PY410 / 505
Computational Physics 1

Salvatore Rappoccio

Probabilistic methods

Casino Monte Carlo

- A large number of processes in nature are random
- To some extent, we're all basically familiar with it
  - Radioactive decay
  - Flipping a coin
- In fact, quantum mechanics inherently is random

Probabilistic methods

- What computational issues can come up with random numbers?
  - Computers are, by very nature, NOT random
  - So we need to make them LOOK random
  - The question is, how random is "random enough"?

- If you want something truly random, you'll need to hook your computer up to a Geiger counter or something, and count decays (say, from atmospheric muons)

- Sounds silly, but it isn't

Probabilistic methods

- Chapter 7 in Numerical Recipes deals with generating random numbers
  - "Deviates"
- How to formalize random number generation (RNG)?
  - Given a set S of N uniformly distributed random numbers x1...xN, then they must satisfy :
    - Given n generated numbers, the next number x_n+1 must be independent and uncorrelated
    - x_n+1 should be equally likely to be a member of S

  - For the example of an unbiased coin toss, S has 1 or 0 (heads or tails)
  - Each toss is independent of the previous and so a priori equally likely to be 1 or 0

Probabilistic methods

- Why do computers have trouble here?

- They are inherently deterministic at the present time (that’s why they’re so great to use!)

- We generate “pseudo” random numbers
  - The statistical properties (i.e. equally likely in some region of interest) is “good enough”, but “good enough” depends on the situation

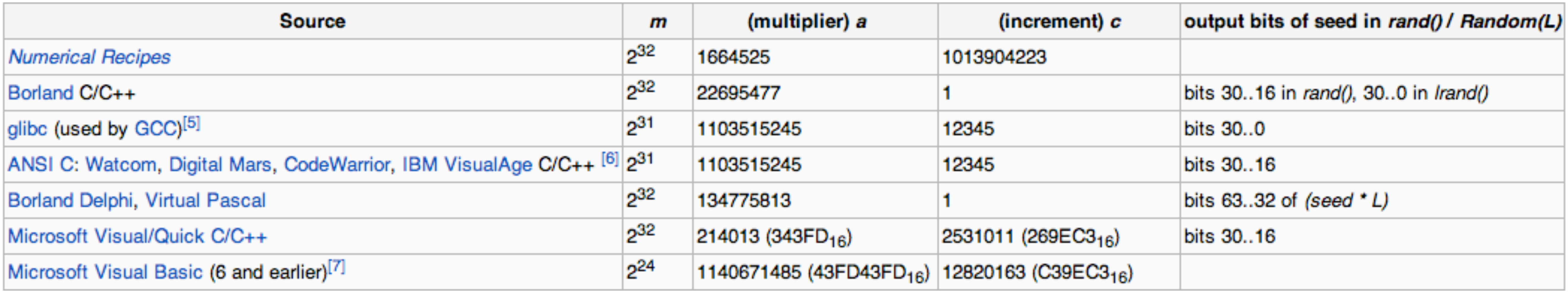
- For instance, if we use <u>linear congruential algorithms</u> of the typ

$$x_{n+1} = (\underset{\nwarrow\text{“multiplier”}}{a} x_n + \underset{\nwarrow\text{“increment”}}{c}) \mod \underset{\nwarrow\text{“modulus”}}{m} ,$$

  - This generates a sequence of random integers in the set {0,1,...,m-1}

Probabilistic methods

- Why does this help?
  - We typically choose a, c, and m to be large(ish) relatively prime numbers
  - If we inappropriately choose a,c, or m, we can have repetition
  - If any x_n = x_0, then the process will repeat
  - We initialize this with a “seed” x0, and away goes the sequence



Source	***m***	**(multiplier) *a***	**(increment) *c***	**output bits of seed in *rand()* / *Random(L)***
Numerical Recipes	2^{32}	1664525	1013904223	
Borland C/C++	2^{32}	22695477	1	bits 30..16 in *rand()*, 30..0 in *lrand()*
glibc (used by GCC)[5]	2^{31}	1103515245	12345	bits 30..0
ANSI C: Watcom, Digital Mars, CodeWarrior, IBM VisualAge C/C++ [6]	2^{31}	1103515245	12345	bits 30..16
Borland Delphi, Virtual Pascal	2^{32}	134775813	1	bits 63..32 of *(seed * L)*
Microsoft Visual/Quick C/C++	2^{32}	214013 ($343FD_{16}$)	2531011 ($269EC3_{16}$)	bits 30..16
Microsoft Visual Basic (6 and earlier)[7]	2^{24}	1140671485 ($43FD43FD_{16}$)	12820163 ($C39EC3_{16}$)	

Probabilistic methods

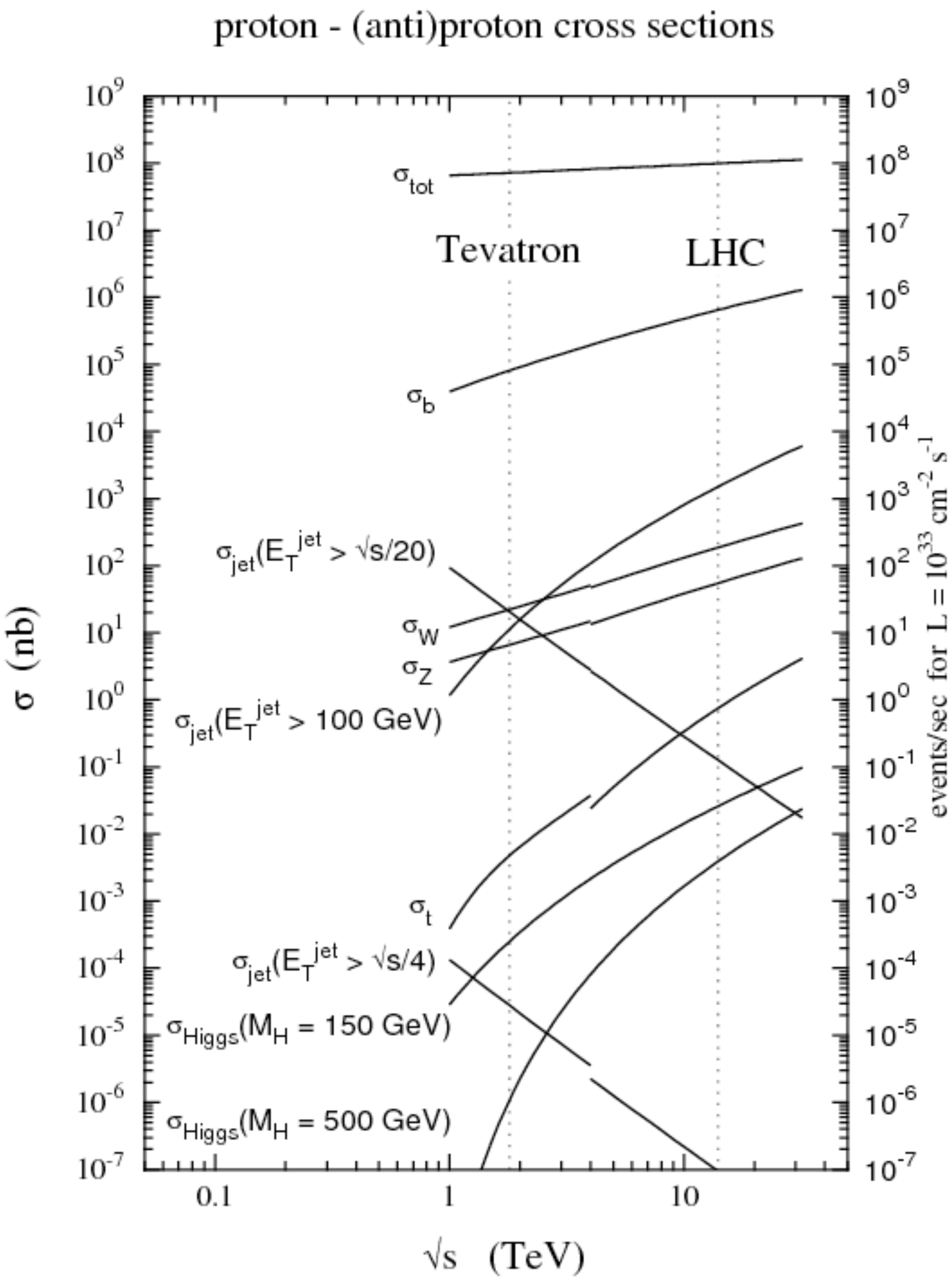
- There are several industrial-strength generators on the market
- But, if you need "really really" random numbers, use with extreme care
  - C++11, R, Python, Ruby, IDL, Maple, Matlab, GNU MPAL, BOOST, Glib, and NAG :
    - http://en.wikipedia.org/wiki/Mersenne_twister
    - Long period of 2^19937 - 1
    - Passes lots of randomness tests
    - NOT suitable for cryptography : observing a certain number of iterations will allow you to predict the rest of the sequence

  - Numerical recipes recommends
    - http://en.wikipedia.org/wiki/Xorshift
    - Period of 2^128 - 1

Probabilistic methods

- Example: the “sparky” trigger
- In particle physics, we throw away almost all of our data
- Only 1 in 10^5 is even “remotely interesting
- The really interesting stuff is only 1 in ~10^{16}!

