \to n)$: The probability of performing a trial move from o to n
- $acc(o \to n)$: The probability that the trial move is accepted. Otherwise, move is rejected and configuration o is retained.

As in the original Metropolis scheme, it is common that α is symmetrical, i.e. $\alpha(o \to n) = \alpha(n \to 0)$.

Detailed balance implies:

$$\frac{\pi(o \to n)}{\pi(n \to 0)} = \frac{acc(o \to n)}{acc(n \to o)} = \frac{P(n)}{P(o)} = \exp\left[-\beta(E_n - E_o)\right] \tag{39}$$

Metropolis acceptance criterion

Many possible choices of $acc(o \to n)$ that satisfy Eq. 39.

The form proposed by Metropolis $et\ al.$ (and most commonly used) is

$$acc(o \to n) = \begin{cases} P(n)/P(o) & P(n) < P(o) \\ 1 & P(n) \geq P(o) \end{cases} \tag{40}$$

For the Boltzmann distribution, this is

$$acc(o \to n) = \begin{cases} \exp\left[-\beta(E_n - E_o)\right] & E_n > E_0 \\ 1 & E_n \leq E_o \end{cases} \tag{41}$$

Downhill moves are always accepted, whereas uphill moves are accepted with a probability that decreases with the energy difference.

Integrating out the momenta

In the partition function, the integral over the momentum coordinates can be easily performed.

$$Q = \frac{1}{N!\Lambda^{3N}} Z \tag{42}$$

where Λ is the thermal wavelength, and Z is the configurational partition function:

$$Z = \int \exp\left[-\beta \mathcal{V}\left(\mathbf{r}^N\right)\right] d\mathbf{r}^N \tag{43}$$

Similarly, for quantities that only depend on $\mathbf{r}^N$, no point using MC to perform the integral over the momenta in the average $\langle A \rangle_{NVT}$ (Eq. 20).

Therefore, in Metropolis MC only $\mathbf{r}^N$ is considered and configurations are sampled with probability

$$P\left(\mathbf{r}^N\right) = \frac{1}{Z} \exp\left[-\beta \mathcal{V}\left(\mathbf{r}^N\right)\right] \tag{44}$$

Basic step in more detail

To move from step k to $k+1$

- Select a particle i at random from $\mathbf{r}_k^N$
- Add a random displacement to this atom: $\mathbf{r}_i' = \mathbf{r}_i + \boldsymbol{\Delta}$
- Calculate the potential energy $\mathcal{V}(\mathbf{r}_{trial}^N)$ for new trial configuration.
- Accept the move $\mathbf{r}_k^N \rightarrow \mathbf{r}_{trial}^N$ with probability

$$acc(o \rightarrow n) = \min\left\{1, \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r}_{trial}^N\right) - \mathcal{V}\left(\mathbf{r}_k^N\right)\right)\right]\right\} \tag{45}$$

  On acceptance

$$\mathbf{r}_{k+1}^N = \mathbf{r}_{trial}^N \tag{46}$$

  On rejection

$$\mathbf{r}_{k+1}^N = \mathbf{r}_k^N \tag{47}$$

- Add $A(\mathbf{r}_{k+1}^N)$ to average.

Notes on algorithm

Random displacement $\boldsymbol{\Delta}$ usually chosen uniformly from a cube:

$$\Delta_\alpha = \Delta_{max}\,(ran[0,1] - 0.5) \qquad \alpha = x, y, z \tag{48}$$

Choice of step size, Δ_{max}:

- If Δ_{max} is too large, most steps will be rejected.
- If Δ_{max} is too small, although most steps will be accepted, the system will only move very slowly through configuration space.

Optimal step size: Lowest statistical error for given CPU time.

Rule of thumb: 50% acceptance rate reasonable for continuous potentials (20% for hard-core systems).

Optimal step size will vary with conditions. E.g. smaller in dense systems.

How to choose Δ_{max}? One way is to adjust the step size dynamically during the equilibration period.

For transition matrix to be symmetric, i.e. $\alpha(o \to n) = \alpha(o \to n)$:

- choice of atom needs to be random
- Maximum step size needs to be constant.

Why move only one atom at a time?

Cost of N single-particle moves is similar to that for a single N-particle move. For pair potential,

$$\mathcal{V}\left(\mathbf{r}^N_{trial}\right) = \mathcal{V}\left(\mathbf{r}^N_k\right) + \sum_{j \neq i} v\left(r^{trial}_{ij}\right) - v\left(r^k_{ij}\right) \tag{49}$$

If the probability of a single-particle move being rejected is p_{rej}, the probability that a collective move of all N particles is $(1 - p_{rej})^N$.

$\therefore$ single-particle move algorithms move more rapidly through configuration space.

Jargon: an MC cycle or sweep corresponds to N single-particle moves.

Markov chains and ergodicity

The random walk performed by an MC simulation is an example of a Markov process: A stochastic process without memory, i.e. $\pi(o \to n)$ is independent of the previous history of moves.

Ergodicity is assumed: i.e. one can go between any two states of the system in a finite number of steps.

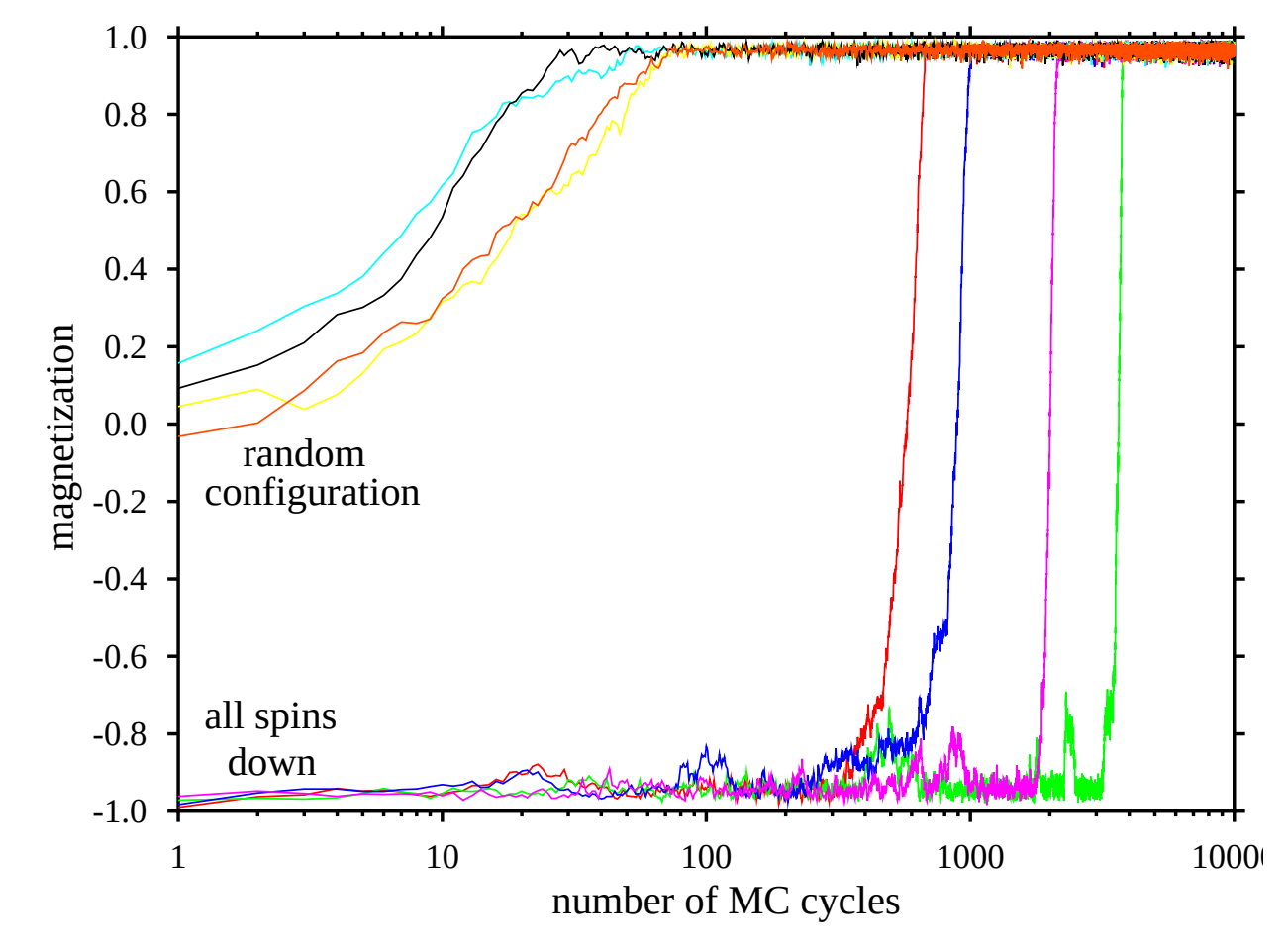
In practice, to prevent errors due to incomplete sampling, this number of steps needs to be shorter than the simulation.

Large (much larger than kT) barriers can make equilibrium sampling very difficult.

Approach to equilibrium: Markov chains approach the equilibrium distribution exponentially fast. But this still can be a long time, and is dependent on the starting configuration.

$\therefore$ need to equilibrate the system, before starting to take averages.

Equilibration in Ising model



2D Ising model: 40×40 *lattice,* $kT/J = 1.8$*,* $H = 0.1$*.*

$$\mathcal{H} = -J \sum_{i,j} S_i S_j + H \sum_i S_i \tag{50}$$

Error analysis in MC trajectories

The variance in a quantity measured in an M step MC simulation is

$$\sigma_M^2(A) = \frac{1}{M}\sum_{i=1}^{M}\left(A_k - \langle A\rangle_M\right)^2 = \left\langle A^2\right\rangle_M - \langle A\rangle_M^2 \tag{51}$$

If these measurements were independent, the variance in the average would be

$$\sigma^2\left(\langle A\rangle_M\right) \approx \frac{1}{M}\sigma_M^2 \tag{52}$$

However, the configurations generated by a MC trajectory are *correlated*.

