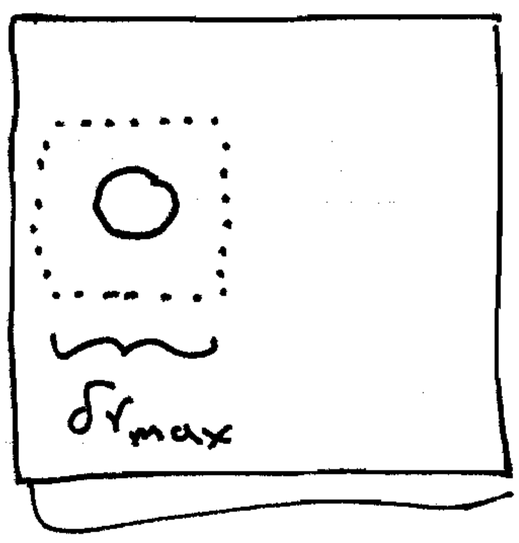
Trial Moves

Should be (a) microscopically reversible (b) ergodic and (c) efficient

Should we move one or many particles at a time?

Computational cost of a move is proportional to the number N of moved particles, but acceptance ratio is proportional to p^N ($p \equiv$ single particle acceptance ratio). Clearly single-particle moves are preferred. (Exception: cluster moves for which acceptance is *not* proportional to p^N).

For *atomic systems*, displacements in each coordinate up to δr_{max}:

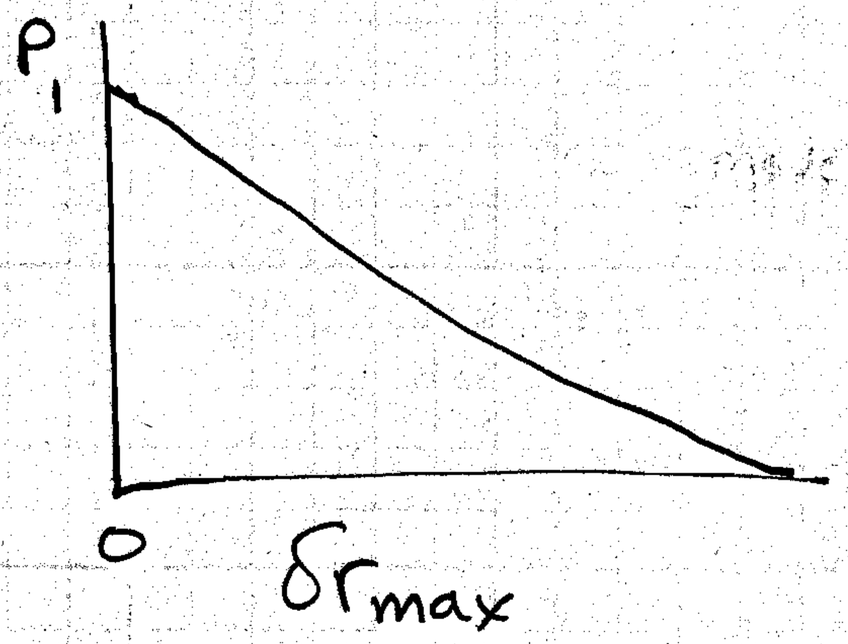


$$x_i = x_i + \delta r_{max} \cdot (rand - 0.5)$$
$$y_i = y_i + \delta r_{max} \cdot (rand - 0.5)$$
$$z_i = z_i + \delta r_{max} \cdot (rand - 0.5)$$

How do we select δr_{max}?

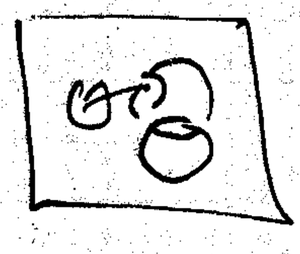
δr_{max} low → high acceptance, but particles don't go far. A common recipe is to select δr_{max} for *50%* acceptance, but many studies (e.g. Mountain + Thirumalai, Physica A 210:453 (1994)] suggest *20%*

Simple analysis: Assume a linear relationship between probability of acceptance p and δr_{max}:



The CPU cost of a
successful move t_s
unsuccessful move t_u
$t_u \leq t_s$ by construction!
$t_s / t_u = \gamma \geq 1$

Reasons for $t_u \leq t_s$:
no need for bookkeeping, also:



overlap → can be found by checking on average 1/2 of existing particles

Actual displacement per step is proportional to
$p \cdot (1-p)$

CPU time per step: $p \cdot t_s + (1-p) \cdot t_u$

$$f(p) = \frac{p \cdot (1-p)}{\gamma p + 1 - p} \qquad \frac{df}{dp} = 0 \Rightarrow -(\gamma-1) p_{opt}^2 - 2 p_{opt} + 1 = 0$$