- Given this, we have to often have “REALLY” random numbers to test
- So, we set up a spark chambe to actually generate random numbers so our triggers could perform adequate tests



Probabilistic methods

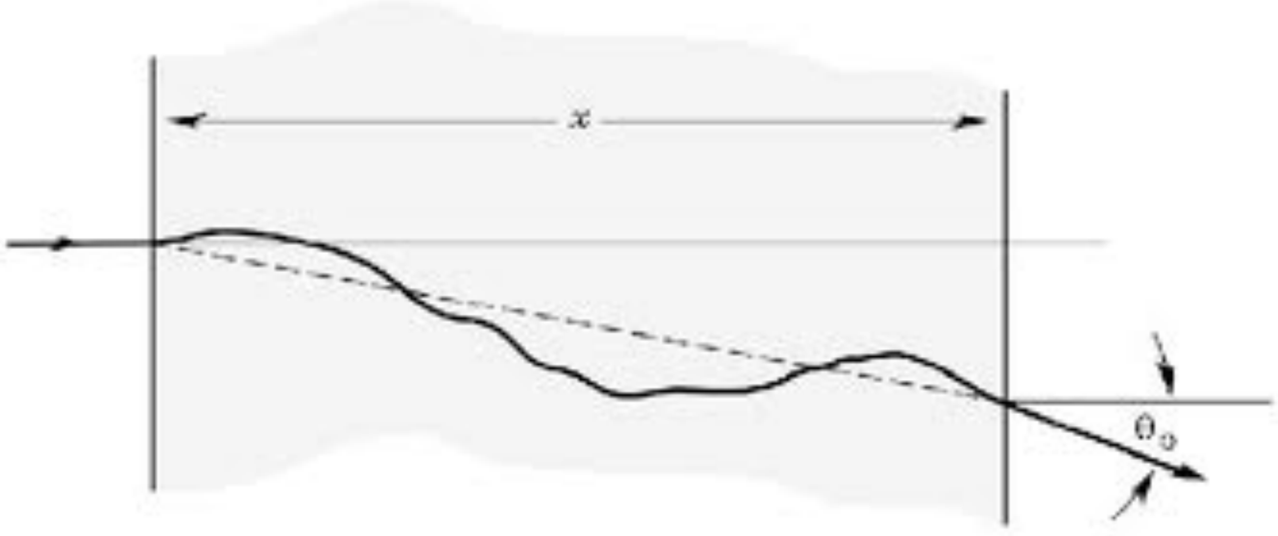
- A few tests will be performed here :
  - The period of the generator should be much larger than the length of the generated sequence
  - A simple “eyeball test” (plotting (x_n,x_{n+1}) as (x,y) pairs) should reveal no structure
  - The chi2 statistic should satisfy $\frac{\chi^2}{\text{d.o.f.}} \simeq 1$.

Probabilistic methods

- Randomness is often used in simulations

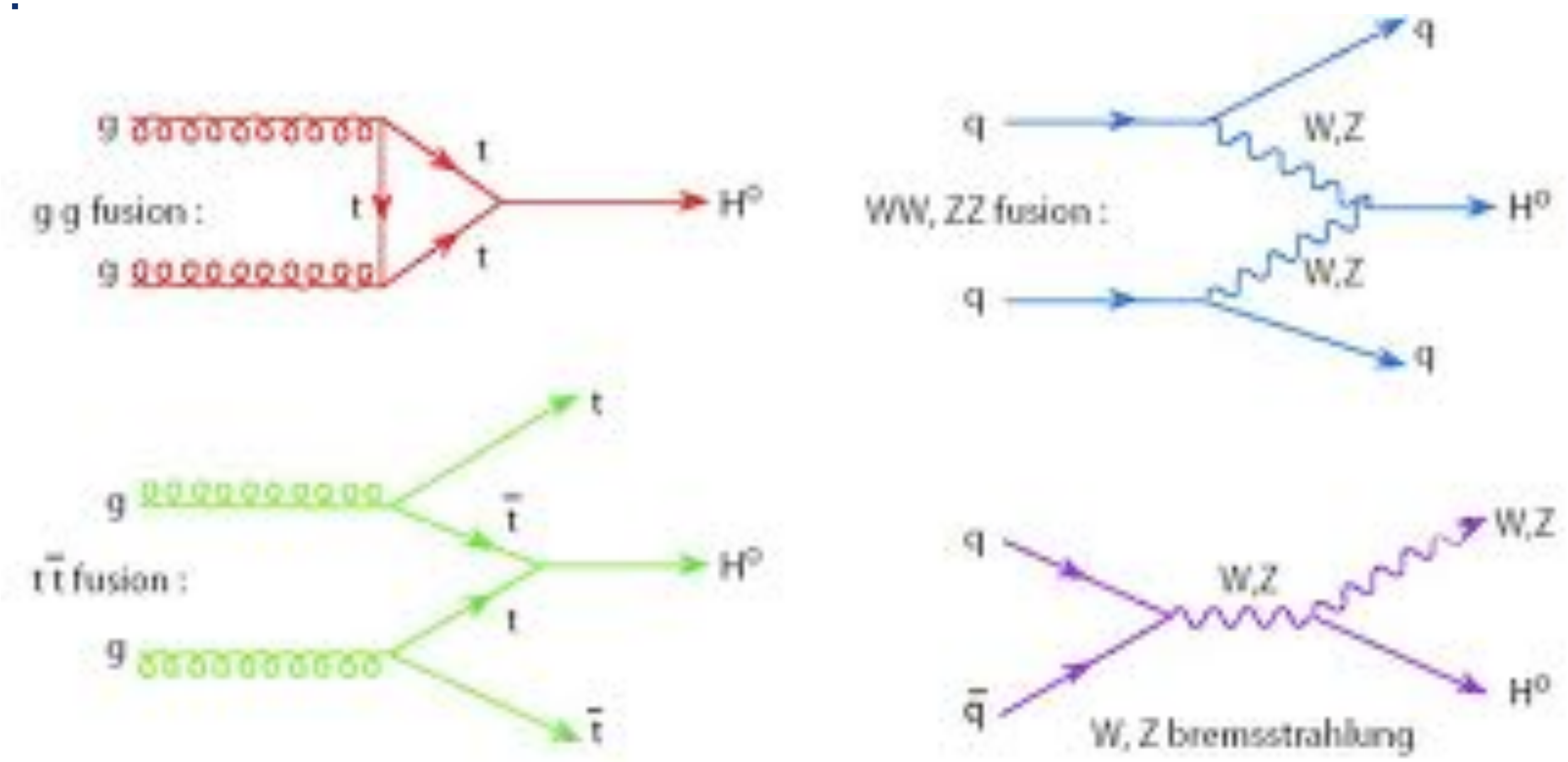
- Lots of things are random in nature
- We often know their distribution, but cannot predict individual events :
  - Standing a pencil on its side, what is the angle (wrt the desk) when it falls?

  - What is the final angle between multiple scatterings of particles off of a lattice?



Probabilistic methods

- So, to simulate these events, we need random (or pseudorandom) generators
- Example : Quantum mechanically, can produce Higgs bosons in several ways :



- Can’t even ask (quantum mechanically) which will occur before it happens
- They all have a likelihood!
- So, to generate them, you have to randomly sample between these four (with appropriate weights)

Probabilistic methods

- These methods are often called “Monte Carlo” after the Casino Monte Carlo in Monaco
- Much better name than “pseudo-random-number probabilistic event simulator”

Probabilistic methods

- Why does this help?
- Suppose you take a system of 100 electrons in either “spin up” or “spin down” configurations
- Total number of states is $= 2^{100} = 1.27 \times 10^{30}$
- Already intractable to list them (at one billion / second, would take 4.02×10^{13} years!)

- If you select one at random for any purpose, need to simply find a sufficiently good randomization algorithm!

Probabilistic methods

- Markov-Chain MC (MCMC) :
  - Sequence of elements chosen from a fixed set using a probabilistic rule
  - Chain is constructed by adding the elements sequentially
  - Given the most recently added element, next element only depends on most recent addition

  - Formally, suppose x and y are members of a set S
  - The transition probability function is

$$T(x \to y)\ , \qquad \sum_{y} T(x \to y) = 1\ .$$

  - Example : Random walks

Probabilistic methods

- Random walks are simple examples of MCMC's
- Suppose a walker can occupy any site on an infinitely-long 1-d lattice
- The walker tosses a coin and decides to go left or right (1 or 0)
- The transition probability is therefore

$$T(x \to y) = \begin{cases} \frac{1}{2} & \text{if } y = x - 1, \\ \frac{1}{2} & \text{if } y = x + 1, \\ 0 & \text{otherwise.} \end{cases}$$

- In equilibrium, suppose the lattice is periodic now, with L lattice points on a circle
  - Then we can take L->infinity to examine the properties
- Probability is $P(x) = \frac{1}{L}$:
  (walker visits each site the same number of times)

Probabilistic methods

- The walker's position after n steps depends on the sequence of tosses in the past, and cannot be predicted

$$x_n = \sum_{i=1}^{n} s_i \ , \qquad\qquad s_i = \pm 1$$

- Over a large number of n-step walks the average is zero
  - (From symmetry : left and right are equally likely!)

- However, each xn is not zero, and DOES increase:

$$\begin{aligned}\langle x_n^2 \rangle &= \left\langle \sum_{i=1}^{n} \sum_{j=1}^{n} s_i s_j \right\rangle \\ &= \left\langle \sum_{i=1}^{n} s_i^2 \right\rangle + \left\langle \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} s_i s_j \right\rangle \\ &= n \ ,\end{aligned}$$

Diffusion equation!

