Summary of SI2530 Computational Physics

Yashar Honarmandi
yasharh@kth.se

27 september 2021

Sammanfattning

Innehåll

1 Basic Concepts

Molecular Dynamics Molecular dynamics is a class of simulation based on simulating a collection of atoms as a set of material points which move according to Newton's laws and consider only forces that arise as interactions between two atoms.

Equilibration Equilibration is the process of using some scheme that fixes a certain set of quantities and running a molecular dynamics simulation until these quantities have stabilized to the desired value.

The Radial Distribution Function The radial distribution function is defined as

$$g(r) = \frac{1}{\rho} \left. \frac{\mathrm{d}N}{\mathrm{d}V} \right|_r .$$

Verlet's Method Euler's method does not respect the laws of physics. Instead we consider a different scheme, based on the Taylor expansions

$$x_{n\pm1} = x_n \pm v_n\tau + \frac{1}{2}a_n\tau^2.$$

This nets two equations

$$v_n = \frac{x_{n+1} - x_{n-1}}{2\tau}, \; x_{n+1} = 2x_n - x_{n-1} + a_n\tau^2.$$

This integration scheme is called Verlet's scheme, or the leapfrog method. Note that it is not self-starting. The way that is usually solved is through integrating a single step from the initial conditions using Euler's method and proceeding with the Verlet scheme from there.

Monte Carlo Integration Monte Carlo integration is a method for performing integration in multiple dimensions. The process is the following:

- Generate n random points in d dimensions.
- For each point, if it within the domain of integration, add it to a list.

If the domain of integration can be enclosed in a d-dimensional box of volume V, the estimate of the integral is then

$$\frac{V}{n} \sum f(x_i).$$

The Metropolis Algorithm Statistical mechanics is concerned with integrals of the form

$$\frac{\int \mathrm{d}x \, f(x)p(x)}{\int \mathrm{d}x \, p(x)}.$$

The idea of the Metropolis algorithm is to introduce a transition probability $T(x_i \to x_j)$ and evolve our initial configuration according to T such that we flow towards a sequence of points distributed according to p. A sufficient condition is the detailed balance condition

$$p(x_i)T(x_i \to x_j) = p(x_j)T(x_j \to x_i).$$

A simple choice which is often used is

$$T(x_i \to x_j) = \min(1, \frac{p(x_j)}{p(x_i)}).$$

The steps of the algorithm is then

- From a given configuration, generate a new test configuration from a uniform random number generator.
- Compute $w = \frac{p(x_j)}{p(x_i)}$.
- If $w = 1$, accept the change. Otherwise, generate a random number r. If $w > r$, accept the change.
- Update the configuration.

Random Walks