Auto-correlation functions can be used to measure how long it takes a system to lose ''memory'' of the state it was in.

$$C_{AA}(k) = \frac{1}{M}\sum_{k'}\left(A_{k'} - \langle A\rangle\right)\left(A_{k+k'} - \langle A\rangle\right) \tag{53}$$

Correlation functions typically show an exponential decay, i.e.

$$C_{AA}(k) \sim \exp\left(-k/n_\tau\right) \tag{54}$$

where $2n_\tau$ is a reasonable estimate of the number of steps between independent measurements.

$\therefore$ $n_m = M/2n_\tau$ is a measure of the number of statistically independent measurements, and

$$\sigma^2\left(\langle A\rangle_M\right) = \frac{1}{n_M - 1}\sigma_M^2 \tag{55}$$

Block averages

An alternative approach is to use block averages.

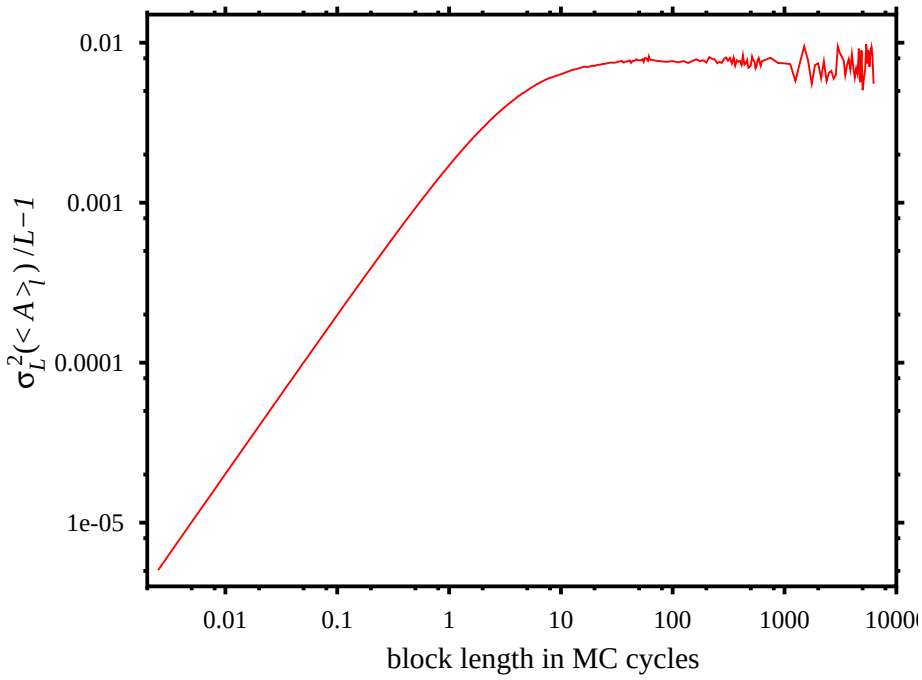
Take L partial averages $\langle A \rangle_l$ over blocks of length $l = M/L$ MC steps.

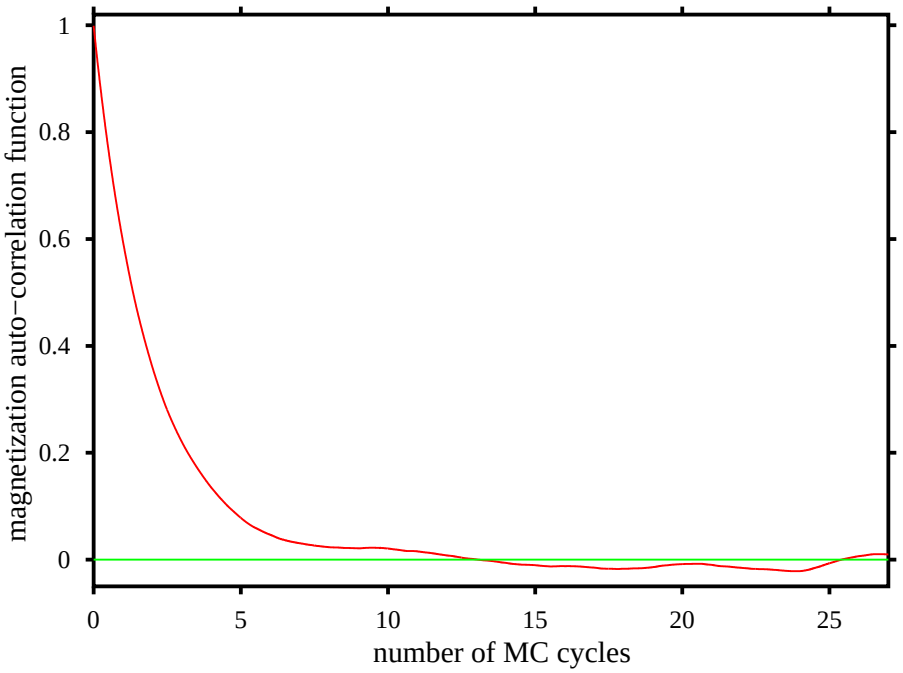
$$\sigma_L^2(\langle A \rangle_l) = \frac{1}{L} \sum_{i=1}^{L} (\langle A \rangle_l - \langle A \rangle_M)^2 \tag{56}$$

As the block size increases, eventually $l > 2n_\tau$ and the $\langle A \rangle_l$ become independent. $\therefore$

$$\frac{\sigma_L^2(\langle A \rangle_l)}{L-1} \approx \sigma^2(\langle A \rangle_M) \tag{57}$$

Plotting $\sigma_L^2(\langle A \rangle_l)/(L-1)$ as a function of L, this quantity will increase with decreasing L and plateau at the correct variance for $L \leq n_M$.





2D Ising model: 40×40 *lattice,* $kT/J = 1.8$*.*

Systematic Errors

• Finite-size errors

If one is interested in estimating bulk properties from MC simulations, there are often errors due to the necessarily finite size of the simulation system.

Such finite-size errors typically scale as

$$Error \sim \frac{1}{\sqrt{N}} \tag{58}$$

Finite-size scaling (i.e. examining how your property depends on N) can often be used to extrapolate to the bulk limit.

• Failure to reach equilibrium

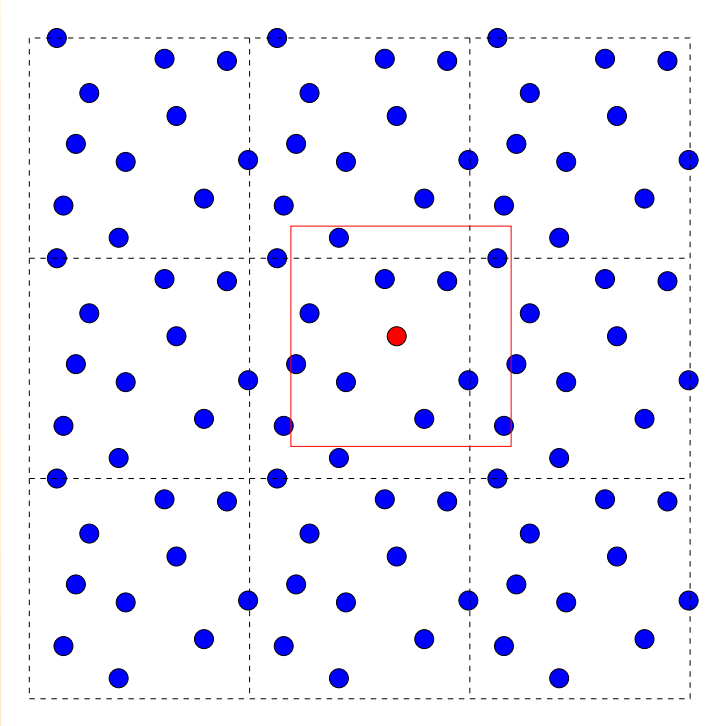
Large barriers can make the convergence of averages very slowly, as passage over these barriers represent *rare events*.

Periodic Boundary conditions

Normally, one wants to simulate a bulk system using a simulation of a finite number of particles.

For small systems, the fraction of particles at any surface is high, e.g. open or hard boundary conditions

Therefore, to minimize such effects, periodic boundary conditions are used.



The simulation cell is treated as the primitive cell of an infinite periodic lattice of identical cells.

The simulation box is often chosen to be cubic, but other choices are possible.

$$\mathcal{V}_{tot} = \frac{1}{2} \sum_{i,j,\mathbf{n}} v\left(|\mathbf{r}_{ij} + \mathbf{n}L|\right), \tag{59}$$

where L is the edge length of the (cubic) box, and $\mathbf{n}$ is any vector of three integers.

An infinite sum!

Minimum image convention: an atom only interacts with the nearest image of any particle.

Potential problems with PBC: properties that depend on long wavelengths.

Truncating the interactions

The most expensive part of any calculation is evaluating the energy and its derivatives.

For a pair potential, the cost of evaluating the potential is $\mathcal{O}(N^2)$, as all pairs of atoms should be formally considered.

However, the interactions between distant atoms for short-ranged interactions are often very small, and so the potential can be truncated.

For periodic boundary conditions, usually choose $r_c < L/2$.

If $v(r_c) \neq 0$ this leads to an error, but this can be corrected for:.

$$\mathcal{V}\left(\mathbf{r}^N\right) = \sum_{r_{ij}<r_c} v(r_{ij}) + \frac{N\rho}{2}\int_{r_c}^{\infty} 4\pi r^2 v(r) \tag{60}$$

Jargon: this extra term is called a tail correction.

Tail correction does not converge for $v(r) \propto r^{-n}$ if ($n \leq 3$). Need to use 'Ewald summation'.