Probabilistic methods

- RMS displacement is

$$\sqrt{\langle x_n^2 \rangle} = \sqrt{n} \; .$$

- Diffusion constant is

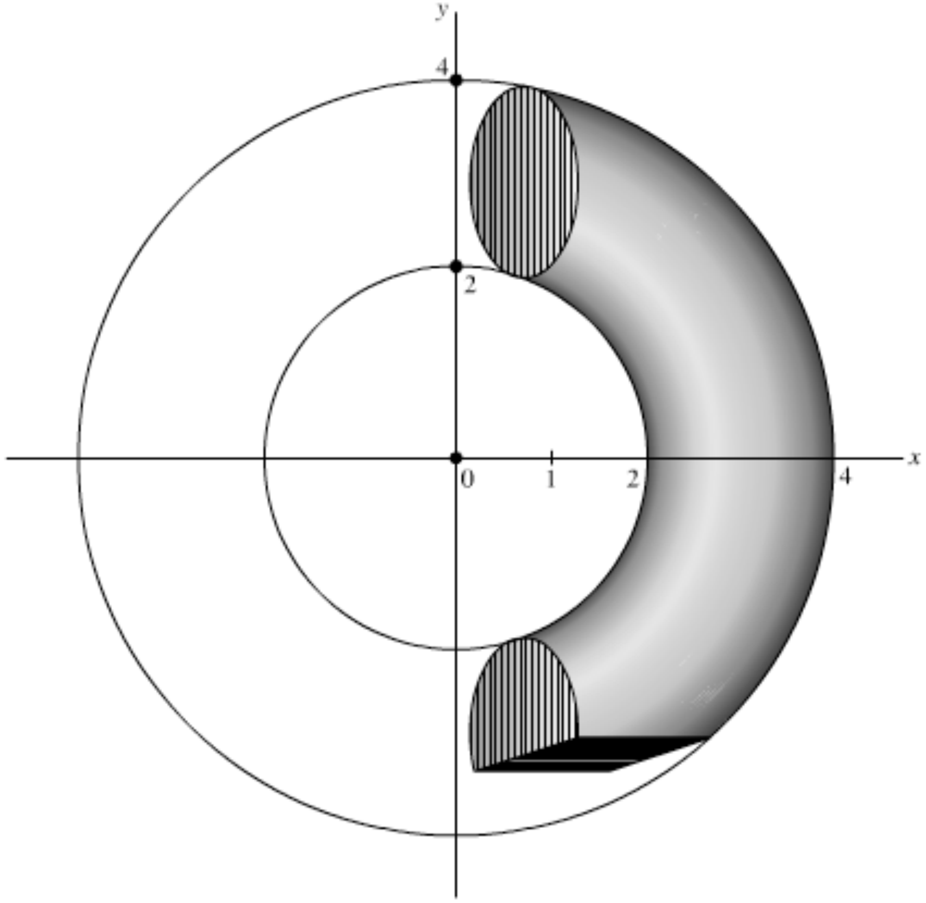
$$\langle x_n^2 \rangle = 2Dn \; .$$

- So, for a 1-d walk, D = 1/2

Probabilistic methods

- Another example of MC methods : integration of functions

- Sprinkle the area you’re interested with a “dust” of points uniformly distributed
- The fraction “below the curve” (or in N-dim, within the surface) is the integral!



Probabilistic methods

- Formally, the integral is :

$$I = \int_V d^d\mathbf{x}\, f(\mathbf{x})$$

- Now we choose N uniformly distributed points in V, and estimate : $I \simeq \frac{V}{N} \sum_{n=0}^{N-1} f(\mathbf{x}_i)$

- The error of repeating this M times is :

$$I_m = \frac{b-a}{N} \sum_{n=0}^{N-1} f(\mathbf{x}_{m,n})\,, \qquad m = 0 \ldots M-1$$

- Can just compute the mean and standard deviation:

$$\bar{f} = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=1}^{N} f(\mathbf{x}_{m,n})$$

$$\sigma_M = \sqrt{\frac{1}{M} \sum_{m=0}^{M-1} I_m^2 - \left(\frac{1}{M} \sum_{m=0}^{M-1} I_m \right)^2} = \sqrt{\frac{1}{M} \sum_{m=0}^{M-1} \left(I_m - \frac{1}{M} \sum_{m'=0}^{M-1} I_{m'} \right)^2}$$

Probabilistic methods

- If the measurements are independent and randomly distributed about the mean then:

$$\sigma_M^2 = \frac{(b-a)^2}{N}\sigma_f^2$$

- where

$$\sigma_f^2 = \overline{f^2} - (\bar{f})^2 = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=1}^{N} f(\mathbf{x}_{m,n})^2 - \left(\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=1}^{N} f(\mathbf{x}_{m,n})\right)^2$$

- Thus for MC integration:

$$I = \int_V d^d\mathbf{x}\, f(\mathbf{x}) \simeq V\left[\frac{1}{N}\sum_{n=1}^{N} f(\mathbf{x}_i) \pm \frac{\sigma_f}{\sqrt{N}}\right]$$

Probabilistic methods

- Compare this with our previous integration methods
- For instance, Midpoint method
- We have $$\frac{\sigma_f}{\sqrt{N}} < \frac{\overline{f''}}{N^{2/d}}$$
- MC integration “wins” when d > 4 (roughly)

Probabilistic methods

- What if you want nonuniform samples?
  - Example : Gaussian distribution, exponential distribution, line segment

  - Still use the uniformly-distributed “deviates”
  - Then change variables (x ->y(x)) to produce another distribution :

$$P(y) = P(x)\left|\frac{dx}{dy}\right| , \qquad \int dy\, P(y) = \int dx\, P(x) = 1 \,.$$

Probabilistic methods

- Uniform line segment :
  - Basically trivial, you just redefine (0,1) to (a,b) and you're done:

$$y(x) = a + (b-a)x \,, \qquad \left|\frac{dx}{dy}\right| = \frac{1}{|b-a|} \,.$$

$$P(x) = \begin{cases} 1 & \text{for } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases} \qquad \Rightarrow \qquad P(y) = \begin{cases} \frac{1}{|b-a|} & \text{for } a < y < b \\ 0 & \text{otherwise} \end{cases} .$$

Probabilistic methods

- Exponential distribution :

$$P(y) = \frac{1}{\lambda} e^{-y/\lambda} , \qquad 0 < y < \infty$$

- Then we have

$$y(x) = -\lambda \log(x) , \qquad \left| \frac{dx}{dy} \right| = \frac{x}{\lambda} = \frac{e^{-y/\lambda}}{\lambda} = P(y) .$$

$$P(x) = \begin{cases} 1 & \text{for } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

- so the distribution is

$$P(y) = \begin{cases} \frac{e^{-y/\lambda}}{\lambda} & \text{for } 0 < y < \infty \\ 0 & \text{otherwise} \end{cases} .$$

Probabilistic methods

- Gaussian distribution ("normal" distribution)

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/(2\sigma^2)} ,$$

- Cannot convert with elementary functions
- However, consider the product of TWO Gaussian distributions in x and y :

$$\int dx \int dy P(x)P(y) = \frac{1}{2\pi\sigma^2} \int dx \int dy e^{-[(x-\mu)^2+(y-\mu)^2]/(2\sigma^2)}$$

$$= \int_0^{2\pi} \frac{d\theta}{2\pi} \int_0^{\infty} \frac{dr^2}{2\sigma^2} e^{-r^2/(2\sigma^2)} ,$$

- We have switched to polar coordinates:

$$x - \mu = r\cos\theta , \qquad y - \mu = r\sin\theta .$$

Probabilistic methods

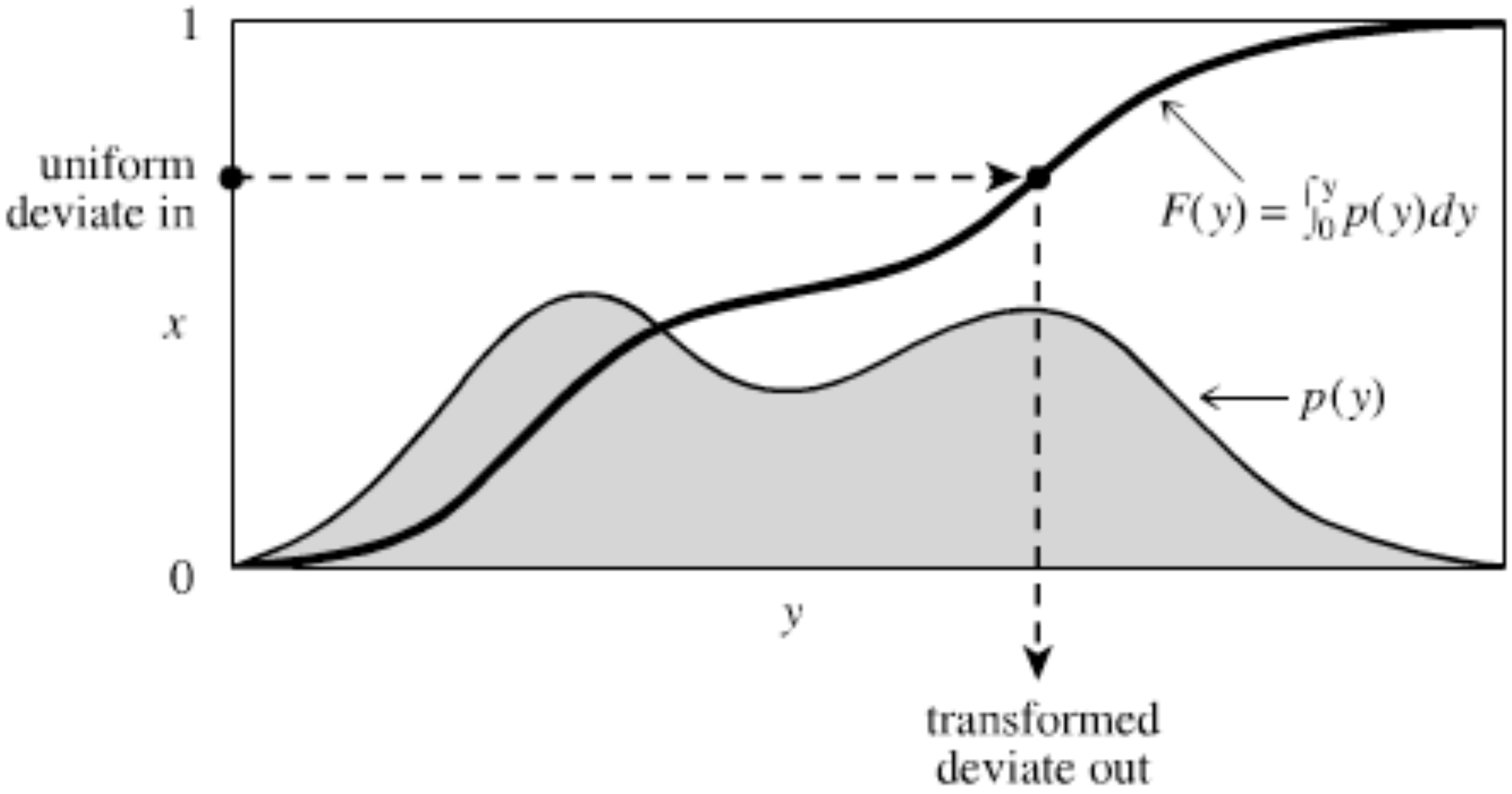
- This is the product of a uniform distribution on [0,2pi] with an exponential distribution in r^2!
- So we can generate theta and r, convert to x,y!

