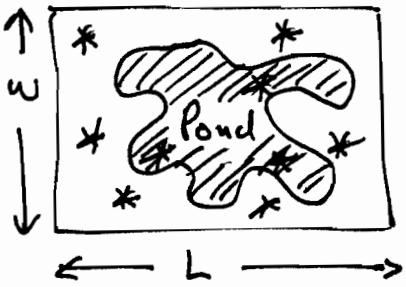
History

- Von Neumann, Ulam + Metropolis (1947) → "Studies of diffusion of neutrons in fissionable matter" Coined the term "Monte Carlo" (MC)
- Metropolis, Rosenbluth[2], Teller[2] (1953) → First MC simulation in Stat. Mech. (hard disks, P.S. #4)

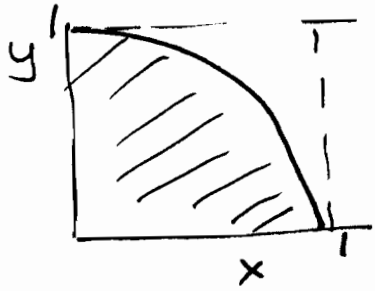
Monte Carlo Integration



random hits *

Area of pond ≈ W × L × $\times \frac{\text{hits in pond}}{\text{total hits}}$

Calculating areas is an integration problem – Monte Carlo sampling is a way to compute integrals.

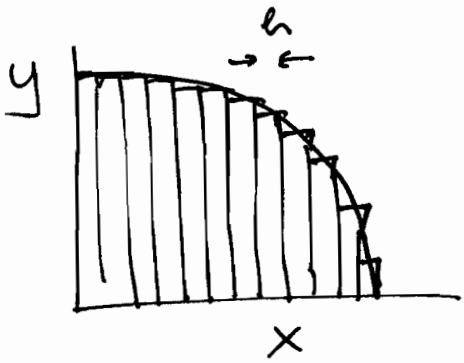


Find area of quarter-circle (or value of π) by random sampling:

After 10^7 trials $\pi \simeq 3.1417$ (4 sig. figs)

gain of 1 sig fig for 10× more trials

For low-dimensional functions, better use regular grid:



- Trapezoidal rule – error $O(h^3)$
- Simpson's rule – error $O(h^5)$ (approximates function by parabolas in intervals)

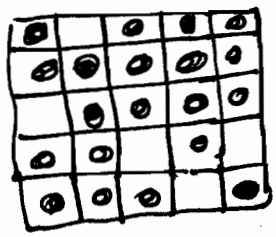
for 10^4 intervals gives $\pi = 3.141593$ (6 sig. figs.)

Multidimensional Integrals

Canonical partition function for $N \simeq 20$ distinguishable particles in 5×5 grid

$$Q = \sum_{\text{pos 1}} \sum_{\text{pos 2}} \sum_{\text{pos 3}} \cdots \sum_{\text{pos N}} \exp(-\beta U)$$

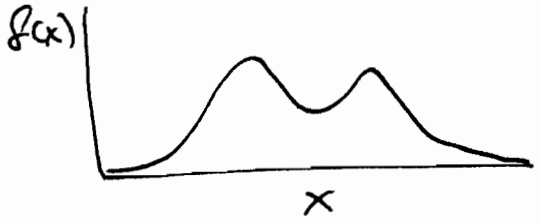
$25^{20} \simeq 10^{28}$ terms to be evaluated



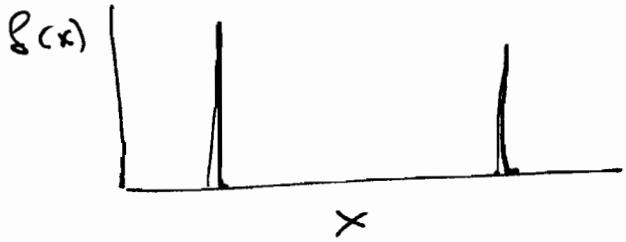
If there is volume exclusion, most of these terms contribute ϕ to Q, because U is very large.

[To be precise: $\frac{25!}{5!} = 1.3 \cdot 10^{23}$ allowable, 0.001% of total]

"normal" function



Stat. Mech functions



Importance Sampling (Metropolis et al.)