Common ways to truncate the potential

- Simple truncation

$$v'(r) = \begin{cases} v(r) & r \leq r_c \\ 0 & r > rc \end{cases} \tag{61}$$

  MC: okay, but needs an additional impulsive correction for the pressure.

  MD: problems with discontinuity in the potential.

- Truncate and shift

$$v'(r) = \begin{cases} v(r) - v(r_c) & r \leq r_c \\ 0 & r > rc \end{cases} \tag{62}$$

  MD: still a discontinuity in the derivatives

Code scaling

Truncation reduces the number of interactions that need to be calculated from $\mathcal{O}(N^2)$ to $\mathcal{O}(mN)$, where m is the average number of neighbours of an atom within the cutoff distance.

However, $\mathcal{O}(N^2)$ distances still need to be checked at each step.

To reduce the overall scaling of the algorithm to $\mathcal{O}(N)$, two common methods:

- Neighbour list
  Store a list of the particles that are within r_v of each atom, where $r_v > r_c$. Only calculate the interactions between particles on the neighbour lists. List needs to be updated when atoms have moved by $r_v - r_c$.

- Cell list
  Divide the simulation box up into cells. Only calculate the interactions between particles in the same or neighbouring cells.

Thermodynamics properties

In the canonical ensemble, the heat capacity can be calculated from fluctuations in the total energy:

$$C_v(T) = \frac{1}{kT^2}\left(\langle E^2\rangle - \langle E\rangle^2\right) \tag{63}$$

In the canonical ensemble,

$$p(E,T) = \frac{1}{Q(T)}\Omega(E)\exp\left(-\beta E\right) \tag{64}$$

Hence one can extract an estimate of the density of states, $\Omega(E)$ (to within a multiplicative constant) from the histogram $p(E,T)$.

This can then be used to calculate thermodynamic properties at any temper-

ature, e.g.

$$U(T) = \frac{\int E\,\Omega(E)\exp\left(-\beta E\right)dE}{\int \Omega(E)\exp\left(-\beta E\right)dE} \tag{65}$$

$$= \frac{\int p(E,T')E\exp\left((\beta'-\beta)E\right)dE}{\int p(E,T')\exp\left((\beta'-\beta)E\right)dE} \tag{66}$$

In the 'multi-histogram' approach $p(E,T)$ distributions from a number of temperatures are used to obtain a 'best' estimate of $\Omega(E)$.

Pressure

In the canonical ensemble, $P = -(\partial A/\partial V)_{N,T}$ leads to the virial expression for the pressure:

$$P = \rho kT + \frac{1}{dV}\left\langle \sum_{i<j} \mathbf{f}(r_{ij}) \cdot \mathbf{r}_{ij} \right\rangle, \tag{67}$$

where ρ is the density and d the dimension of the system.

This can be expressed in terms of the radial distribution function:

$$P = \rho kT - \frac{2\pi\rho^2}{3}\int r^3 g(r)\frac{dv(r)}{dr}dr \tag{68}$$

This form can be used to derive a tail correction for the measurement of the pressure.

The virial expression cannot straightforwardly be applied to hard particles. In MC it is also possible to relate the pressure to the probability of acceptance of a trial move that changes the volume of the simulation box.

Diffusion

Diffusion can be related to the mean-square displacement through the Einstein relation:

$$\Delta r^2(t) = \left\langle \left(\mathbf{r}_i(t) - \mathbf{r}_i(0)\right)^2 \right\rangle = 2dDt \tag{69}$$

where the average is over an equilibrium ensemble of initial conditions.

At short times, the motion of the particles is ballistic, so the limiting slope of a plot of $\Delta r^2(t)$ versus time is used to compute D.

The diffusion coefficient can also be related to the velocity auto-correlation function. As

$$\mathbf{r}_i(t) - \mathbf{r}_i(0) = \int_0^t v_i(t')dt' \tag{70}$$

$$\frac{d}{dt}\Delta r^2(t) = \frac{d}{dt}\int_0^t \int_0^t \langle \mathbf{v}(t') \cdot \mathbf{v}(t'') \rangle \, dt' dt'' = 2\int_0^t \langle \mathbf{v}(t) \cdot \mathbf{v}(t') \rangle \, dt' \tag{71}$$

Using the invariance of such correlation functions to the origin of time

$$D = \frac{1}{2d} \lim_{t\to\infty} \frac{d}{dt}\Delta r^2(t) = \frac{1}{d}\int_0^\infty \langle \mathbf{v}(\tau) \cdot \mathbf{v}(0) \rangle \, d\tau \tag{72}$$

Constant pressure Monte Carlo

For the isobaric ensemble, the relevant thermodynamic functions are the enthalpy $H = U + PV$ and the Gibbs free energy G.

$$\langle A \rangle_{NPT} = \frac{1}{Z_{NPT}} \int_0^\infty V^N \exp(-\beta P V) dV \int A\left(\mathbf{s}^N; V\right) \exp\left[-\beta \mathcal{V}\left(\mathbf{s}^N; V\right)\right] d\mathbf{s}^N \tag{73}$$

Scaled coordinates ($\mathbf{s}_i = \mathbf{r}_i / V^{1/3}$) have been used to separate out the volume dependence of the integral over $\mathbf{r}^N$.

A NPT MC scheme must sample configurations with the probability distribution:

$$P_{NPT}\left(\mathbf{s}^N; V\right) \propto \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{s}^N\right) + PV - N\log(V)/\beta\right)\right] \tag{74}$$

and must involve two kinds of moves

- moves that randomly displace a particle
- moves that randomly change the volume V

for both of which the transition matrix α is symmetric.

The choice of move type must be random to maintain detailed balance.

For particle moves the acceptance criterion reduces to that for the canonical ensemble.

For volume moves the acceptance criterion is

$$acc(o \to n) = \min\left\{1, \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{s}^N; V_n\right) - \mathcal{V}\left(\mathbf{s}^N; V_o\right) + P\Delta V - \frac{N}{\beta}\log\left[\frac{V_n}{V_o}\right]\right)\right]\right\} \tag{75}$$

where $\Delta V = V_n - V_o$.

The $P\Delta V$ term which favours a shrinkage of the volume is opposed by the $\log(V_n/V_o)$ term.

For potentials, such as the soft-sphere potential:

$$\mathcal{V}\left(\mathbf{r}^N\right) = \epsilon \sum_{i<j}\left(\frac{\sigma}{r_{ij}}\right)^m \tag{76}$$

the effect of the volume change on the energy is simple to compute

$$\mathcal{V}\left(\mathbf{s}^N; V_n\right) = \left(\frac{V_o}{V_n}\right)^{m/d} \mathcal{V}\left(\mathbf{s}^N; V_o\right) \tag{77}$$

However, for more complex potentials, it may be necessary to recompute the full potential. In this case, a volume move is roughly as expensive as N single-particle moves.

The relative frequency with which the two move types are chosen should reflect their relative expense.

The average size of the volume move should be optimized to give reasonable acceptance ratios.

Grand Canonical Monte Carlo

In the grand canonical ensemble, the system can exchange particles with a 'particle bath', thus fixing the chemical potential μ.

In this ensemble, the weight of each state is $\exp\left[-\beta\left(E_i - \mu N_i\right)\right]$ After integrating over the momenta, the probability distribution that the MC needs to sample is obtained:

$$P\left(\mathbf{s}^N; N\right) \propto \frac{V^N}{\Lambda^{3N} N!} \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{s}^N\right) - \mu N\right)\right] \tag{78}$$

Nb: as the integral over the momentum depends on N it has to be explicitly considered.

In addition to particle displacement moves, there are two new move types

- Removal of a randomly chosen particle
- Addition of a particle at a random position

These moves are performed with equal probability so that $\alpha(N \rightarrow N+1) = \alpha(N+1 \rightarrow N)$.

The acceptance criteria are given by

$$acc(N \rightarrow N-1) = \min\left\{1, \frac{\Lambda^3 N}{V}\exp\left[-\beta\left(\mu + \mathcal{V}\left(\mathbf{s}^{N-1}\right) - \mathcal{V}\left(\mathbf{s}^{N}\right)\right)\right]\right\} \quad (79)$$

$$acc(N \rightarrow N-1) = \min\left\{1, \frac{V}{\Lambda^3 (N+1)}\exp\left[-\beta\left(-\mu + \mathcal{V}\left(\mathbf{s}^{N+1}\right) - \mathcal{V}\left(\mathbf{s}^{N}\right)\right)\right]\right\} \quad (80)$$

Pros of grand-canonical ensemble:

- Direct access to μ
- Natural ensemble for processes, such as adsorption.

Cons

- Insertion probabilities can be very low for dense systems. Biasing techniques need to be used.