$$\theta = 2\pi r_1 \ , \qquad r = \sqrt{-2\sigma^2 \log r_2} \ ,$$

  - “Box-Muller” algorithm

Probabilistic methods

- General version of transformation method:



  - –Can be used even if you only know F(y) numerically

Probabilistic methods

- To generate this kind of sequence, though, the Metropolis-Hastings algorithm works well
    - http://en.wikipedia.org/wiki/Metropolis–Hastings_algorithm
    - Type of Markov Chain Monte Carlo (MCMC)

    - One major advantage : it does not require the overall normalization of a distribution to be known to draw a random sequence from it!
        - Advantageous when using Bayesian statistics

- Original papers are from Metropolis et al and Hastings :
    - J. Chem. Phys. 21, 1087 (1953)
    - Biometrika 57, 97 (1970)

Probabilistic methods

- This is a Markov Chain MC so the “next” step depends only on the “current” step
  - This is a type of random walk, similar to other MCMC methods
- Suppose we want to generate a sequence from a probability distributio$P(\vec{x})$
- If we visualize this in d-dimensions, then P(x) is a mountainous terrain and the sequence is the steps of a hiker

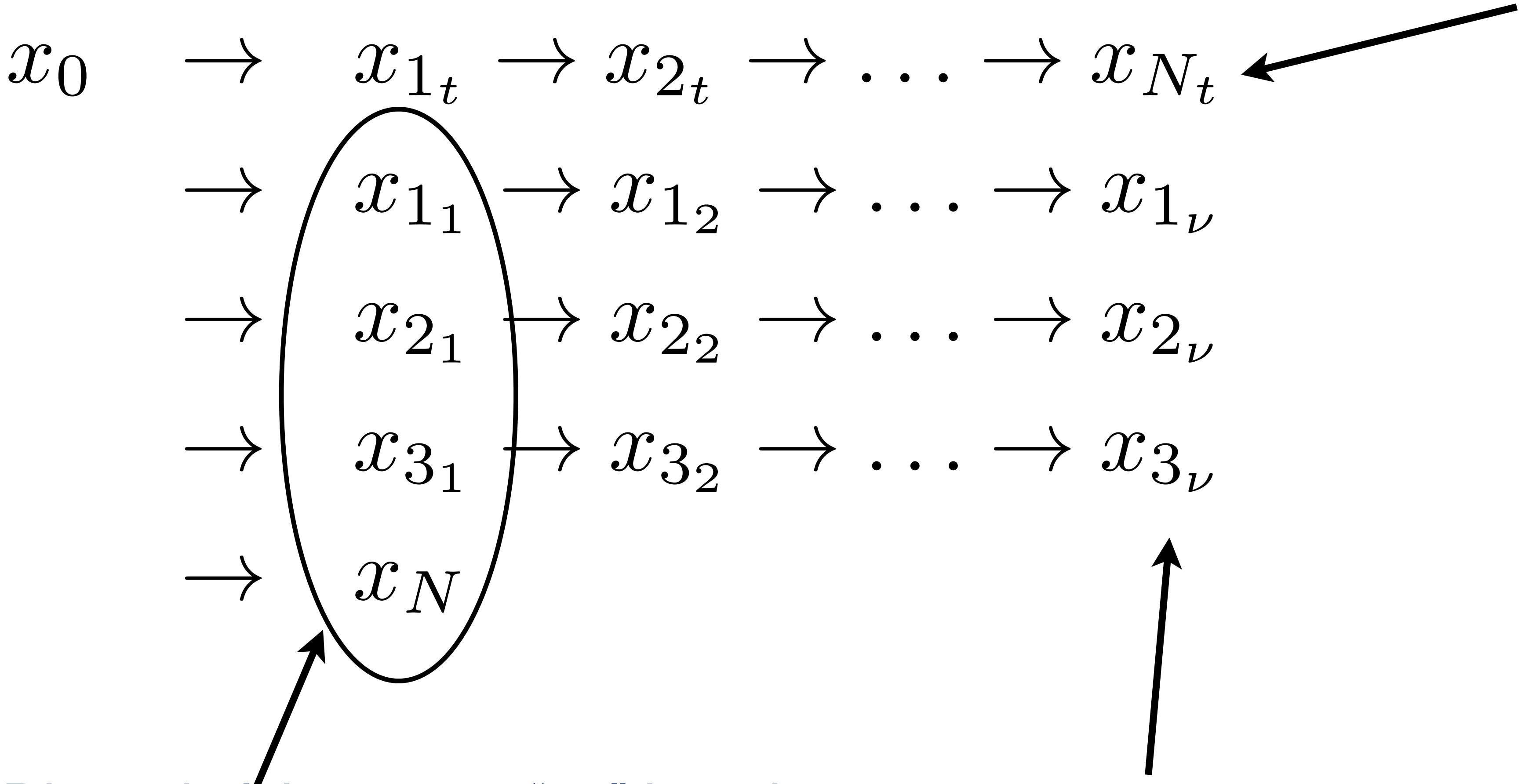
Probabilistic methods

- Heuristically :
  - Explore the “terrain” a bit
    - Find peaks, valleys
  - Then generate the steps for the MC application
  - Record steps every so often

- As more and more sample values are produced, the distribution more closely approximates the desired distribution P(x)

Probabilistic methods

- More specifically, we have steps :



"thermalization" steps to find the peaks

$$\begin{array}{lllllllll}
x_0 & \to & x_{1_t} & \to & x_{2_t} & \to & \dots & \to & x_{N_t} \\
 & \to & x_{1_1} & \to & x_{1_2} & \to & \dots & \to & x_{1_\nu} \\
 & \to & x_{2_1} & \to & x_{2_2} & \to & \dots & \to & x_{2_\nu} \\
 & \to & x_{3_1} & \to & x_{3_2} & \to & \dots & \to & x_{3_\nu} \\
 & \to & x_N & & & & & &
\end{array}$$

Discard all but every "nu" iterations

"Sampling frequency" nu

Probabilistic methods

- Why sample?
  - Individual steps may be correlated
    - If walker’s step size is much smaller than typical distances between “peaks” in the terrain
    - To emulate a “truly” random sequence, we don’t want to get “stuck” between peaks

Probabilistic methods

- Walking algorithm :
- Suppose we have the walker at a point $\vec{x}_{\text{current}}$
- Then the next step $\vec{x}_{\text{next}}$ is determined by
  - Choose a “trial” point randomly in the neighborhood of current x ($\vec{x}_{\text{trial}}$)
    - 1-d : maximum step size (delta) may be used, and choose trial x in a uniform interval $[x_{\text{current}} - \delta, x_{\text{current}} + \delta]$
    - n-d : could try a fixed sphere of radius delta
  - Requirement is that the probability $P_{\text{try}}(\vec{x}_{\text{current}} \to \vec{x}_{\text{trial}})$ should be SYMMETRIC: $P_{\text{try}}(\vec{x} \to \vec{y}) = P_{\text{try}}(\vec{y} \to \vec{x})$
    - In other language, the conditional probabilities must be equal :

$$P(x|y) = P(y|x)$$

  - Calculate the next step by the ratio of probabilities (or “likelihoods”) : $r = P(\vec{x}_{\text{trial}}) / P(\vec{x}_{\text{current}})$

Probabilistic methods

- Given $r = P(\vec{x}_{\text{trial}})/P(\vec{x}_{\text{current}})$
- If r >=1, walker is moving “uphill” (toward higher probability)
  - Trial step is accepted, iterate
  - The “acceptance probability” is 1 by definition (it’s accepted, $P_{\text{accept}}(\vec{x}_{\text{current}} \to \vec{x}_{\text{trial}}) = 1$)
- If r < 1, walker is moving “downhill” (toward lower probability)
  - Accept only if displacement is not too large :
    - Generate uniform deviate n_{random}
    - Accept if $r > n_{\text{random}}$, iterate
    - Reject otherwise, remain at current “step”
    - “Acceptance probability” in this case is r

Probabilistic methods

- What about those thermalization steps? Do they really thermalize?

