METROPOLIS SIMULATIONS IN THE CANONICAL ENSEMBLE FOR A FLUID

For a canonical ensemble

$$\pi(\bar{r}_1, \ldots, \bar{r}_N) = \frac{1}{Z} e^{-\beta U(\bar{r}_1 \ldots \bar{r}_N)}$$

We wish to define a <u>proposal</u>

The system is in a configuration $x = \{r_1 \ldots r_N\}$
We must define (choose) a new configuration y

We proceed as follows:

i) we choose one particle among the N particles

$i = 1 +$ integer part of $(N * RAN(\,))$

[we choose a random number i UNIFORMLY among $1, 2 \ldots N$]

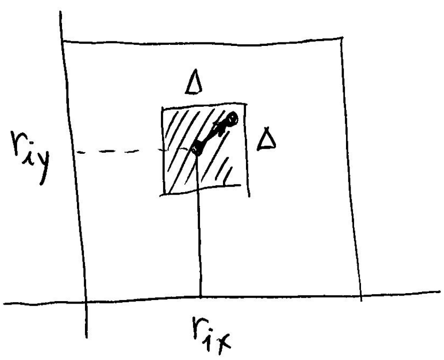
ii) we set $r'_i = (x_i, y_i, z_i)$

$$x_i = r_{ix} + \Delta(RAN(\,) - 0.5)$$
$$y_i = r_{iy} + \Delta(RAN(\,) - 0.5)$$
$$z_i = r_{iz} + \Delta(RAN(\,) - 0.5)$$

$\boxed{\Delta \text{ is a parameter}}$ TO BE FIXED

The new conf is

$$y = (r_1 \ldots r_{i-1}, r'_i, r_{i+1}, \ldots r_N)$$



IN PRACTICE:
We "move" the particle inside a cube of size Δ centered around the old position

[IGNORE THE BOUNDARIES, FOR NOW]

The probability density of this move is

$$\frac{1}{N} \times \frac{1}{\Delta^3}$$

$\uparrow$ we choose i among N numbers

$\uparrow$ we choose a point in a cube of size Δ^3

The probability density is independent of x

The probability of going from $x \longrightarrow y$ = probability of going from $y \longrightarrow x$

THE PROPOSAL MATRIX IS SYMMETRIC

<u>Acceptance</u>: $A_{xy} = \min\left(1, \frac{\pi_y}{\pi_x}\right) = \min\left(1, e^{-\beta U(y) + \beta U(x)}\right)$

Assume $U(r_1, \ldots r_N) = \sum_{i<j} V(|r_i - r_j|)$ [two-body interactions]

We need to compute $U(y) - U(x) = \Delta U$.

In the differences all terms that do not include i cancel

$$U(y) - U(x) = \sum_{j \neq i} V(|\bar{r}_i' - \bar{r}_j|) - \sum_{j \neq i} V(|\bar{r}_i - \bar{r}_j|)$$

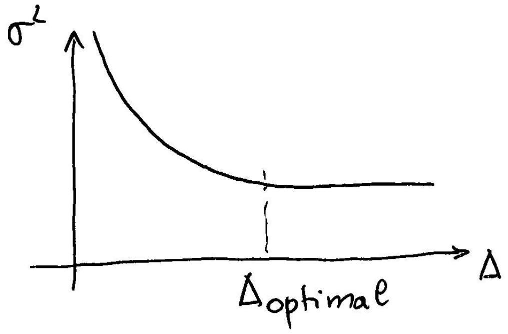
$\uparrow$ we fix i and sum over all $j \neq i$ (all particles that interact with i)

THE PRACTICAL PROBLEM:

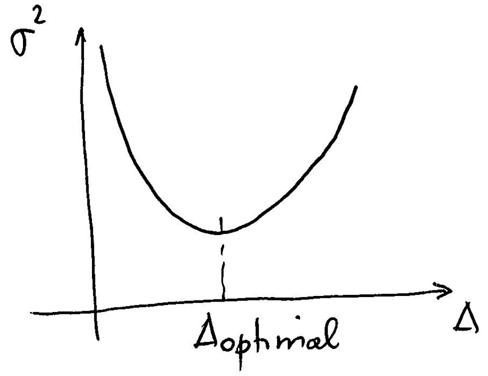
The algorithm depends on Δ.
This is a crucial parameter for the efficiency.

AIM: Choose Δ so as to obtain the smallest error for a given number N of iterations.

TYPICAL Δ dependence



rarified systems



dense systems

THE RULE OF THUMB:

Compute acceptance $f = \dfrac{\text{number of accepted moves}}{\text{total number of attempted moves}}$

Choose Δ so that $30\% < f < 50\%$

SEQUENTIAL VERSUS RANDOM UPDATE

The Metropolis update we have discussed work as follows (RANDOM UPDATE)

(a) choose a particle i (RANDOMLY)
(b) propose a movement of the particle i
(c) accept / reject the move.

Often sequential updates are used

FOR $i = 1, \dots N$ (N = number of particles
    (a) propose a movement of the particle i
    (b) accept/reject the move.
END FOR

In this sequential update, we move particles sequentially first we update particle 1, then particle 2, then particle 3, ... , up to particle N and then we start again with particle 1.

<u>BASIC RESULT</u> (it is not completely obvious):
The sequential algorithm satisfies the stationarity condition if its random version satisfies detailed balance.
The sequential algorithm does NOT satisfies detailed balance

<u>CAVEAT</u>: Sequential algorithms may have problems with ergodicity for system with discrete degrees of freedom (spin systems)

To avoid the problem one can mix random and sequential updates

EFFICIENCY OF THE UPDATES:

- For systems in which the relative position of the particles changes in the simulation (a gas, a liquid for example) sequential and random updates are EQUIVALENT.

  The sequential update is (slightly) faster as it does not require 1 random number (the one used to choose the particle)

- For spin systems, the sequential update is often more efficient, smaller errors for a given number of updates, and faster, less RNumbers. But, in this case, beware of ergodicity!