Widom Insertion

An alternative way to access the chemical potential is to measure it in a canonical simulation using 'Widom insertion'.

$$\mu = \left(\frac{\partial A}{\partial N}\right)_{V,T} \approx -kT \log\left[\frac{Q_{N+1}}{Q_N}\right] \tag{81}$$

Rewriting this in terms of scaled coordinates

$$\begin{aligned} \mu &= -kT \log\left[\frac{V}{\Lambda^3(N+1)}\right] - kT \log\left[\frac{\int \exp\left[-\beta\mathcal{V}\left(\mathbf{s}^{N+1}\right)\right] d\mathbf{s}^{N+1}}{\int \exp\left[-\beta\mathcal{V}\left(\mathbf{s}^{N}\right)\right] d\mathbf{s}^{N}}\right] && (82) \\ &= \mu_{id} + \mu_{ex} && (83) \end{aligned}$$

$\mu_{id} = -kT \log\left[\rho\Lambda^{-3}\right]$ is the chemical potential of an ideal gas.

$$\begin{aligned} \mu_{ex} &= -kT \log\left[\int \left(\frac{\int \exp\left[-\beta\Delta\mathcal{V}\right] \exp\left[-\beta\mathcal{V}\left(\mathbf{s}^{N}\right)\right] d\mathbf{s}^{N}}{\int \exp\left[-\beta\mathcal{V}\left(\mathbf{s}^{N}\right)\right] d\mathbf{s}^{N}}\right) d\mathbf{s}_{N+1}\right] && (84) \\ &= -kT \log\left[\int \left\langle \exp\left[-\beta\Delta\mathcal{V}\right]\right\rangle_{NVT} d\mathbf{s}_{N+1}\right] && (85) \end{aligned}$$

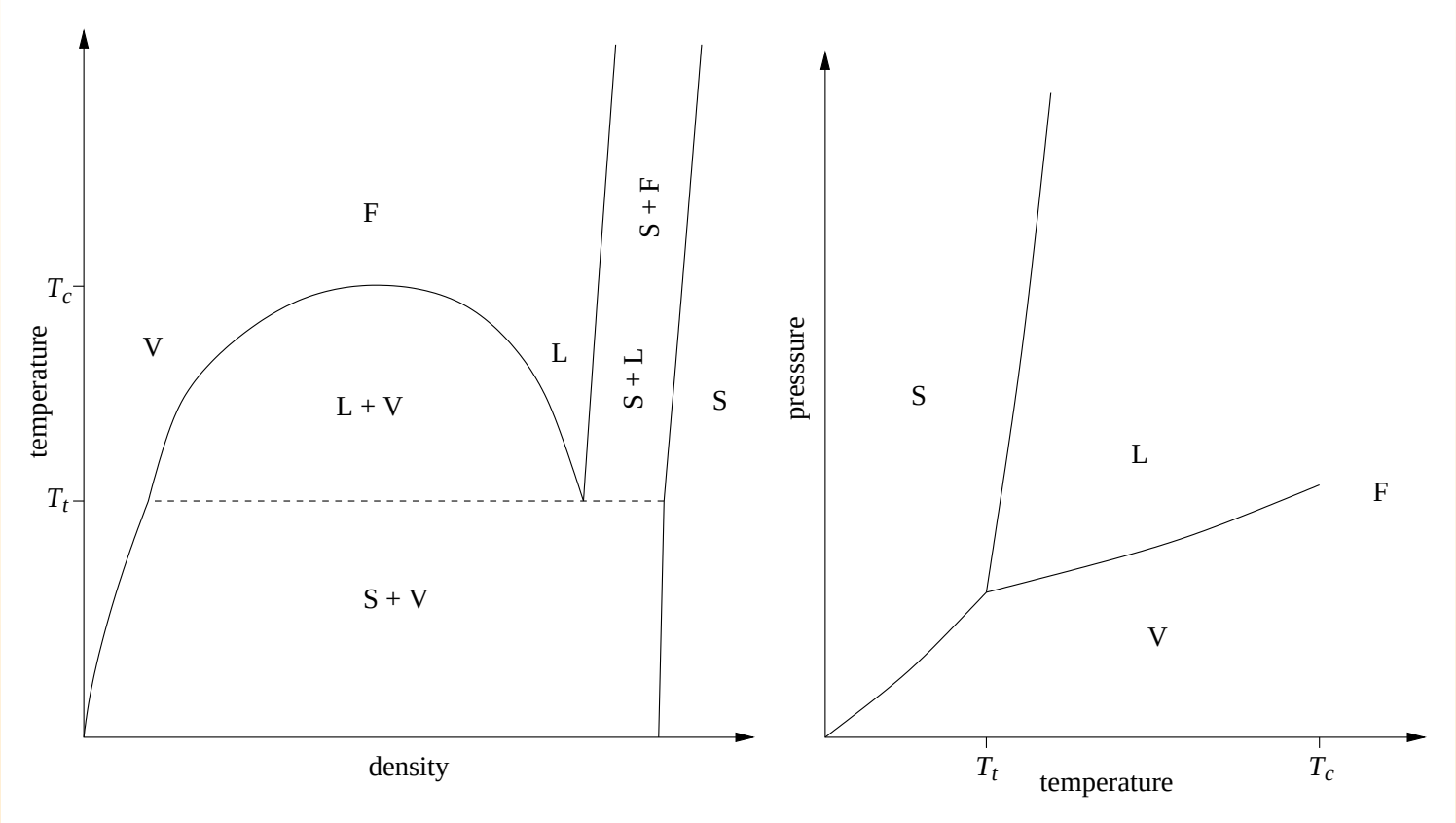
where $\Delta\mathcal{V} = \mathcal{V}\left(\mathbf{s}^{N+1}\right) - \mathcal{V}\left(\mathbf{s}^{N}\right)$.

Therefore, the excess chemical potential can be related to the average Boltzmann factor for inserting an extra particle into an N-particle system. The extra particle is a 'ghost' particle whose only purpose is to measure the chemical potential. It is never actually added to the system.

As with grand-canonical MC the insertion probability can be very low for dense systems leading to bad statistics.

Computing Phase Diagrams

One important use of simulation techniques is to calculate phase diagrams.



Direct simulation approaches hindered by:

- dynamics of phase transition may be slow
- the effect of the interface between the two phases

Points on coexistence lines are defined by where the free energies or chemical potentials of the two phases are equal.

However, computing free energies is hard because it is equivalent to knowing the partition function, e.g. $A = -kT \log Q(N, V, T)$.

Computing free energies differences is more tractable.

Thermodynamic Integration

Trick: Compute the difference in free energy between the system of interest and a reference system for which the free energy is known (e.g. ideal gas, harmonic crystal).

A path between these states can be defined.

$$V(\lambda) = V_{ref} + \lambda \left(V_{sys} - V_{ref} \right) \tag{86}$$

The partition function for a given λ is

$$Q(N, V, T; \lambda) = \frac{1}{\Lambda^{3N} N!} \int \exp\left[-\beta V(\lambda) \right] d\mathbf{r}^N \tag{87}$$

Hence,

$$\left(\frac{\partial A(\lambda)}{\partial \lambda} \right) = \frac{\int \left(\partial V(\lambda) / \partial \lambda \right) \exp\left[-\beta V(\lambda) \right] d\mathbf{r}^N}{\int \exp\left[-\beta V(\lambda) \right] d\mathbf{r}^N} = \left\langle \frac{\partial V(\lambda)}{\partial \lambda} \right\rangle_\lambda \tag{88}$$

Integrating gives

$$A(\lambda = 1) = A(\lambda = 0) + \int_{\lambda=0}^{\lambda=1} \left\langle \frac{\partial V(\lambda)}{\partial \lambda)} \right\rangle_{\lambda} d\lambda \tag{89}$$

In practice, simulations at a number of different values of λ are performed and the above equation integrated numerically.

n.b. the path must be reversible, i.e. not cross a phase transition line.

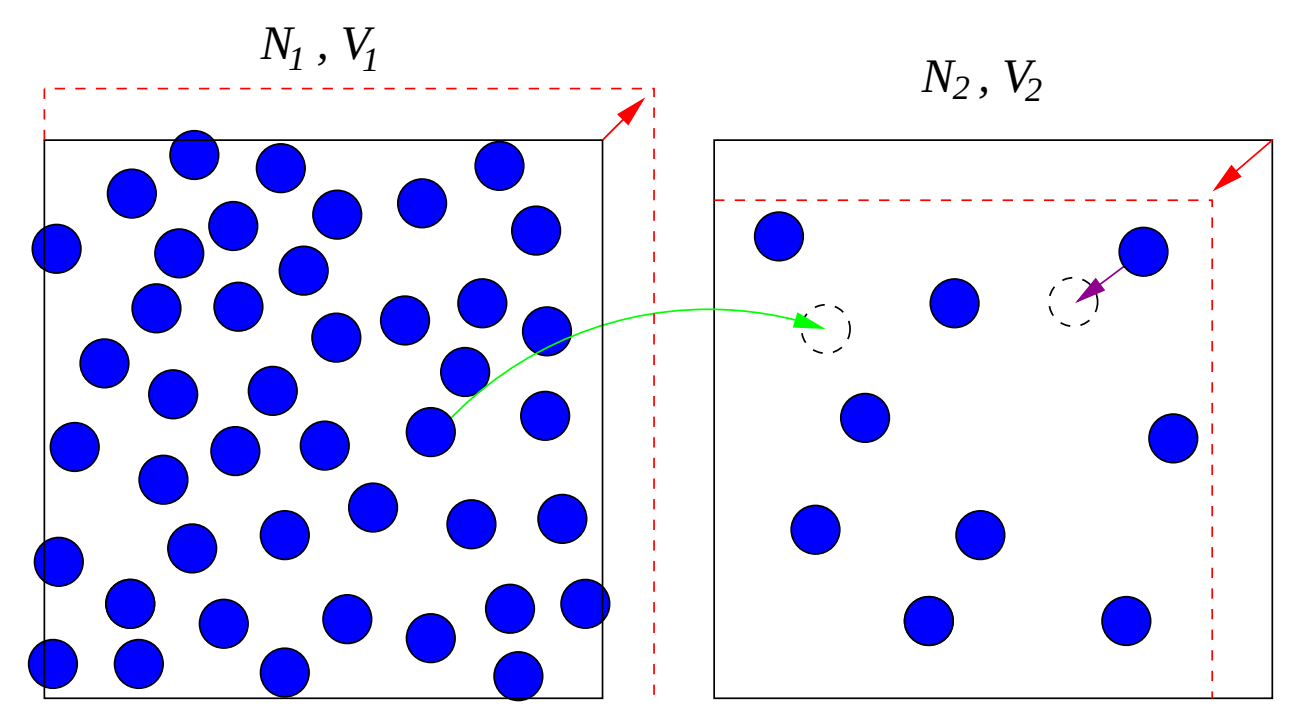
To locate a single point on the phase boundary, need to compute the free energies of both phase at a number of different state points, and then locate where they cross. i.e. lots of work!