- Imagine a large ensemble of walkers distributed with density $\rho(\vec{x})$

- Want to show that the ensemble density becomes proportional to P(x) and then remain invariant

- According to the ergodic hypothesis, the average properties of an ensemble of systems should be the same as the average over time in a single system in thermal equilibrium
  - http://en.wikipedia.org/wiki/Ergodic_hypothesis

- So, we'll use the former to demonstrate the latter

Probabilistic methods

- Recall the transition probability for a walker to move from x to y is determined by two consecutive and independent decisions:

$$T(\vec{x} \to \vec{y}) = P_{\rm try}(\vec{x} \to \vec{y}) \times P_{\rm accept}(\vec{x} \to \vec{y}) \ .$$

- If these T(x->y) are symmetric, then:

$$\frac{T(\vec{x} \to \vec{y})}{T(\vec{y} \to \vec{x})} = \frac{P_{\rm accept}(\vec{x} \to \vec{y})}{P_{\rm accept}(\vec{y} \to \vec{x})} = \frac{P(\vec{y})}{P(\vec{x})} \ .$$

- Second equality can just be verified directly:
  - –if P(y) > P(x), $P_{\rm accept}(\vec{x} \to \vec{y}) = 1$,

    $P_{\rm accept}(\vec{y} \to \vec{x}) = P(\vec{x})/P(\vec{y})$
  - –if P(y) < P(x), $P_{\rm accept}(\vec{x} \to \vec{y}) = P(\vec{y})/P(\vec{x})$,

    $P_{\rm accept}(\vec{y} \to \vec{x}) = 1$

Probabilistic methods

- The change in number density of walkers at x when all the walkers in the ensemble take a step:

$$\Delta\rho(\vec{x}) = \int d^d y \left[\rho(\vec{y})T(\vec{y}\to\vec{x}) - \rho(\vec{x})T(\vec{x}\to\vec{y})\right]$$

$$= \rho(\vec{x}) \int d^d y \, T(\vec{y}\to\vec{x}) \left[\frac{\rho(\vec{y})}{\rho(\vec{x})} - \frac{P(\vec{y})}{P(\vec{x})}\right] .$$

- If rho(y)/rho(x) exceeds “equilibrium value” P(y)/P(x):
  - will get a + correction
- If rho(y)/rho(x) is too small :
  - will get a - correction
- If rho is proportional to P, we’ll have “detailed balance”:

$$\rho(\vec{y})T(\vec{y}\to\vec{x}) = \rho(\vec{x})T(\vec{x}\to\vec{y})$$

Probabilistic methods

- Choices to make :
  - Step size
  - Thermalization steps
  - Sampling frequency

- All of these depend on the problem at hand
- You have to generally use trial and error to make sure there aren’t large dependencies on the parameters

Probabilistic methods

- An excellent example of using probabilistic methods is in statistical mechanics
- Makes sense : it's all about probability and statistics!

- A few definitions :
  - Microstates : configuration of a system
  - Probability of ith microstate is

$$p_i = \lim_{\mathcal{N} \to \infty} \frac{n_i}{\mathcal{N}} ,$$

Number of systems in the ith microstate

  - Average value (e.g. for energy) i $\langle E \rangle = \sum_i p_i E_i$.
  - Variance is $\langle (E - \langle E \rangle)^2 \rangle = \sum_i p_i E_i^2 - \left(\sum_i p_i E_i \right)^2$.

Probabilistic methods

- Examine the canonical and microcanonical ensembles:
  - Canonical :
    - fixed number of constituents (N), fixed volume (V)
    - exchange energy with thermal reservoir at temperature T
    - Probability of a microstate with energy E is Bolzmann distribution:

$$p_i = \frac{e^{-E_i/(k_\mathrm{B}T)}}{Z} \,,$$

    - Partition function is:

$$Z(N,V,T) = \sum_i e^{-E_i/(k_\mathrm{B}T)} \,,$$

  - Microcanonical :
    - fixed number of constituents (N), fixed total energy (E), fixed volume (V)
    - Probability of a microstate with energy E is :

$$p_i = \begin{cases} \frac{1}{N} & \text{if } E = E_i \\ 0 & \text{otherwise} \end{cases} .$$

Probabilistic methods

- Example of a system we can investigate is the hard-disk gas in two dimensions
  - Investigated by Metropolis et al with the MC methods above
  - J. Chem. Phys. 21, 1087 (1953)
  - "Equation of State Calculations by Fast Computing Machines"
    - "Fast" here meaning it could multiply two 40-bit ints in 1 millisecond (1 kHz)
    - For contrast, one core on your phone is 2.4 million times faster

Probabilistic methods

- Look at the system as a Maxwell-Boltzmann gas at fixed volume and temperature
- Energy is $$E = \text{K.E.} + \text{P.E.} = \frac{m}{2}\sum_i \mathbf{v}_i^2 + \sum_{\text{pairs } ij} U(r_{ij}) \, ,$$
- Pairwise potential energy function :

$$U(r) = \begin{cases} 0 & \text{if } r > \sigma \\ \infty & \text{if } r \leq \sigma \end{cases} \, ,$$

- In an ensemble of systems, at temperature T, probability that the system has energy E is a Boltzmann distribution

$$\sim W \exp\left[-\frac{E}{k_{\mathrm{B}}T}\right] = \exp\left[-\frac{E-TS}{k_{\mathrm{B}}T}\right] = \exp\left[-\frac{F}{k_{\mathrm{B}}T}\right] \, ,$$

Number of microstates

Entropy

Free energy

Probabilistic methods

- Partition function is

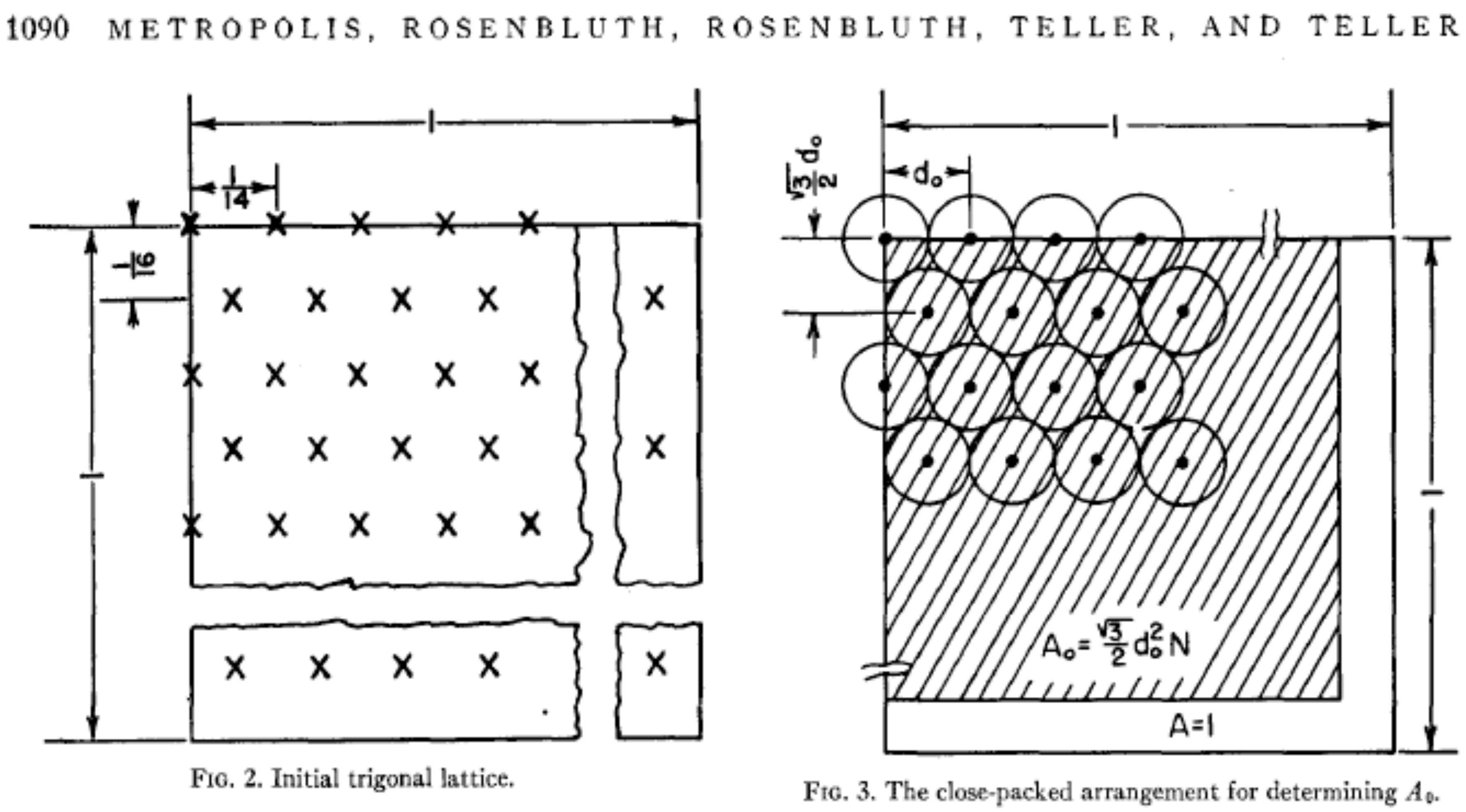
$$Z = \sum_E W(E) \exp\left[-\frac{E}{k_{\mathrm{B}}T}\right] .$$

- The equation of state relates pressure p, volume V and temperature T:

$$pV = N k_{\mathrm{B}} T \left. \frac{\partial \log Z}{\partial \log V} \right|_{T,N} .$$

Probabilistic methods

- MC simulation of hard disks from Metropolis et al:
  - Close-pack N=224 disks of diameter d0 as follows



1090 METROPOLIS, ROSENBLUTH, ROSENBLUTH, TELLER, AND TELLER

l
$\frac{l}{14}$
$\frac{l}{16}$
l
$\frac{\sqrt{3}}{2}d_0$
d_0
l
$A_0=\frac{\sqrt{3}}{2}d_0^2N$
A=1
l

Fig. 2. Initial trigonal lattice.