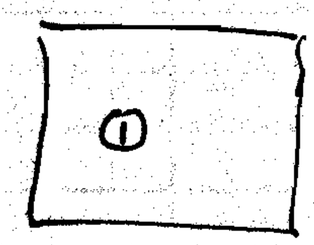
$$\Rightarrow \begin{cases} \gamma = 1 & p_{opt} = 1/2 \\ \gamma > 1 & \boxed{p_{opt} = \dfrac{\sqrt{\gamma}-1}{\gamma-1}} \end{cases}$$

γ	p_{opt}
2	41%
3	37%
10	24%

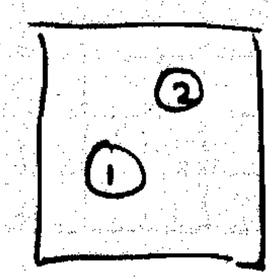
Initial Configuration

→ Random: requires repeated attempts to place particles in non-overlapping positions

Place particle 1 in random position



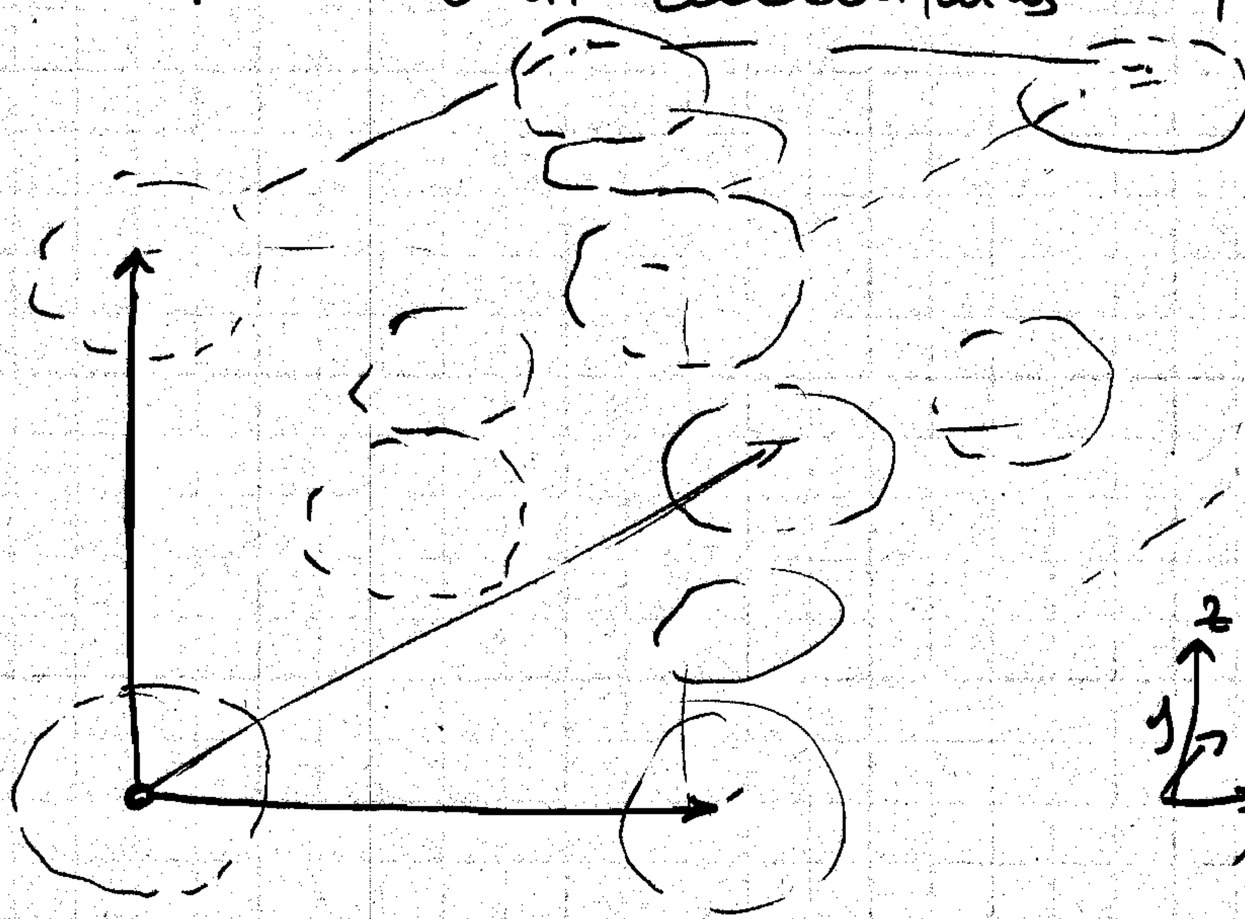
Place particle 2 at random
Check for overlaps with all existing particles