Gibbs Ensemble

Direct simulation of phase coexistence is hindered by the problems of nucleation and the role of the interface in a small system.

One way round this is the Gibbs Ensemble introduced by Panagiotopoulos, where separate boxes contain the two phases and particles and volume can be exchanged between the two boxes.

Gives thermodynamic contact without physical contact.



Standard method for liquid-vapour coexistence.

Partition function for this 'Gibbs ensemble'

$$Q_G(N,V,T) = \sum_{N_1=0}^{N} \int_0^V Q_1(N_1,V_1,T)\, Q_2(N-N_1,V-V_1,T) dV_1 \quad (90)$$

$$= \sum_{N_1=0}^{N} \frac{1}{\Lambda^{3N} N_1!(N-N_1)!} \int_0^V V_1^{N_1} (V-V_1)^{N-N_1} dV_1$$

$$\times \int \exp\left[-\beta \mathcal{V}\left(\mathbf{s}^{N_1}\right)\right] d\mathbf{s}^{N_1} \int \exp\left[-\beta \mathcal{V}\left(\mathbf{s}^{N-N_1}\right)\right] d\mathbf{s}^{N-N_1} \quad (91)$$

Hence the distribution we wish to sample is:

$$P(N_1, V_1, \mathbf{s}_1^N, \mathbf{s}^{N-N_1}) \propto \frac{V_1^{N_1}(V-V_1)^{N-N_1}}{N_1!(N-N_1)!} \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{s}^{N_1}\right) + \mathcal{V}\left(\mathbf{s}^{N-N_1}\right)\right)\right] \quad (92)$$

Types of move:

- random particle displacement
- transfer of a randomly selected particle
- coupled volume change of the boxes (total volume constant)

Standard Metropolis acceptance criterion for particle displacements.

Volume move:

$$acc(o \rightarrow n) = \min \left\{ 1, \frac{\left(V_1^n\right)^{N_1} \left(V - V_1^n\right)^{N-1}}{\left(V_1^o\right)^{N_1} \left(V - V_1^o\right)^{N-1}} \exp \left[-\beta \left(\mathcal{V}\left(\mathbf{s}_n^N\right) - \mathcal{V}\left(\mathbf{s}_o^N\right) \right) \right] \right\} \tag{93}$$

Particle move from 1 to 2:

$$acc(o \rightarrow n) = \min \left\{ 1, \frac{N_1(V - V_1)}{(N - N_1 + 1)V_1} \exp \left[-\beta \left(\mathcal{V}\left(\mathbf{s}_n^N\right) - \mathcal{V}\left(\mathbf{s}_o^N\right) \right) \right] \right\} \tag{94}$$

nb: box selected at random, then a particle from that box randomly selected.

Problems:

- Particle insertion slow when one system dense (worse for liquid as T is decreased)
- Identity of the boxes can change close to the critical point
- Magnitude of fluctuations and correlation lengths increase close to the critical point

Tracing coexistence curves

Idea:

- First find one point on the coexistence curve.
- Then trace the curve by integrating the Clausius Clapeyron equation:

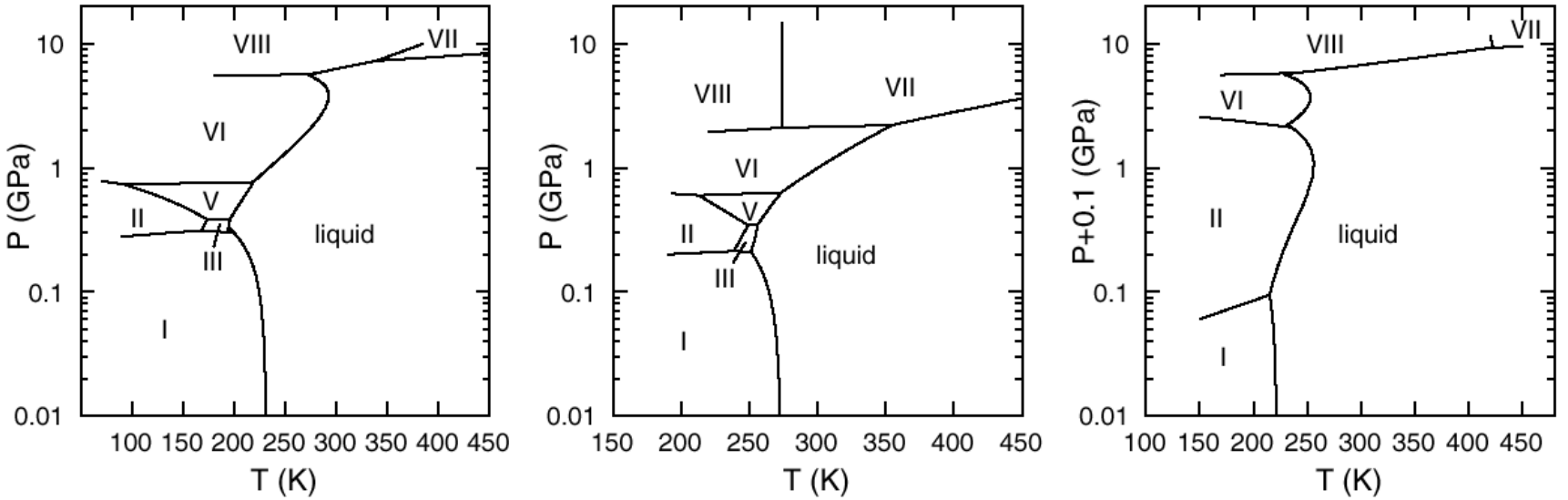
$$\frac{dP}{dT} = \frac{\Delta H}{T \Delta V} \tag{95}$$

n.b. All these quantities can be calculated direct from simulation without further free energy calculations.

Method often called Gibbs-Duhem integration.

Analogous equations can be derived for tracing the phase boundaries as a function of other variables, e.g. the parameters of the intermolecular potential.

Water phase diagrams



Experimental water phase diagram (middle) compared to that for TIP4P (left) and SPC/E (right) water models.

Clever tricks to speed up simulations

Free energy barriers that are significantly larger than kT make sampling an equilibrium distribution with standard simulation techniques tricky.

Possible solutions:

- Methods to enhance barrier crossings.
- Cleverly chosen non-local moves that take much bigger steps in configuration space.

Parallel tempering

Barriers that are prohibitively high at low temperature can be easily crossed at sufficiently high temperature.

Basic idea: maintain equilibrium at low temperature by coupling runs to higher temperature through the occasional swapping of configurations between parallel simulations at different temperatures.

The partition function for this extended system of M coupled simulations is

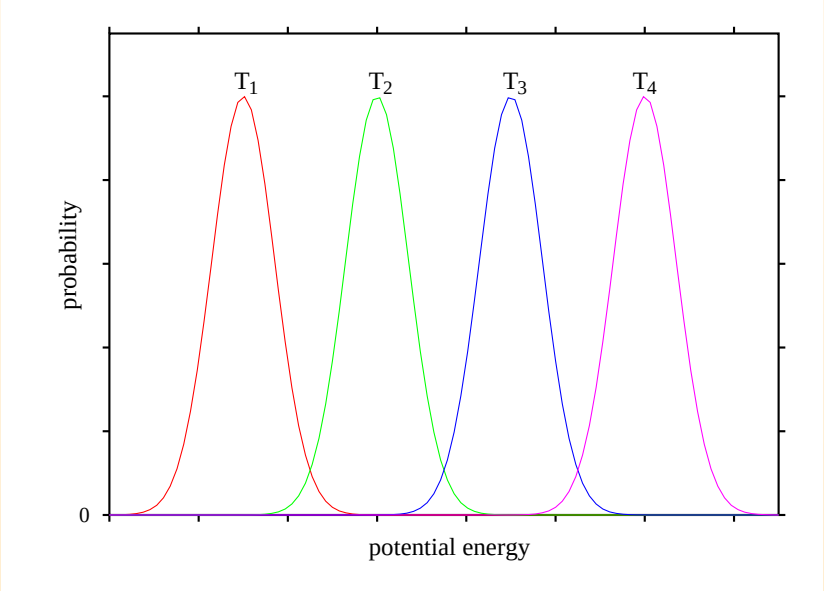
$$Q_{extended}\left(N,V,\{T_k\}\right)=\prod_{k=1}^{M}Q(N,V,T_k)=\prod_{k=1}^{M}\frac{1}{\Lambda_k^{3N}N!}\int\exp\left[-\beta_k\mathcal{V}\left(\mathbf{r}_k^N\right)\right]d\mathbf{r}_k^N \tag{96}$$

For the swapping moves, chose $\alpha(o\rightarrow n)$ to be symmetric. E.g. by randomly choosing two of the M systems, and then switching their temperatures.

The acceptance criterion must satisfy:

$$\frac{acc\left(\left\{\left(\mathbf{r}_a^N, T_a\right), \left(\mathbf{r}_b^N, T_b\right)\right\} \rightarrow \left\{\left(\mathbf{r}_b^N, T_a\right), \left(\mathbf{r}_a^N, T_b\right)\right\}\right)}{acc\left(\left\{\left(\mathbf{r}_b^N, T_a\right), \left(\mathbf{r}_a^N, T_b\right)\right\} \rightarrow \left\{\left(\mathbf{r}_a^N, T_a\right), \left(\mathbf{r}_b^N, T_b\right)\right\}\right)} = \frac{P(\left(\mathbf{r}_b^N, T_a\right) P(\left(\mathbf{r}_a^N, T_b\right)}{P((\mathbf{r}_a^N, T_a) P(\left(\mathbf{r}_b^N, T_b\right)}$$

$$= \frac{\exp\left[-\beta_a \mathcal{V}\left(\mathbf{r}_b^N\right) - \beta_b \mathcal{V}\left(\mathbf{r}_a^N\right)\right]}{\exp\left[-\beta_a \mathcal{V}\left(\mathbf{r}_a^N\right) - \beta_b \mathcal{V}\left(\mathbf{r}_b^N\right)\right]} = \exp\left[\left(\beta_a - \beta_b\right)\left(\mathcal{V}\left(\mathbf{r}_a^N\right) - \mathcal{V}\left(\mathbf{r}_b^N\right)\right)\right] \quad (97)$$



The probability of a switching move will be very low unless the potential energy distributions for the two temperatures being switched have sufficient overlap. ∴ choice of temperatures important.

