ANNOUNCE

→ MIDTERMS

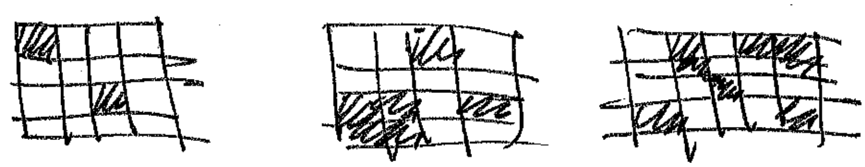
today {
1. Julián : Binary Search
2. Gus : Stacks
3. Nick : ÷ & conquer
4. BAE : QUICKSORT
5. Steven : HASH TABLE

Thu : next 4

INDEX CARD
- QUESTIONS/ Feedback re: class
- SUGGESTION for future midterm talks

Review

- Stat Mech is hard : $N \sim 10^{23}$ — eg 2 state system is 2^N dim. space!!
  - many $\boxed{\mu\text{states}}$ that we have to average over
    "microstate"
    this is a WEIGHTED average
  - care about macro properties
    eg Magnetization is an average over spins



if we just averaged over all config of 2^N μstates ↳ $M = 0$
no magnetism.

but: what we're missing:

DIFF. ENERGY CONFIGURATIONS ... weighting:

$$P_i = e^{-\beta E_i}/Z$$

indep of i, but impossible to calculate.

So: let's throw computers @ this problem.

[alternative: fancy math... leave that to a proper stat mech course]

- Monte Carlo / Canto Bight
  - use random # generator to sample $N \ll 10^{23}$ μ states.
  - since the macro observables are averages, we just need enough samples to get a good average.

eg: 

$S_i = 1$ in circ.

$S_j = 0$ outside circ.

small circle marble game

$A_\circ / A_\square = \frac{\pi}{4}$ analytic.

as a statmech problem:

assuming uniform probability across square

can ask: $\langle s \rangle = \sum_i S_i P_i$

0 or 1

$= \frac{1}{\#\text{ grid points}}$ if 

or: $1/\tilde{N}$ some unknown BUT CAN DIVIDE OUT

eg: $r_i = 1$ in square

$\langle r \rangle = \sum_i r_i P_i$

$= \sum \frac{1}{\tilde{N}}$

$\sim e^{-\beta E_i}/Z$

then calc: $\langle s \rangle' = \frac{\langle s \rangle}{\langle r \rangle} = \sum_i S_i \frac{P_i}{\sum_j P_j} = \sum_i S_i$

so that's easy – direct sampling

↳ we know P_i & it's trivial.

<u>ANALOGY</u>: each marble position ~ μ state.

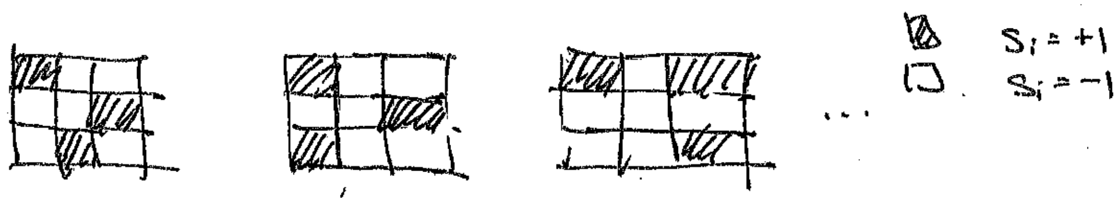
S_i ~ observable to avg over.

BUT

<u>REAL STAT MECH</u> : $P_i \sim e^{-\beta E_i}/Z$ is <u>hard</u> to calc.

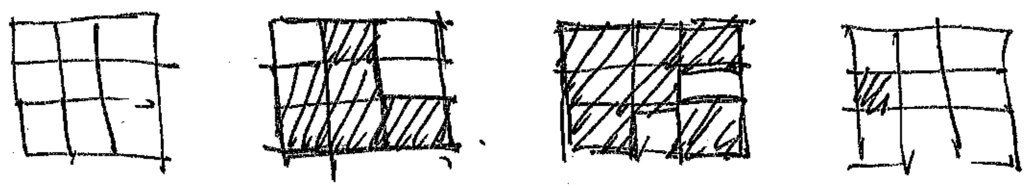
WE NEED: A SAMPLE OF μ STATES THAT ARE REPRESENTATIVE OF THE BOLTZMANN DISTRIBUTION

so:

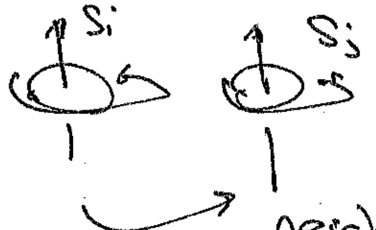


not really a great sample of spins

expect (@ low temp)



<u>clumpy</u>!



neighboring spins affect each others' energy configs!

BTW: BOTH SAMPLES HAVE $M = \langle s \rangle = 0$

~~but second has~~

~~M = ...~~ but imagine a <u>small</u> BG $\vec{B}$ field...

SO

<u>NEED WAY TO SAMPLE faithfully</u>

<u>MARKOV CHAIN</u>:

generate sample sequentially

given $\mu state_i$, $\boxed{\text{RULES}}$ $\rightarrow$ $\mu state_{i+1}$

we have to choose these (<u>MCMC</u> algorithm choice)

Markov chain Monte Carlo

. . . (RUN for A WHILE)

then take this as our ENSEMBLE.

<u>Markov</u>: probability to go to a given state dep. <u>only</u> on <u>current state</u>

(locality in <u>time</u>, analog of why our ~~PDE~~ EOM are always PDEs)

REMEMBER: $P(\text{state})$: <u>unknown</u> (Boltzmann) PROB to ~~...~~ sample a state

<u>physics</u>

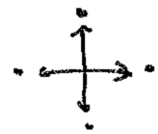
$p(\text{state}_i \rightarrow \text{state}_j)$: <u>transfer</u> probability of our MCMC algo.

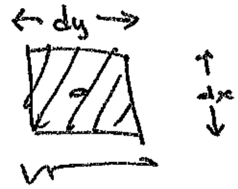
<u>ALGORITHM</u>.

BIG MARBLE GAME : MCMC, Drunkards walk (RANDOM WALK)

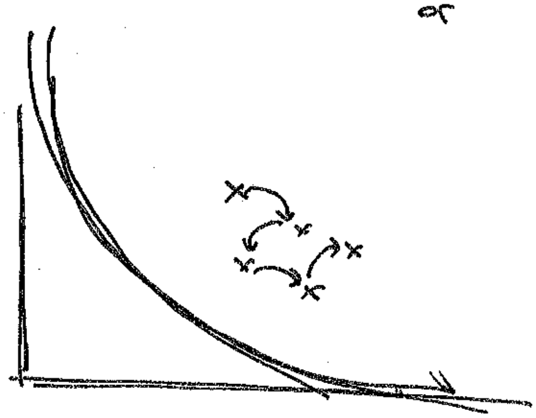
Random walk is an eg of a MCMC algo.

↳ each step: uniform TRANSITION probability to a neighboring site

eg  $p(i \to nbr) = 1/4$

or  or 

what we did in sample code



last time: how to pick a MCMC algo?

↳ DETAILED BALANCE

we will return to this: Metropolis algo for ISING MODEL