When many particles are already in the system, one may try several random positions (e.g. ~10,000) until one is found to be acceptable.
If no position can be found, discard configuration and start over.
This does not work very well for densities greater than $\rho^* \sim 0.7$. Instead, the following method is preferable:

Place on lattice

For 3-D, close packing of spheres is obtained on an fcc lattice – unit cell contains 4 particles



n	$4 \cdot n^3$
3	108
4	256
5	500
6	864
⋮	⋮

8 particles on edge of cube, each shared by 8 unit cells
$= \underline{1}$ net $(0,0,1)$
6 particles on faces, each shared by 2 cells
$= 3$ net $\left(\frac{1}{2},\frac{1}{2},1\right), \left(1,\frac{1}{2},\frac{1}{2}\right), \left(\frac{1}{2},1,\frac{1}{2}\right)$

Energy Calculation

Two types of energy calculation

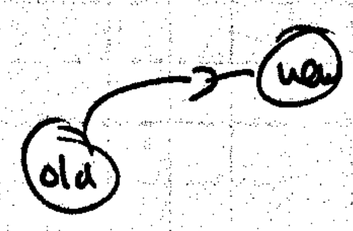
1. Calculation from "scratch" of all interactions:

$\frac{N \cdot (N-1)}{2}$ particle pairs

```
do particle1 = 1, N
   do particle2 = 1, particle1 - 1
      (calculate distance + energy)
   enddo
enddo
```

2. Calculation of "changed" interactions

$2 \times (N-1)$ particle pairs



new position: Calculate interaction with N-1 other particles
If there is overlap, get out of the loop (saves calculations)

old position: (N-1) interactions

$\Delta u = u_{new} - u_{old}$

Acceptance: If (exp(-Beta * du) > ran2(seed) then
--- accept
else
--- reject
endif

Bookkeeping on acceptance:

x(moved) = modulo(xmoved, L) (to reset coordinates within box)

⋮

u = u + du

success_steps = success_steps + 1

Output

Coordinates file + Energy at given steps

How often? Coordinates → at end of run

Energy → every N steps

Molecular Configuration Display

Rasmol (free) from www.umass.edu/microbio/rasmol

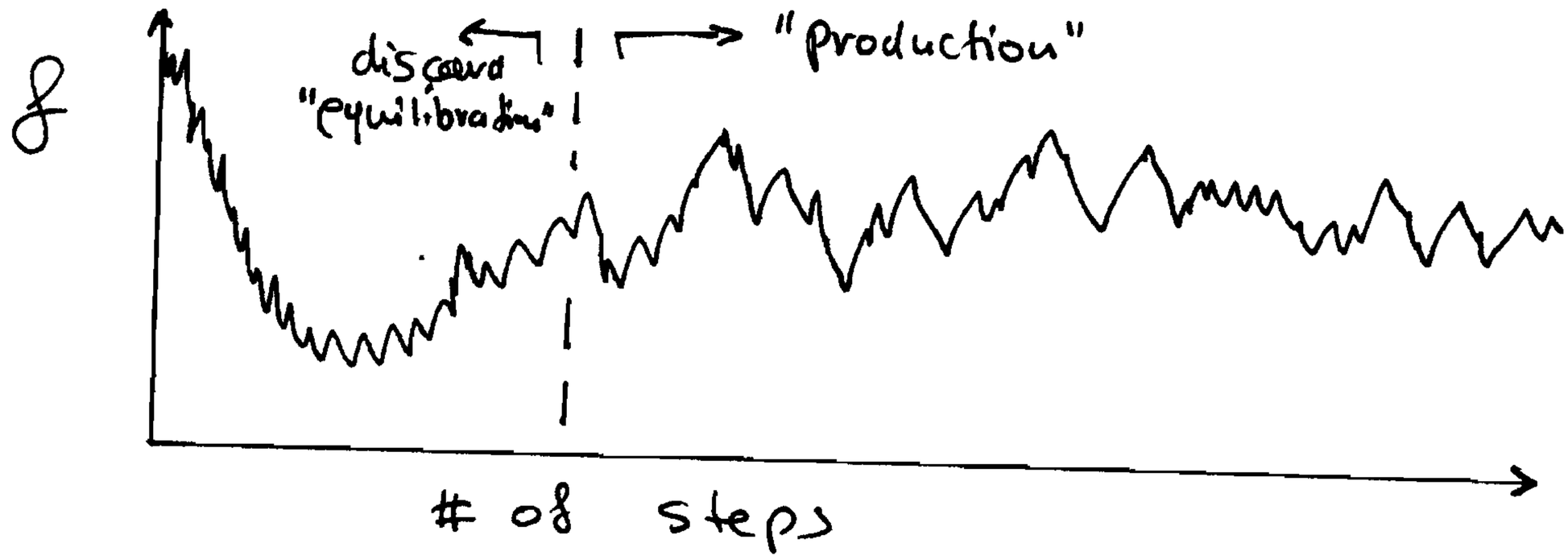
→ very helpful for detecting bugs

Also: VMD (Visual Molecular Dynamics)

Energy Conservation: At end of simulation, always recalculate energy "from scratch", compare to energy the program (which was obtained by adding du's every successful step). They should be equal!