Umbrella sampling

Basic idea: Bias the system to sample configurations near the top of the barrier by modifying the Hamiltonian.

Need to know the reaction coordinate/order parameter, Q, for which the free energy barrier is a function.

The free energy as a function of Q is defined as

$$A(Q') = -kT \log \int \delta\left(Q' - Q\left(\mathbf{r}^N\right)\right) \exp\left[-\beta \mathcal{V}\left(\mathbf{r}^N\right)\right] d\mathbf{r}^N \tag{98}$$

and is related to $p(Q)$ by

$$A(Q) = A - kT \log p(Q) \tag{99}$$

$p(Q)$ can then be related to the probability distribution in a biased non-

Boltzmann simulation with an effective potential $\mathcal{V} + W$.

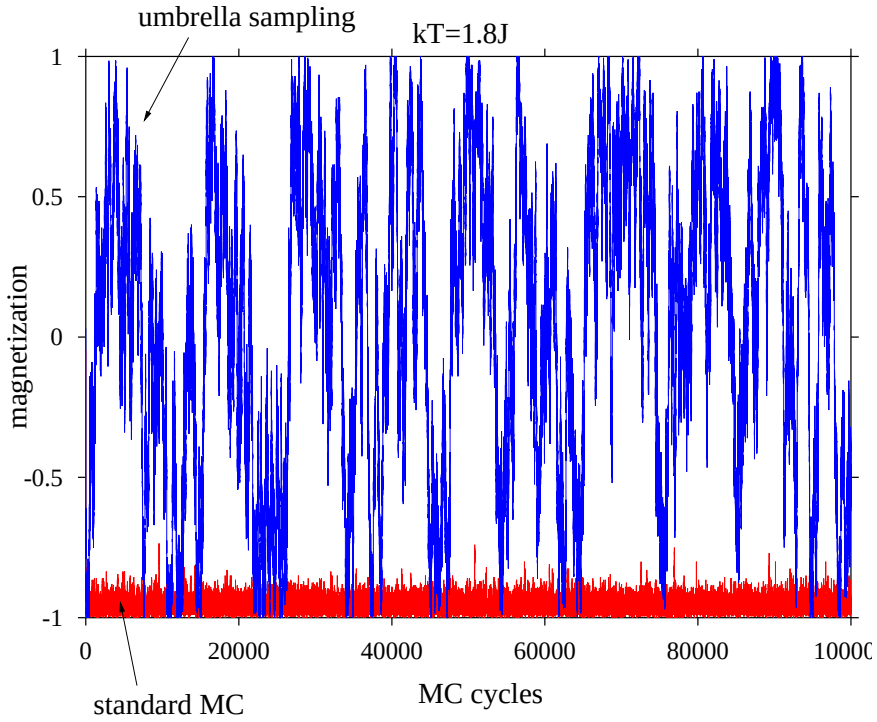
$$\begin{aligned} p(Q') &= \frac{\exp[\beta W(Q')]}{Z} \int \delta\left(Q' - Q\left(\mathbf{r}^N\right)\right) \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r}^N\right) + W(Q)\right)\right] d\mathbf{r}^N \\ &= \frac{Z_W}{Z} \exp[\beta W(Q')] p_W(Q') \end{aligned} \tag{100}$$

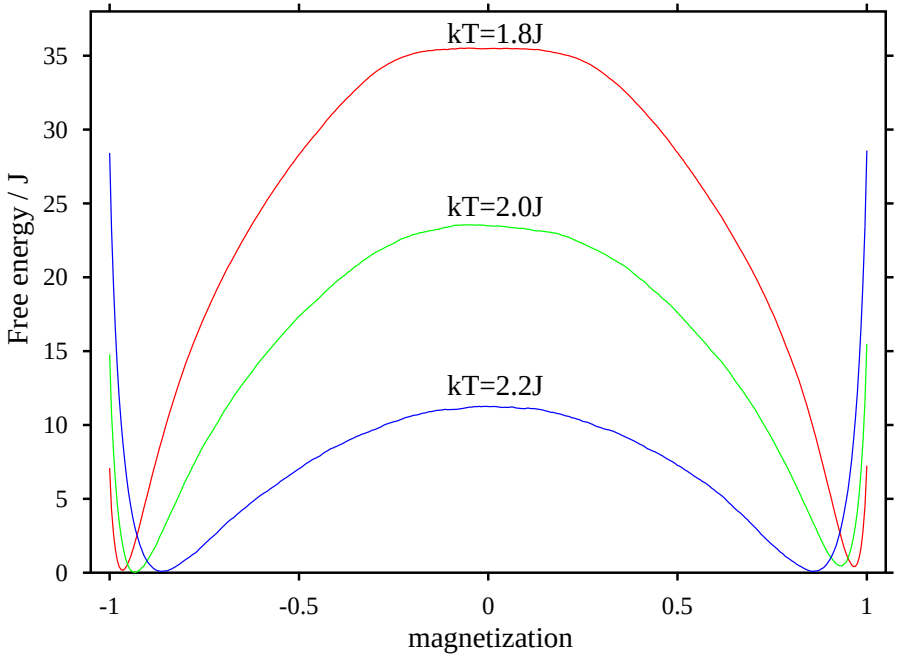
The ideal choice for the biasing distribution is $W(Q) = -A(Q)$ since then $p_W(Q)$ is constant.

In practice, only need a reasonable estimate to sufficiently facilitate passage between the free energy minima. It can be obtained iteratively.

$$\begin{aligned} \frac{Z}{Z_W} &= \frac{\int \exp\left[-\beta\mathcal{V}\left(\mathbf{r}^N\right)\right] d\mathbf{r}^N}{\int \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r}^N\right) + W(Q)\right)\right] d\mathbf{r}^N} \\ &= \frac{\int \exp[\beta W(Q)] \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r}^N\right) + W(Q)\right)\right] d\mathbf{r}^N}{\int \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r}^N\right) + W(Q)\right)\right] d\mathbf{r}^N} \\ &= \langle \exp[\beta W(Q)] \rangle_W \end{aligned} \tag{101}$$

$$
\begin{aligned}
\langle A \rangle &= \frac{1}{Z} \int A \exp\left[-\beta \mathcal{V}\left(\mathbf{r}^N\right)\right] d\mathbf{r}^N \\
&= \frac{Z_W}{Z} \frac{1}{Z_W} \int A \exp[\beta W(Q)] \exp\left[-\beta \left(\mathcal{V}\left(\mathbf{r}^N\right) + W(Q)\right)\right] d\mathbf{r}^N \\
&= \frac{\langle A \exp[\beta W(Q)] \rangle_W}{\langle \exp[\beta W(Q)] \rangle_W}
\end{aligned}
\tag{102}
$$



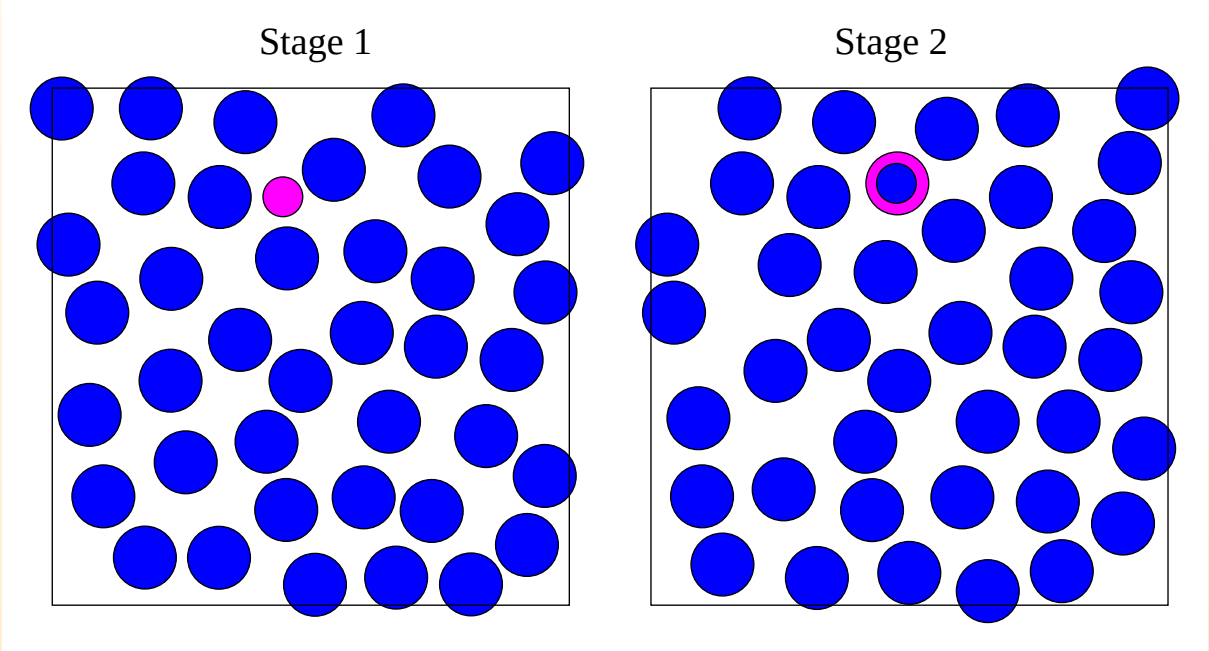


2D Ising model: 20×20 *lattice.*

Widom insertion and multiple staging

In the Widom insertion method, the probability of insertion is very low for dense configurations.