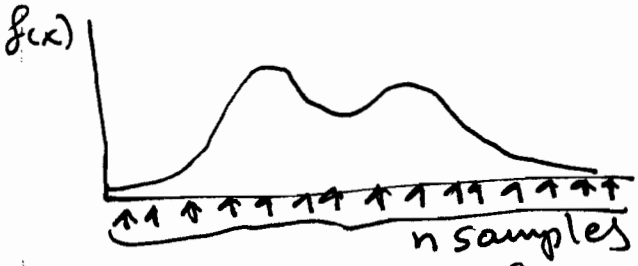
Instead of uniform sampling, which gives us

$\langle f \rangle = \Sigma f_i / n$

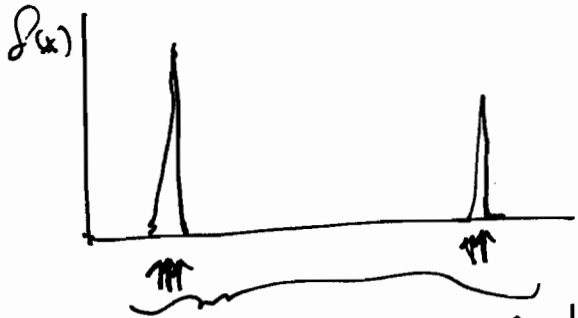
$\langle P \rangle = \dfrac{\Sigma f_i P_i}{\Sigma f_i}$

$\langle f \rangle = ?$ cannot compute

$\langle P \rangle = \left\langle \dfrac{\Sigma P_i}{n} \right\rangle$



randomly (uniformly) distributed



n sample, probability proportional to $f(x)$

How do we sample states with the correct probability?

Consider the system of 20 particles in a 5x5 grid.

Pick any two out of the 10^{28} states: old (o)
New (n)

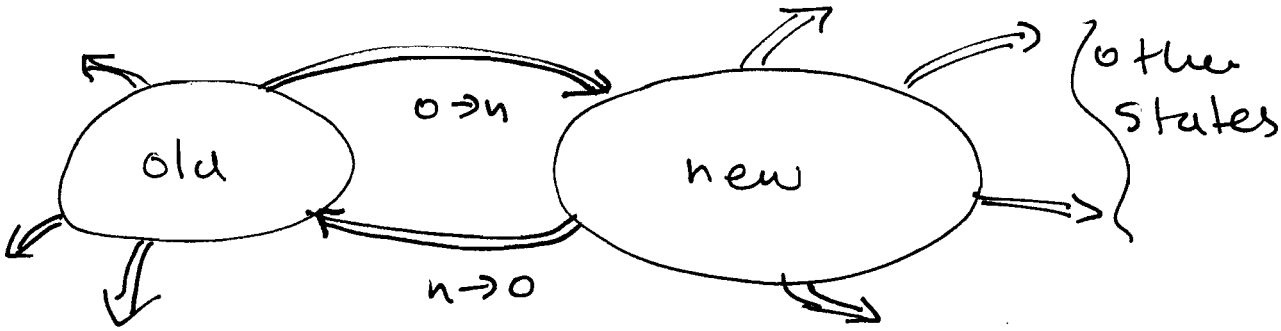
We would like to develop rules to allow a <u>sequence</u> of states to be generated so that in a very large sample ($\gg 10^{28}$ steps) each state appears with the correct probability, eg

$$\left.\begin{aligned} P_{old} &\propto \exp(-\beta U(o)) \\ P_{new} &\propto \exp(-\beta U(n)) \end{aligned}\right\}$$

In a large sample, $N(o)$, $N(n)$ instances of each

$$\frac{N(o)}{N(n)} = \frac{\exp(-\beta U(o))}{\exp(-\beta U(n))} = \exp(-\beta \Delta U)$$ @ NVT conditions

Think of water reservoirs with volumes prop. to $N(n)$, $N(o)$

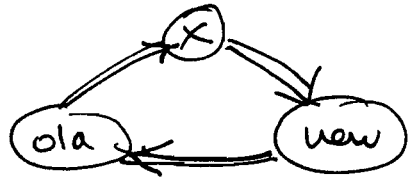


Arrows represent "flows" between states

<u>Detailed Balance Condition</u> $N(o) \cdot \Pi(o \to n) = N(n) \cdot \Pi(n \to o)$

Sufficient but not necessary for keeping the volumes of all reservoirs unchanged at equilibrium (steady state).