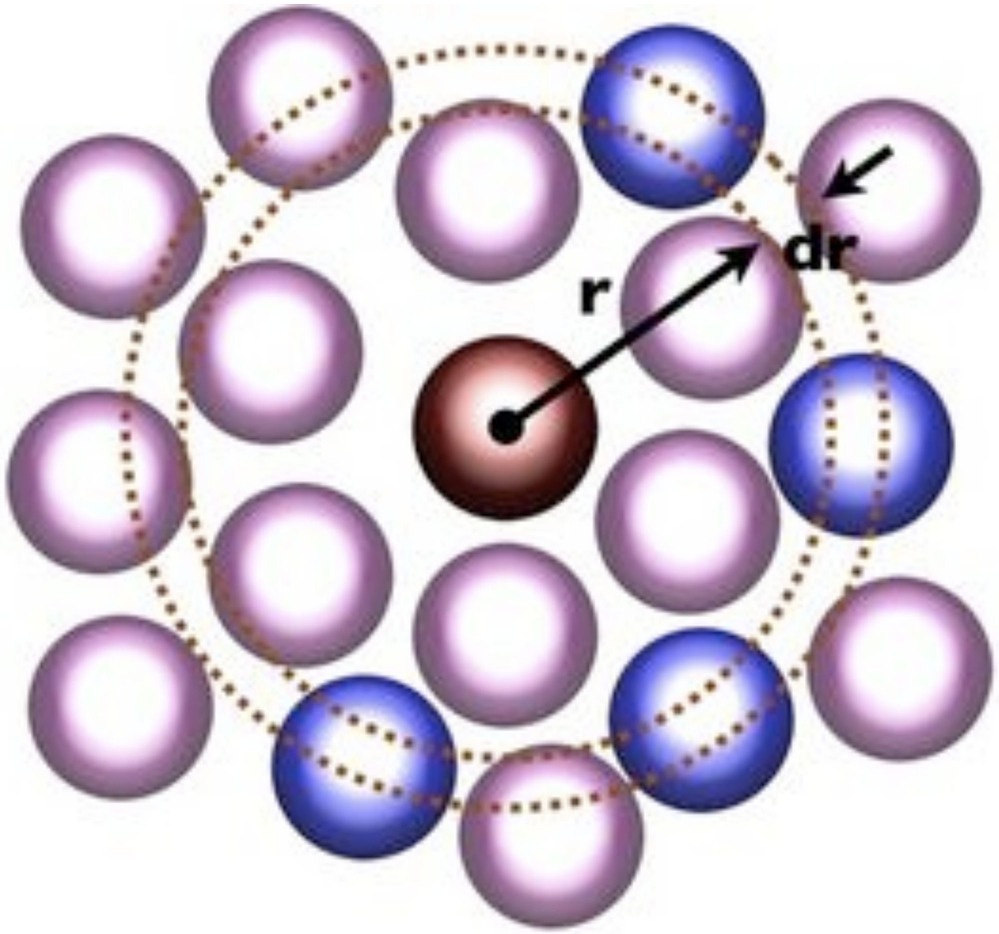
Fig. 3. The close-packed arrangement for determining A_0.

- Fix $A_0 = \sqrt{3}d_0^2N/2 = 1$ and vary d0, hence number is N

- Apply periodic boundary conditions

Probabilistic methods

- Radial distribution function measures correlations between particles separated at distance r
- Can be used to distinguish solid, liquid, gases
  - See Gould-Tobochnik Chapter 8 Section 8.5
- Equation of state is deduced from the radial distribution function:



CALCULATION OF STATE BY FAST MACHINES 1091

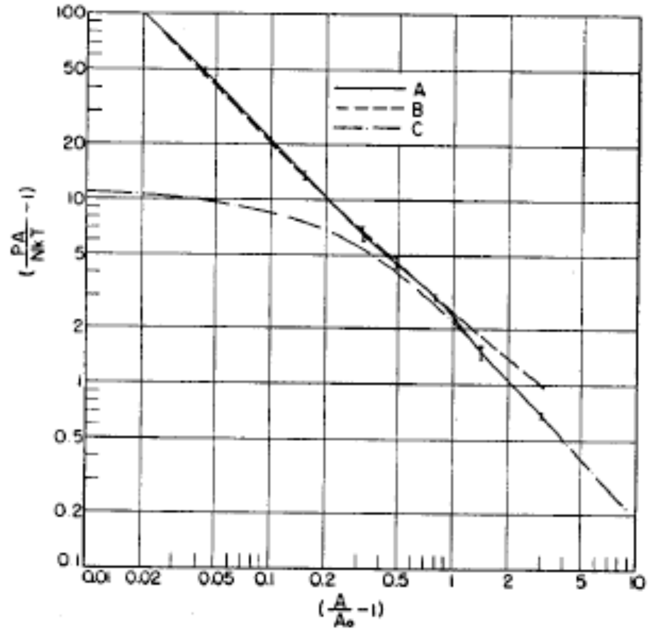


Fig. 4. A plot of $(PA/NkT)-1$ *versus* $(A/A_0)-1$. Curve A (solid line) gives the results of this paper. Curves B and C (dashed and dot-dashed lines) give the results of the free volume theory and of the first four virial coefficients, respectively.

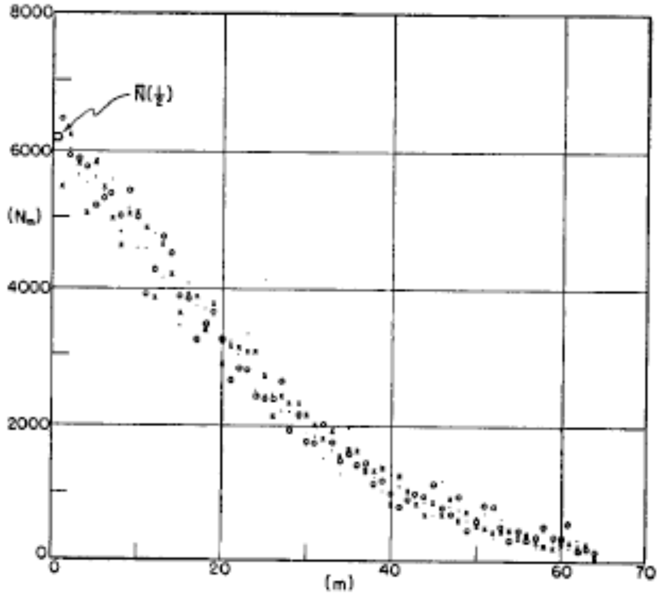


Fig. 5. The radial distribution function N_m for $\nu=5$, $(A/A_0)=1.31966$, $K=1.5$. The average of the extrapolated values of $N_{\frac{1}{2}}$ in $\bar{N}_{\frac{1}{2}}=6301$. The resultant value of $(PA/NkT)-1$ is $64\bar{N}_{\frac{1}{2}}/N^2(K^2-1)$ or 6.43. Values after 16 cycles, ●; after 32, ×; and after 48, ○.

Probabilistic methods

- They showed

$$PA = Nk_{\mathrm{B}}T\left(1+\frac{\pi d_0^2\overline{n}}{2}\right) , \quad \text{where} \quad \overline{n} = n(d) ,$$

- Using the Virial Theorem :
  - States that $\mathrm{K.E.} = N \times \frac{1}{2}m\overline{v}^2 = pA + \frac{1}{2}\left\langle \sum_i \mathbf{r}_i \cdot \mathbf{X}_i^{\mathrm{int}} \right\rangle .$

- Derivation is based on this figure, defines variables involved in a collision between two disks:

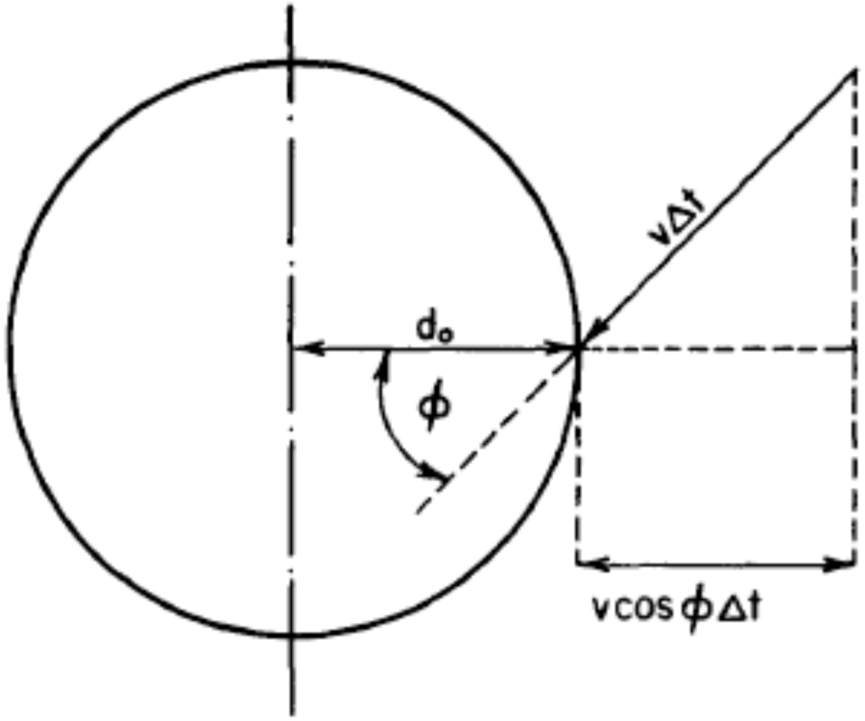


FIG. 1. Collisions of rigid spheres.