E.g. For hard spheres: One way to improve is to use insertion of a small sphere as an intermediate.



Stage 1: Measure fraction of time that small sphere finds no overlap.

Stage 2: Measure fraction of time that there is no overlap when small sphere is grown to full size.

Biased moves

If techniques such as Widom insertion can be problematic for dense spheres, problems much greater for complex molecules such as polymers.

Need more sophisticated means of generating configurations.

Use general Metropolis recipe, but do not choose α to be symmetric.

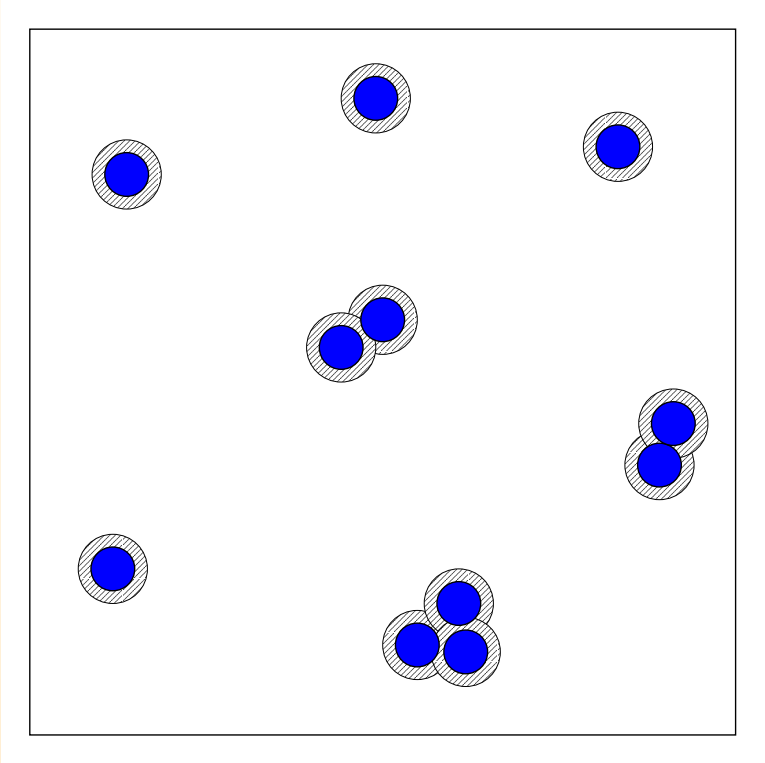
I.e. $acc(o \to n) = \min\{1, \chi\}$ and $acc(n \to o) = \min\{1, 1/\chi\}$, where

$$\chi = \frac{P_n \, \alpha(n \to o)}{P_o \, \alpha(o \to n)} \tag{103}$$

E.g. For polymers ‘‘configurational bias Monte Carlo’’ involves regrowing a section of polymer at each MC step. At each step in the regrowth process a number of directions for the new chain are explored, and the direction selected is biased towards those with lower energy.

Association Bias Monte Carlo

In a low density system where the interparticle attraction is strong, the system will typically form clusters.



Standard MC with random displacements inefficiently samples phase space, both because the acceptance probability for break-up of the clusters will be low, and because most moves for the free particles will give lead to free diffusion and not association.

Adding cluster association moves and cluster breakup moves could accelerate convergence.

Cluster association moves:

- Choose a random particle (probability = $1/N$)
- Move this particle from $\mathbf{r}_o$ to a random trial position $\mathbf{r}_n$ in V_a, where V_a is the union of all the bonding regions around each particle. (probability = $1/V_a$)

$$\alpha_a(o \to n) = \frac{1}{NV_a} \tag{104}$$

Cluster breakup moves:

- Choose a random particle from those that are involved in clusters (probability = $1/N_a$)
- Move this particle from $\mathbf{r}_o$ to a random trial position $\mathbf{r}_n$ (probability = $1/V$)

$$\alpha_b(n \to o) = \frac{1}{N_a V} \tag{105}$$

If the probability of performing association and break-up moves are equal, then the acceptance criteria for cluster association follows from

$$\chi = \frac{NV_a}{N_a V} \exp\left[-\beta\left(\mathcal{V}\left(\mathbf{r_n}^N\right) - \mathcal{V}\left(\mathbf{r_o}^N\right)\right)\right] \tag{106}$$

The Boltzmann factor will tend to favour association, but the prefactor the reverse.

In practice determining V_a is not straightforward, but methods are available.

Molecules

If work with a fully atomistic representation, then can simply treat the atoms within a molecule in the same way as any other atom.

One practical difficulty with this approach is that the bonds within a molecule are much stiffer than the interactions between non-bonded atoms.

In standard molecular dynamics, these stiff interactions will determine the time step, and constrain it to be much smaller than is necessary for the intermolecular interactions. $\therefore$ inefficient.

Other issues: Volume scaling moves in MC need to be reformulated. E.g. scale the centre of mass coordinates.

Rigid molecules

Removes the times scale separation between intra- and intermolecular degrees of freedom.

Each molecule now has three degrees of freedom associated with the position of the centre of mass, and two (linear) or three (non-linear) degrees of freedom associated with the orientation of the molecule.

Orientational degrees of freedom are most easily handled using quaternions (not Euler angles).

In MD angular velocities and accelerations need to be incorporated.

In MC, need rotational moves as well as translational moves. Need to be generated uniformly.

Partially rigid molecules

For non-rigid molecules, such as polymers, some internal degrees of freedom are much stiffer than others. e.g. bond stretches compared to bond torsions.

Very stiff coordinates can be treated as rigid. E.g. constrain some bond lengths to be constant.

In MD, these constraints can be implemented by algorithms, such as SHAKE.

However, such constraints do change the system, as now no longer any kinetic energy in that degree of freedom.

A more recent alternative is to use a multiple time step integration algorithm, e.g. RESPA (reference system propagator algorithm).

They take multiple steps in the fast degrees of freedom (short-range forces) for every step in the slow degrees of freedom (long-range forces).

Constant temperature Molecular Dynamics

Two ways:

1) Introduce stochastic collisions with a ''heat bath'' to thermalize the system.

2) Introduce a deterministic 'non-Newtonian' dynamics that generates the correct ensemble.

Andersen Thermostat

Coupling to the heat bath is represented by stochastic impulsive forces that occasionally act on randomly selected particles.

These stochastic collisions move the system between different constant energy shells.

Strength of coupling controlled by ν, the frequency of stochastic collisions.

Algorithm:

- integrate the equations of motion: $\mathbf{r}^N(t) \rightarrow \mathbf{r}^N(t + \Delta t)$
- Check whether any particles are selected to undergo a collision with the heat bath. The probability that a particle is selected is $\nu \Delta t$.
- If a particles has been selected to undergo a collision, its new velocity is drawn from a Maxwell-Boltzmann distribution.

Pros: Simple,

Cons: The dynamics is not so realistic.

The Lagrangian

The Lagrangian formulation of classical mechanics is based on a variational principle.

The trajectory followed by a system between two points is the one that minimizes the classical action S, where

$$S = \int_{t_i}^{t_f} \mathcal{L}(q, \dot{q}) dt \tag{107}$$

and the Lagrangian is

$$\mathcal{L}(q, \dot{q}) = \mathcal{K}(\dot{q}) - \mathcal{V}(q) \tag{108}$$

It can be shown that the action is minimized when

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}}\right) = \frac{\partial \mathcal{L}(q, \dot{q})}{\partial q} \tag{109}$$

This is the Lagrangian equation of motion.

We can write this in terms of a generalized momentum

$$p = \frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}} \tag{110}$$

Hence

$$\dot{p} = \frac{\partial \mathcal{L}(q, \dot{q})}{\partial q} \tag{111}$$

As a check, for standard Cartesian coordinates

$$\mathcal{L}(x, \dot{x}) = \frac{1}{2} m \dot{x}^2 - \mathcal{V}(x) \tag{112}$$

$$p = \frac{\partial \mathcal{L}(x, \dot{x})}{\partial \dot{x}} = m\dot{x} \tag{113}$$

and the equation of motion is

$$m\ddot{x} = -\frac{\partial \mathcal{V}(x)}{\partial x} \tag{114}$$

Hamiltonian mechanics

The Hamiltonian is related to the Lagrangian by

$$\mathcal{H}(q,p) = p\dot{q} - \mathcal{L}(q,\dot{q},t) \tag{115}$$

To derive the equations of motion in terms of $\mathcal{H}$

$$d\mathcal{H}(q,p) = \frac{\partial \mathcal{H}}{\partial p}dp + \frac{\partial \mathcal{H}}{\partial q}dq + \frac{\partial \mathcal{H}}{\partial t}dt \tag{116}$$

$$\begin{aligned} d\mathcal{H}(q,p) &= d(p\dot{q}) - d\mathcal{L}(q,\dot{q},t) \\ &= p d\dot{q} + \dot{q}dp - \left[\frac{\partial \mathcal{L}}{\partial q}dq + \frac{\partial \mathcal{L}}{\partial \dot{q}}d\dot{q} + \frac{\partial \mathcal{L}}{\partial t}dt\right] \\ &= p d\dot{q} + \dot{q}dp - \dot{p}dq - p d\dot{q} - \frac{\partial \mathcal{L}}{\partial t}dt \\ &= \dot{q}dp - \dot{p}dq - \frac{\partial \mathcal{L}}{\partial t}dt \end{aligned} \tag{117}$$

Comparing the above equations gives:

$$\frac{\partial \mathcal{H}}{\partial p} = \dot{q} \tag{118}$$

$$\frac{\partial \mathcal{H}}{\partial q} = -\dot{p} \tag{119}$$

When the Lagrangian does not explicitly depend on time, the Hamiltonian is conserved

$$\frac{d\mathcal{H}(q,p)}{dt} = \frac{\partial \mathcal{H}}{\partial p}\dot{p} + \frac{\partial \mathcal{H}}{\partial q}\dot{q} = -\frac{\partial \mathcal{H}}{\partial p}\frac{\partial \mathcal{H}}{\partial q} + \frac{\partial \mathcal{H}}{\partial q}\frac{\partial \mathcal{H}}{\partial p} = 0 \tag{120}$$

and expresses the conservation of energy.