Why not necessary? Think of this →

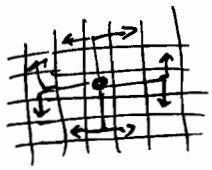


In practice, it is very hard to ensure that flows among all possible states are balanced, unless we enforce the (unnecessarily strong) detailed balance condition.

What are the probabilities of transition $\Pi(o \to n)$, $\Pi(n \to o)$?

$$\Pi(o \to n) = \underbrace{\alpha(o \to n)}_{\text{a priori probability of selecting state } n \text{ from state } o \text{ contains rules on moves}} \times \underbrace{\text{Accept}(o \to n)}_{\text{acceptance rule, to be determined}}$$

e.g.  attempt to move particle to a new position to left, righ, top, bottom of old position

 attempt to move particle using Chess rules for knights

α needs to satisfy <u>microscopic reversibility</u>

$$\alpha(o \to n) = \alpha(n \to o) \qquad \text{(symmetric)}$$

The acceptance rule for <u>any</u> symmetric α is now:

$$N(o) \cdot \Pi(o \to n) = N(n) \cdot \Pi(n \to o) \Rightarrow$$

$$N(o)\, \cancel{\alpha(o \to n)} \cdot \text{Accept}(o \to n) = N(n) \cdot \cancel{\alpha(n \to o)} \cdot \text{Accept}(n \to o)$$

$$\Rightarrow \frac{\text{Accept}(o \to n)}{\text{Accept}(n \to o)} = \frac{N(n)}{N(o)} = \exp\left(-B\left[U(n) - U(o)\right]\right)$$

There are several options for $\text{Accept}(o \to n)$;

Metropolis Rule:

$$acc(o \to n) = 1 \quad \text{if } U(n) < U(o)$$

$$acc(o \to n) = \exp(-\beta[U(n) - U(o)]) \quad \text{if } U(n) \geq U(o)$$

Symmetrical (Barker) Rule:

$$acc(o \to n) = \frac{\exp(-\beta U(n))}{\exp(-\beta U(o)) + \exp(-\beta U(n))}$$

Example of appliation of these rules:

Consider a system with exactly 10 states, numbered j:

j	1	2	3	4	5	6	7	8	9	10
$\exp(-\beta U(j))$	1	2	3	4	5	6	7	8	9	10

} probability is equal to index

What is the partition function? (Class Problem)

What is the equilibrium probability of state 1?

Now, let's make an importance sampling simulation.

Transition rule? $\alpha(o \to n) = 1/10$ for any pair of states o and n (including $o = n$)

$\Pi(o \to n)$ Metropolis

$o \downarrow$ / $\to n$	1	2	3	...	10
1	1/10	1/10	1/10	...	1/10
2	1/20	3/20	1/10	...	1/10
3	1/30	2/30	...		1/10
⋮					
10	1/100	2/100	...		11/20

$\Pi(o \to n)$ Barker

$o \downarrow$ / $\to n$	1	2	3	...	10
1	0.252	$\frac{2}{30}$	$\frac{3}{40}$	...	$\frac{10}{110}$
2	1/30	0.371	$\frac{3}{50}$	...	$\frac{10}{120}$
3	1/40				
⋮					
10	1/110		...		0.719

Barker's probability of remaining in the same state is higher than for Metropolis acceptance. In general, convergence from an arbitrary starting state to equilibrium is faster with Metropolis acceptance.

