Monte Carlo

Content:

- **Monte Carlo integration**
- **Variance reduction / Importance sampling**
- **Markov chains**
- **The Metropolis algorithm**
- **Error estimate**

The Metropolis algorithm

A general method of sampling arbitrary highly-dimensional probability distributions by taking a random walk through configuration space.

- It was introduced by Metropolis *et al.* 1953 to determine the equation of state for a hard sphere liquid.

- It uses the Markov chain technique to generate configurations with knowledge of only relative probabilities, no absolute probabilities have to be known.

- The algorithm guides the Markov chain to important regions by rejecting unlikely configurations.

- The generated configurations become correlated and care has to be taken when evaluating proper error bars.

Basic idea

If you can find a matrix $\mathbf{W}$ with the following properties

1. $0 \leq w_{nm} \leq 1 \qquad \forall\; n$ and m
2. $\displaystyle\sum_{n=1}^{M} w_{nm} = 1 \qquad \forall\; m$
3. w_{nm} is ergodic
4. $w_{mn} p_n = w_{nm} p_m$

then the Markov process will, in the long run, produce states distributed according to the probability distribution $\mathbf{P}$.

The Metropolis algorithm

The Metropolis algorithm is a particular way of ensuring that the transition rule satisfies detailed balance.

The transition matrix $w_{nm} = w_{n\leftarrow m}$ is split into two parts

$$w_{nm} = \tau_{nm}\alpha_{nm}$$

where τ_{nm} is the probability of making a trial change from state Ω_m to state Ω_n and α_{nm} is the probability of accepting the trial state.

The acceptance probability is assumed to satisfy

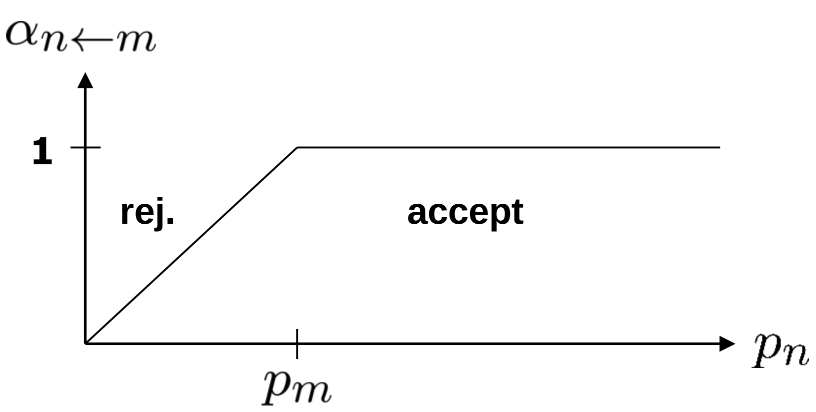
$$\alpha_{nm} = \begin{cases} 1 & \text{if} \;\; p_n \geq p_m \\ p_n/p_m & \text{if} \;\; p_n < p_m \end{cases}$$

and to ensure detailed balance the trail change then has to be symmetric

$$\tau_{nm} = \tau_{mn}$$

Moves that are not accepted are rejected and remain at the same location for at least one more step

$$w_{mm} = 1 - \sum_{n(\neq m)} w_{nm}$$



Göran Wahnström, Department of Physics, Chalmers

The Metropolis algorithm

Consider a system that can be in different states Ω_m with corresponding probabilities p_m.
Decide how to make trail changes $\tau_{nm} = \tau_{n\leftarrow m}$ and establish an initial configuration $\Omega(s=0)$.
To advance the Markov chain one step, from configuration $\Omega(s) = \Omega_m$ to $\Omega(s+1)$

1) Choose a trail state Ω_t according to τ_{tm}
2) calculate the ratio $q = p_t/p_m$
3) generate a random number ξ between 0 and 1
   if $q \geq \xi$
     accept the change and let $\Omega(s+1) = \Omega_t$
   otherwise
     count the old state once more and let $\Omega(s+1) = \Omega_m$

Repeat step 1)-3) many times.
Throw away a sufficient number of states in the beginning. Determine average quantities with proper error bars.

Monte Carlo

Content:

- Monte Carlo integration
- Variance reduction / Importance sampling
- Markov chains
- The Metropolis algorithm
- Error estimate

Error estimate

Consider the average value

$$I = \frac{1}{N}\sum_{i=1}^{N} f_i$$

We would like to determine the statistical error in the evaluation of I, *i.e.* its variance Var$[I]$. Only if $\{f_i\}$ are uncorrelated Var$[I] = \frac{1}{N}$Var$[f]$. For the general case, with correlated values, we introduce the statistical inefficiency s according to

$$\text{Var}[I] = \frac{s}{N}\text{Var}[f]$$

The number of statistically independent values of f_i is given by N/s.

Here, two different mehods of estimating s are given. The first is based on calculating the auto-correlation function, the second on block-averaging.

Error estimate

Method 1: **Correlation function**

Determine the auto-correlation function

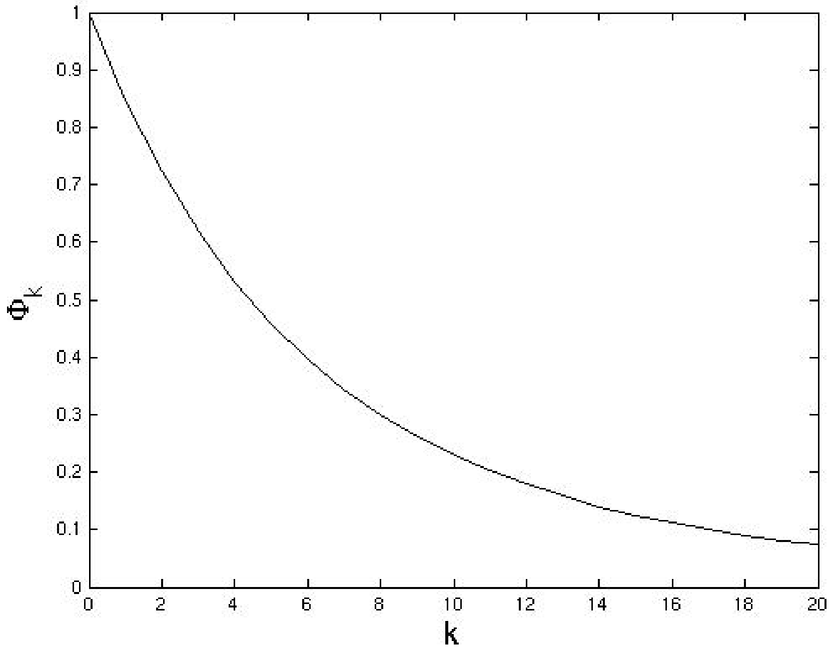
$$\Phi_k = \frac{\langle f_{i+k} f_i \rangle - \langle f_i \rangle^2}{\langle f_i^2 \rangle - \langle f_i \rangle^2}$$

where $\Phi_{-k} = \Phi_k$. Estimate s from

$$s = \sum_{k=-M}^{M} \Phi_k$$

where $s < M \ll N$. In most cases Φ_k decays essentially exponentially. You can then also estimate s from the length (k-value) when Φ_k has decayed to $\exp(-2)$, *i.e.*

$$\Phi_{k=s} = 0.135 \approx 0.1$$



Error estimate

Method 2: **Block average**

Determine the block averages

$$F_j = \frac{1}{B} \sum_{i=1}^{B} f_{i+(j-1)B}$$

and estimate s from

$$s = \lim_{B \text{ large}} \frac{B \,\mathrm{Var}[F]}{\mathrm{Var}[f]}$$