Probabilistic methods

- They showed

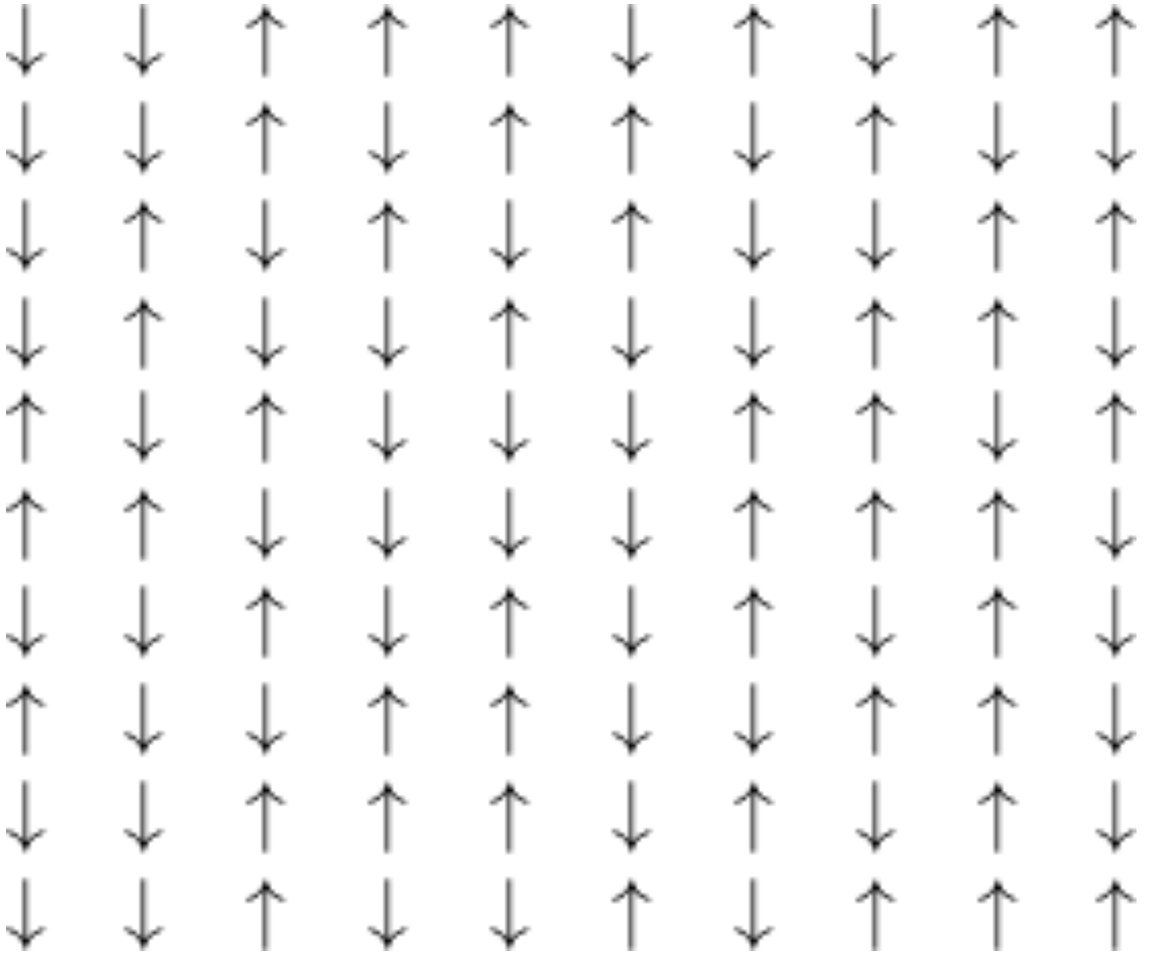
$$\left\langle \sum_i \mathbf{r}_i \cdot \mathbf{X}_i^{\mathrm{int}} \right\rangle = -\frac{1}{2} \sum_i \sum_{j \neq i} r_{ij} F_{ij} = N \times \frac{1}{2} m \overline{v}^2 \times \pi d_0^2 \overline{n} \, .$$

- MC simulation measured the radial distribution function as a histogram:
  - For each MC configuration
    - For each disk:
      - Divide region from r=d to r=rmax into 64 annular zones of equal area
      - Count disks in each zone, store in histogram
  - Average over configurations
  - Fit histogram to model function and extrapolate to r=d

Probabilistic methods

- Another very popular application of probabilistic methods is the simulation of idealized magnetic materials
- We use the Ising Model
  - http://en.wikipedia.org/wiki/Ising_model
- William Lenz assigned this to his student Ernst Ising as a PhD thesis problem
- This was solved by Ising exactly for 1-d
  - Didn't exhibit ferromagnetism
  - Details of 1-d Ising model : Gould-Tobochnik Chapter 5.

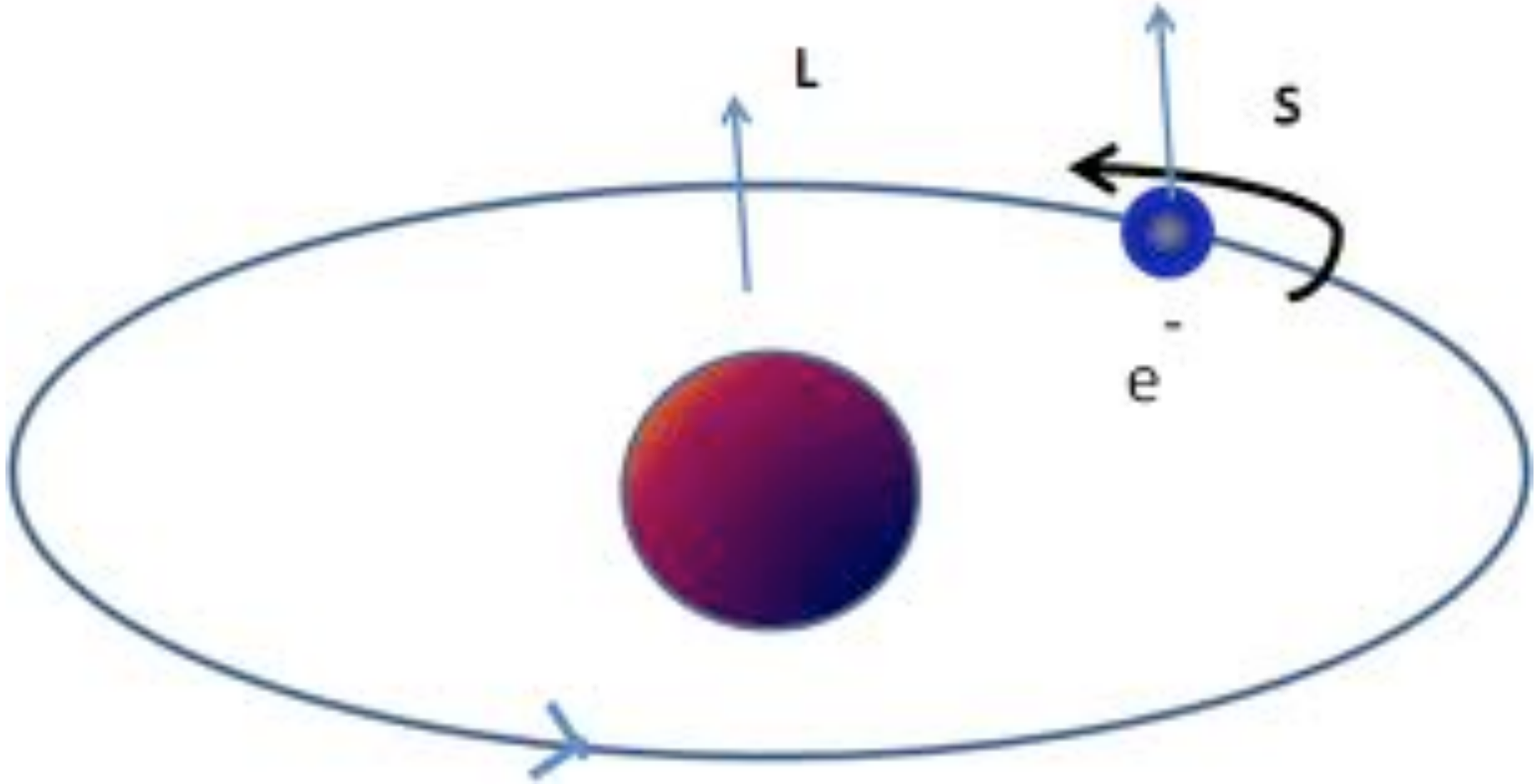
- Kramers, Wannier, Onsanger, et al worked on the 2-d model
  - Richer phenomenology : ferromagnetic phase at low temperatures, paramagnetic phase at high temperatures
  - Second order phase transition at the Curie temperature T_C



Probabilistic methods

- Recall : magnetism caused by charged particles "spinning" in closed orbits or about their axes
- For elementary particles, of course, we mean "spinning" in the quantum mechanical sense, not the rotational Newtonian sense

- So atoms will have both an "L" and and "S" contribution to the magnetic properties



Probabilistic methods

- We’ll take a simple classical approximation called the “Ising spin” which has two values :