In Cartesians

$$\mathcal{H}(x,p_x) = p_x\dot{x} - \mathcal{L}(x,\dot{x}) = \frac{p_x^2}{2m} + \mathcal{V}(x) \tag{121}$$

and the equations of motion are

$$\dot{x} = \frac{\partial \mathcal{H}}{\partial p_x} = \frac{p_x}{m} \tag{122}$$

$$\dot{p}_x = -\frac{\partial \mathcal{H}}{\partial x} = -\frac{\partial \mathcal{V}(x)}{\partial x} \tag{123}$$

Note, that this formulation gives the equations of motion in terms of two first-order differential equations, rather than a single second-order differential equation.

Nose-Hoover thermostat

We use an extended Lagrangian, where the extra degree of freedom acts as a reservoir and allows energy to flow back and forth between the system and the reservoir.

$$\mathcal{L}_{Nose} = \sum_{i=1}^{N} \frac{m_i}{2} s^2 \dot{\mathbf{r}}_i^2 - \mathcal{V}\left(\mathbf{r}^N\right) + \frac{Q}{2}\dot{s}^2 - \frac{L}{\beta}\log(s) \tag{124}$$

L is a parameter and Q is an effective mass associated with s that controls the rate of temperature fluctuations. The momenta are

$$\mathbf{p}_i = \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{r}}_i} = m_i s^2 \dot{\mathbf{r}}_i \tag{125}$$

$$p_s = \frac{\partial \mathcal{L}}{\partial \dot{s}} = Q\dot{s} \tag{126}$$

The Hamiltonian of this extended system is

$$\mathcal{H}_{Nose} = \sum_{i=1}^{N} \frac{\mathbf{p}_i^2}{2m_i s^2} + \mathcal{V}\left(r^N\right) + \frac{p_s^2}{2Q} + \frac{L}{\beta}\log(s) \tag{127}$$

The partition function of this system of $6N+2$ degrees of freedom in the microcanonical ensemble is

$$Q_{Nose} = \frac{1}{N!}\int \delta\left(E - \mathcal{H}_{Nose}\right) dp_s\, ds\, d\mathbf{r}^N d\mathbf{p}^N \tag{128}$$

It can be shown that this is proportional to the canonical partition function when $L = 3N+1$.

The equations of motion are

$$\frac{d\mathbf{r}_i}{dt} = \frac{\partial \mathcal{H}_{Nose}}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i s^2} \tag{129}$$

$$\frac{d\mathbf{p}_i}{dt} = -\frac{\partial \mathcal{H}_{Nose}}{\partial \mathbf{r}_i} = \frac{\partial \mathcal{V}\left(\mathbf{r}^N\right)}{\partial \mathbf{r}_i} \tag{130}$$

$$\frac{ds}{dt} = \frac{\partial \mathcal{H}_{Nose}}{\partial p_s} = \frac{p_s}{Q} \tag{131}$$

$$\frac{dp_s}{dt} = -\frac{\partial \mathcal{H}_{Nose}}{\partial s} = \frac{1}{s}\left(\sum_{i=1}^{N}\frac{\mathbf{p}_i^2}{2m_i s^2} - \frac{L}{\beta}\right) \tag{132}$$

These equations can be reformulated in terms of the real momenta, $\mathbf{p}'_i = \mathbf{p}_i/s$ and time $t' = t/s$, and a friction coefficient $\xi = p_s/Q$.

Constant pressure molecular dynamics

An extended Lagrangian method can also be introduced to generate trajectories in the isobaric (NPT) ensemble in an analogous way.

It involves the introduction of a kinetic energy term $Q_V \dot{V}^2/2$ and a term in the potential PV. The momentum associated with the volume coordinate is $p_V = Q_V \dot{V}$.

Q_V again plays the role of an effective mass regulating the fluctuations in volume.

(In practice, as with constant pressure MC, better to work in $\log V$).

Action and the Verlet algorithm

Classical trajectories correspond to ones where the action S is a minimum.

For a trajectory of discrete steps (e.g. as generated by MD) one could instead try to minimize a discretized version of the action.

$$S_{disc} = \sum_{i_i}^{i_f - 1} \Delta t \left[\frac{1}{2} m \left(\frac{x_{i+1} - x_i}{\Delta t} \right)^2 - \mathcal{V}(x_i) \right] \tag{133}$$

Minimizing this gives:

$$m \left(\frac{2x_i - x_{i+1} - x_{i-1}}{\Delta t} \right) - \Delta_t \frac{\partial \mathcal{V}(x_i)}{\partial x_i} = 0 \tag{134}$$

which on rearranging gives the Verlet algorithm

$$x_{i+1} = 2x_i - x_{i-1} - \frac{\Delta t^2}{m} \left(\frac{\partial \mathcal{V}(x_i)}{\partial x_i} \right) \tag{135}$$

'Car-Parrinello' Approaches

For certain systems, evaluation of the energy requires an optimization of internal degrees of freedom (e.g. electrons, induced dipoles of polarizable atoms) of the atoms.

As the time scales associated with these internal degrees of freedom are much faster than those associated with the nuclear motion, it is assumed that they respond adiabatically to changes in the positions of the atoms. i.e. the electrons are always in their ground state.

Thus, in a standard molecular dynamics one would need to perform an iterative optimization at each time step.

Problems:

- Computationally expensive
- Incomplete convergence leads to the exertion of a drag on the nuclei.

Car-Parrinello molecular dynamics treats the electrons using density functional theory in the local density approximation.

It circumvents these problems using an extended Lagrangian that includes the electronic density. I.e. there are equations of motion for the electron density that are integrated in an analogous way to those for the nuclear coordinates.

Although the electron density is not always optimal - instead it fluctuates around this optimal value - it does not exert any systematic drag on the nuclei.

Optimization of the electron density occurs 'on-the-fly', rather than by iteration at each step.

Note that although CPMD computes the potential energy at each point using electronic density functional theory (rather than using a potential function) the nuclear motion is still treated classically.

Application to polarizable atoms

For a system of polarizable atoms (say with permanent dipoles), the total potential energy can be written as $\mathcal{V} = \mathcal{V}_0 + \mathcal{V}_{ind}$ the induction energy is given by

$$\mathcal{V}_{ind} = -\sum_i \mathbf{E}_i \cdot \mu_i + \frac{1}{2\alpha}\sum_i \mu_i^2 \tag{136}$$

where μ_i is the induced dipole on atom i, $\mathbf{E}_i$ is the electric field at atom i due to the other atoms, and α is the polarizability.

Minimization of $\mathcal{V}_{ind}$ with respect to the μ_i implies that the induced dipoles must satisfy

$$\mu_i = \alpha \mathbf{E}_i \tag{137}$$

Calculation of these induced dipoles, and hence the induction energy, requires the solution of these 3N linear equations, typically by iteration to self-consistency.

To avoid this iteration in an MD simulation, one can use the extended

Lagrangian

$$\mathcal{L}\left(\mathbf{r}^N, \mu^N\right) = \frac{1}{2}\sum_i m\dot{\mathbf{r}}_i^2 + \frac{1}{2}\sum_i M\dot{\mu}_i^2 - \mathcal{V}\left(\mathbf{r}^N, \mu^N\right) \tag{138}$$

where M is the inertial 'mass' associated with the dipoles.

The equations of motion for the dipoles is then

$$M\ddot{\mu}_i = \frac{\partial \mathcal{L}}{\partial \mu_i} = \mathbf{E}_i - \frac{\mu_i}{\alpha} \tag{139}$$

The RHS of this equation represents the forces on the dipoles and is zero when they take their optimal value (Eq. 137).

Writing $\mu_i = \mu_i^{opt} + \Delta\mu_i$, gives $M\ddot{\mu}_i = -\Delta\mu_i/\alpha$.

Therefore, the induced dipoles will oscillate around their optimal values.

To ensure that the dipoles respond rapidly to changes in the atomic positions: $M \ll m$

To ensure that the dipoles only exhibit small oscillations about their optimal value: $T_\mu \ll T_r$ (temperatures defined through kinetic energies)

Latter achieved by two Nosé-Hoover thermostats.

Other quantum simulation methods

Other simulation methods that attempt to solve the Schrodinger equation:

- Variational Monte Carlo
  A trial wave-function is optimized according to the variational principle using MC.
- Diffusion Monte Carlo
  The Schrodinger equation is rewritten as a diffusion equation in imaginary time, which is then solved using a set of random walkers.
- Path-integral Monte Carlo
  Based on the path integral formulation of quantum mechanics. Exploits an isomorphism between quantum mechanics and the statistical mechanics of ring polymers.

Coarse-grained simulations

So far, mainly considered atomic and molecular systems.

But can apply simulation methods to systems where (many) degrees of freedom have been coarse-grained, and where the interactions are described by effective potentials.

E.g. Under certain conditions colloidal particles can behave like hard spheres.

Can apply Monte Carlo techniques straightforwardly to the coarse-grained systems.

However, the systems may no longer follow Newton's equations of motion at this coarse-grained level.

E.g. colloidal particles undergo Brownian motion due to collisions with the solvent (which in the coarse-grained description is usually 'integrated out').

Langevin equation more appropriate:

$$m\ddot{\mathbf{r}} = F(\mathbf{r}) - \beta\mathbf{v} + \eta(t) \tag{140}$$

- $-\beta\mathbf{v}$ is a viscous drag force due to motion through the solvent.
- $\eta(t)$ is a random force that represents the effects of collisions with the solvent.

Brownian dynamics is a simulation method that generates trajectories that obey this stochastic differential equation.