Example of flow of probabilities with the system with 10 states – The starting state is state 3, which we represent as [0 0 1 0 0 0 0 0 0 0]

Metropolis

Trial										
0	0	0	1	0	0	0	0	0	0	0
1	0.033	0.067	0.200	0.100	0.100	0.100	0.100	0.100	0.100	0.100
2	0.024	0.049	0.083	0.089	0.102	0.114	0.124	0.132	0.139	0.145
3	0.02	0.04	0.062	0.079	0.097	0.113	0.128	0.142	0.154	0.165
4	0.019	0.038	0.057	0.075	0.093	0.111	0.128	0.145	0.16	0.174
5	0.018	0.037	0.055	0.074	0.092	0.11	0.128	0.145	0.162	0.178
6	0.018	0.037	0.055	0.073	0.091	0.109	0.128	0.145	0.163	0.18
7	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.146	0.163	0.181
8	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.164	0.182
OO	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.164	0.182

Barker

0	0	0	1	0	0	0	0	0	0	0
1	0.025	0.040	0.454	0.057	0.063	0.067	0.070	0.073	0.075	0.077
2	0.025	0.046	0.225	0.075	0.086	0.095	0.103	0.109	0.115	0.12
3	0.023	0.044	0.128	0.079	0.093	0.106	0.118	0.128	0.137	0.145
4	0.021	0.041	0.087	0.078	0.095	0.11	0.124	0.137	0.149	0.16
5	0.019	0.039	0.069	0.077	0.094	0.111	0.127	0.141	0.155	0.168
6	0.019	0.038	0.061	0.075	0.093	0.111	0.128	0.144	0.159	0.173
7	0.019	0.037	0.057	0.074	0.093	0.11	0.128	0.145	0.161	0.176
8	0.018	0.037	0.056	0.074	0.092	0.11	0.128	0.145	0.162	0.178
9	0.018	0.037	0.055	0.073	0.092	0.11	0.128	0.145	0.163	0.18
10	0.018	0.036	0.055	0.073	0.091	0.11	0.128	0.145	0.163	0.18
11	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.163	0.181
12	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.146	0.163	0.181
13	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.164	0.181
14	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.164	0.182
OO	0.018	0.036	0.055	0.073	0.091	0.109	0.127	0.145	0.164	0.182

The Metropolis acceptance rule is asymmetric and works very well in most cases:

$$P[\text{Accept } (o \to n)] = \begin{cases} 1 & \text{if } u(n) < u(o) \\ \exp(-\beta[u(n) - u(o)]) & u_n \geq u(o) \end{cases}$$

In practice this can be implemented simply as:

```
If ( exp(-β Δu) > rand() )
    accept move ...
else
    reject move ...
end
```

If β is large (T low) acceptance of only lower energy states, Metropolis reduces to energy minimization.

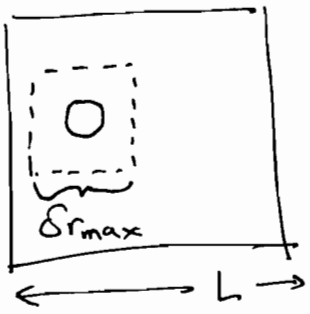
Trial Moves

Should be (a) microscopically reversible (symmetric)
(b) ergodic: able to cover <u>all</u> of phase space
(c) efficient: to explore phase space <u>quickly</u>

e.g. Should we move one or more particles at a time?

If p_1 = single particle acceptance probability, then $p_N \simeq p_1^N$. Cost of move is prop. to N, so clearly better move them one at a time

Displacements (continuum systems)



$$x_{new} = x_{old} + \delta r_{max} \cdot (rand() - 0.5)$$
$$y_{new} = y_{old} + \delta r_{max} (rand() - 0.5)$$
$$z_{new} = z_{old} + \delta r_{max} (rand() - 0.5)$$

How do we pick δr_{max}?

If δr_{max} is too low → high acceptance, but slow motion in space

If δr_{max} is too high → low acceptance, also slow motion

"Optimal" δr_{max} → to give acceptance of moves of ~ 30%

Why not 50%? (mid-point between 100% and 0%)?

Reasons have to do with the fact that failed moves are cheaper than successful ones, e.g. by finding out that two particles have overlapped.

Initial Configuration

In principle, any initial configuration can be used, "memory" of initial state disappears after a while if sampling is efficient

Common choices
1. random, non-overlapping (may be very hard at high ρ)
2. on a lattice - see Problem Set #4