Analysis of Simulation Results

At the end of a simulation, one has a list of observed values of f as a function of MC (or MD) step - e.g.



How do we obtain estimates for $\langle f \rangle$ and (equally important) an uncertainty estimate? For $\langle f \rangle$ itself, one needs to obtain an average of values obtained after an initial "equilibration" period is discarded, to eliminate artifacts of initial configuration,

$$\langle f \rangle = \frac{\sum f_\nu}{\#\text{ of "production" steps}}$$

The "simulation uncertainty" is not the standard deviation of the $\{f_\nu\}$ values - Recall fluctuation expressions,

$$\langle (\delta \underline{u})^2 \rangle = k_B T^2 \underline{\subseteq}_\nu$$

} so the standard deviation (fluctuations) are thermodynamic (extensive) properties

To obtain an estimate for the statistical uncertainties, we need to take into account the correlations among succesive samples

Elementary Statistics

Consider a normally distributed random variable with mean value $\bar{x}_\infty$ and Std. deviation σ_∞

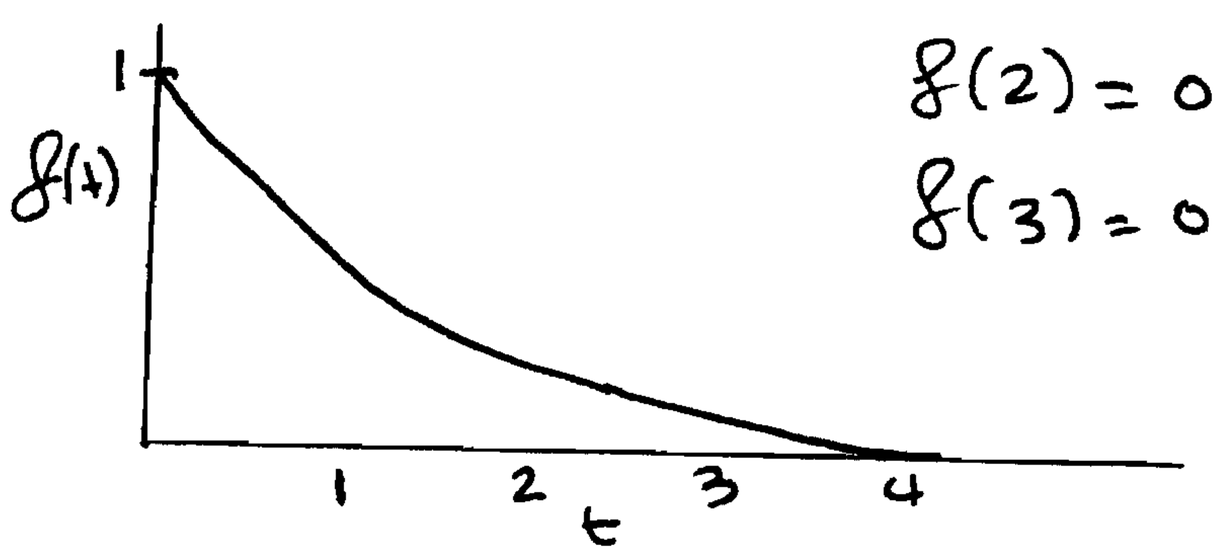
Draw N independent samples of x $\Big\}$: $\begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{Bmatrix}$ as $N \to \infty$

$\bar{x}_N \to \bar{x}_\infty$

$\sigma_N \to \sigma_\infty$

The normalized difference between sample mean $\bar{x}_N$ and true mean $\bar{x}_\infty$, $t = \dfrac{\bar{x}_N - \bar{x}_\infty}{s}$

where $s = \sigma_N/\sqrt{N}$ follows Student's t distribution w/ ∞ degrees of freedom:



$f(2) = 0.048$

$f(3) = 0.003$

$\Big\} \to$ multiply sample variance by

2 for 95% confidence

3 for 99.7%

→ In practice: Split production period of run into N "independent" blocks (typically 10-100).

Statistical uncertainties $\propto \dfrac{1}{\sqrt{\#\text{ of steps}}}$ $\Big\}$ ×10 in CPU time → ×3 in accuracy