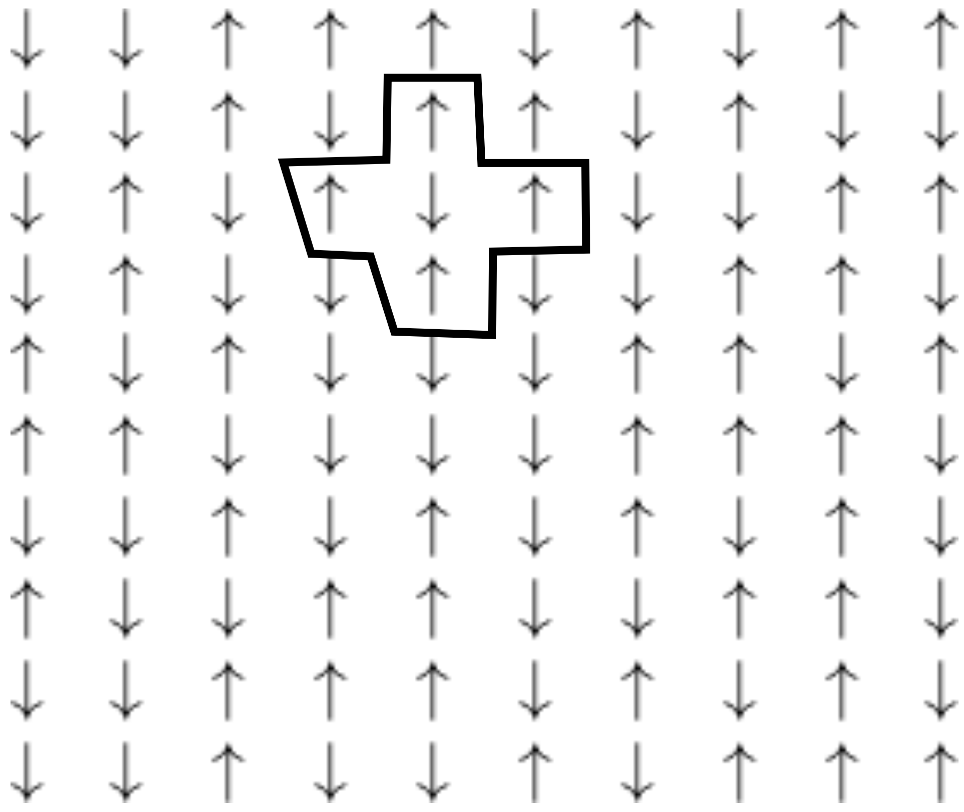
$$s_i = \begin{cases} +1, & \text{represents “spin up”} \uparrow \\ -1, & \text{represents “spin down”} \downarrow \end{cases}$$

- 2-d magnet is a set of Ns spins on a lattice

$$\begin{array}{cccccccccc}
\downarrow & \downarrow & \uparrow & \uparrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \uparrow \\
\downarrow & \downarrow & \uparrow & \downarrow & \uparrow & \uparrow & \downarrow & \uparrow & \downarrow & \downarrow \\
\downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \downarrow & \uparrow & \uparrow \\
\downarrow & \uparrow & \downarrow & \downarrow & \uparrow & \downarrow & \downarrow & \uparrow & \uparrow & \downarrow \\
\uparrow & \downarrow & \uparrow & \downarrow & \downarrow & \downarrow & \uparrow & \uparrow & \downarrow & \uparrow \\
\uparrow & \uparrow & \downarrow & \downarrow & \downarrow & \downarrow & \uparrow & \uparrow & \uparrow & \downarrow \\
\downarrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow \\
\uparrow & \downarrow & \downarrow & \uparrow & \uparrow & \downarrow & \downarrow & \uparrow & \uparrow & \downarrow \\
\downarrow & \downarrow & \uparrow & \uparrow & \uparrow & \downarrow & \uparrow & \downarrow & \uparrow & \downarrow \\
\downarrow & \downarrow & \uparrow & \downarrow & \downarrow & \uparrow & \downarrow & \uparrow & \uparrow & \uparrow
\end{array}$$

Probabilistic methods

- The force between magnets falls of like r^{-3}
- So, approximate that the spins only interact with their nearest neighbors :



Probabilistic methods

- The interaction energy is proportional to the alignment of the spins :

$$E = -J \sum_{\langle ij \rangle} s_i s_j - H \sum_i s_i \ .$$

- If J > 0 : ferromagnetic
  - Energy is minimized if the spins point in the same direction
- If J < 0 : antiferromagnetic
  - Energy is minimized if spins locally point in the opposite direction
- “H” is an external field which couples to the total magnetization: $M = \sum_i s_i \ ,$
- Spins will align to this external magnetization

Probabilistic methods

- For H=0, the system will be in one of two states :
  - Low temp (below Curie temperature) : magnetized
  - High temp (above Curie temperature) : sum of magnetization is zero
- Critical value is the second-order phase transition between ferromagnetic to paramagnetic phases

Probabilistic methods

- We want to compute the observables (averages over the samples)

- Define a state as a specific set of spin values (up or down)

$$s_1 = +1, s_2 = -1, s_3 = +1, \ldots, s_{N_s} = +1 \ ,$$

- Avg. of an observable calculated by weighting configurations by the Boltzmann factor
- Example : average magnetization :

$$\langle M \rangle = \frac{\sum_{\text{configs}} M e^{-E/k_{\text{B}}T}}{\sum_{\text{configs}} e^{-E/k_{\text{B}}T}} \ .$$

Probabilistic methods

- Total number of configurations is very large
- If we have N spins, we have 2^N configurations
- If L=20, N=20*20 = 400, so 2^{400} configurations (2.6e120 !!!)
- That's completely intractable to even list them, much less compute the total magnetization

- So instead, we use Monte Carlo methods
  - We generate a reasonable number of configurations at random
  - Use the Boltzmann factors to define the probability of each
  - Perfect for MC methods!

Probabilistic methods

- The probability for each state is :

$$p(s_1, s_2, \ldots, s_{N_s}) = \frac{e^{-E(s_1,s_2,\ldots,s_{N_s})/k_\mathrm{B}T}}{\sum_\mathrm{configs} e^{-E/k_\mathrm{B}T}} \ .$$

- Now, we need to generate N statistically independent configurations according to this probability
- The average magnetization and energy are :

$$\langle M \rangle = \frac{1}{N} \sum_{i=1}^{N} M(s_1^{(i)}, s_2^{(i)}, \ldots, s_{N_s}^{(i)}) \ ,$$

$$\langle E \rangle = \frac{1}{N} \sum_{i=1}^{N} E(s_1^{(i)}, s_2^{(i)}, \ldots, s_{N_s}^{(i)}) \ .$$

Probabilistic methods

- Now use the Metropolis algorithm from last class to “map out” the configuration space
  - Choose initial configuration
  - For all spins :
    - Trial flip (+1 --> -1 or vice versa)
    - Compute change in energy
    - If $w = e^{-\Delta E / k_{\mathrm{B}} T} > r$, flip the spin (where r is a uniform deviate)
- Make sure to pick out the thermalization and skip steps

Probabilistic methods

- IT is expensive to compute all of the Boltzmann factors at each step
- Instead, we can realize that for a simulation at fixed T and H, there are only 10 distinct values
- Can precompute them and then just refer to that
- First, consider the sum of the 4 neighboring spins:

$$\sum_{\text{neighbors } j} s_j = s_{(i_x+1,i_y)} + s_{(i_x-1,i_y)} + s_{(i_x,i_y-1)} + s_{(i_x,i_y+1)} \ .$$

- The value is
  - $\sum_{\text{neighbors } j} s_j = +4$ if all four neighbors point up,
  - $\sum_{\text{neighbors } j} s_j = +2$ if three neighbors point up and one down,
  - $\sum_{\text{neighbors } j} s_j = 0$ if two neighbors point up and two down,
  - $\sum_{\text{neighbors } j} s_j = -2$ if one neighbor points up and three down,
  - $\sum_{\text{neighbors } j} s_j = -4$ if all four neighbors point down.
- Product can only have one of these five values!

$$s_i \sum_{\text{neighbors } j} s_j = +4, +2, 0, -2, -4 \ .$$

Probabilistic methods

- Now, consider H != 0
- Have terms like H*s_i
- For those we get the other 5 terms
- Can store this as a 2-d array of 5 rows and 2 columns
  - First index : $$2 + \frac{1}{2} s_i \sum_{\text{neighbors } j} s_j = 0, 1, 2, 3, 4 \,,$$
  - Second index : $$\frac{1 + s_i}{2} = 0, 1 \,.$$

Probabilistic methods

- Metropolis step :
  - Choose a spin at random
  - Take the metropolis step
  - If the spin is at the boundary, use periodic boundary conditions

- Because the MC step of a single spin-flip will be highly-correlated with “this” step, we need to make sure to choose sufficiently high sampling factors
  - Typically need AT LEAST N steps
  - Each spin has a chance to flip

Probabilistic methods

- Below the Curie temperature, Kramers and Wannier showed that there is a duality argument to compute the exact value : Phys. Rev. 60, 252 (1941),

$$\frac{k_\mathrm{B}T_\mathrm{c}}{J} = \frac{2}{\log(1+\sqrt{2})} = 2.269\ldots$$

- Osnager showed the same model in the thermodynamic limit N-> infinity with H=0 (Phys. Rev. 65, 117 (1944))

$$m = \lim_{N\to\infty} \frac{\langle \sum_i s_i \rangle}{N} = \begin{cases} \left[1 - \left\{\sinh\left(\frac{2J}{k_\mathrm{B}T}\right)\right\}^{-4}\right]^{1/8}, & \text{for } T \le T_\mathrm{c} \\ 0, & \text{for } T > T_\mathrm{c} \end{cases}$$

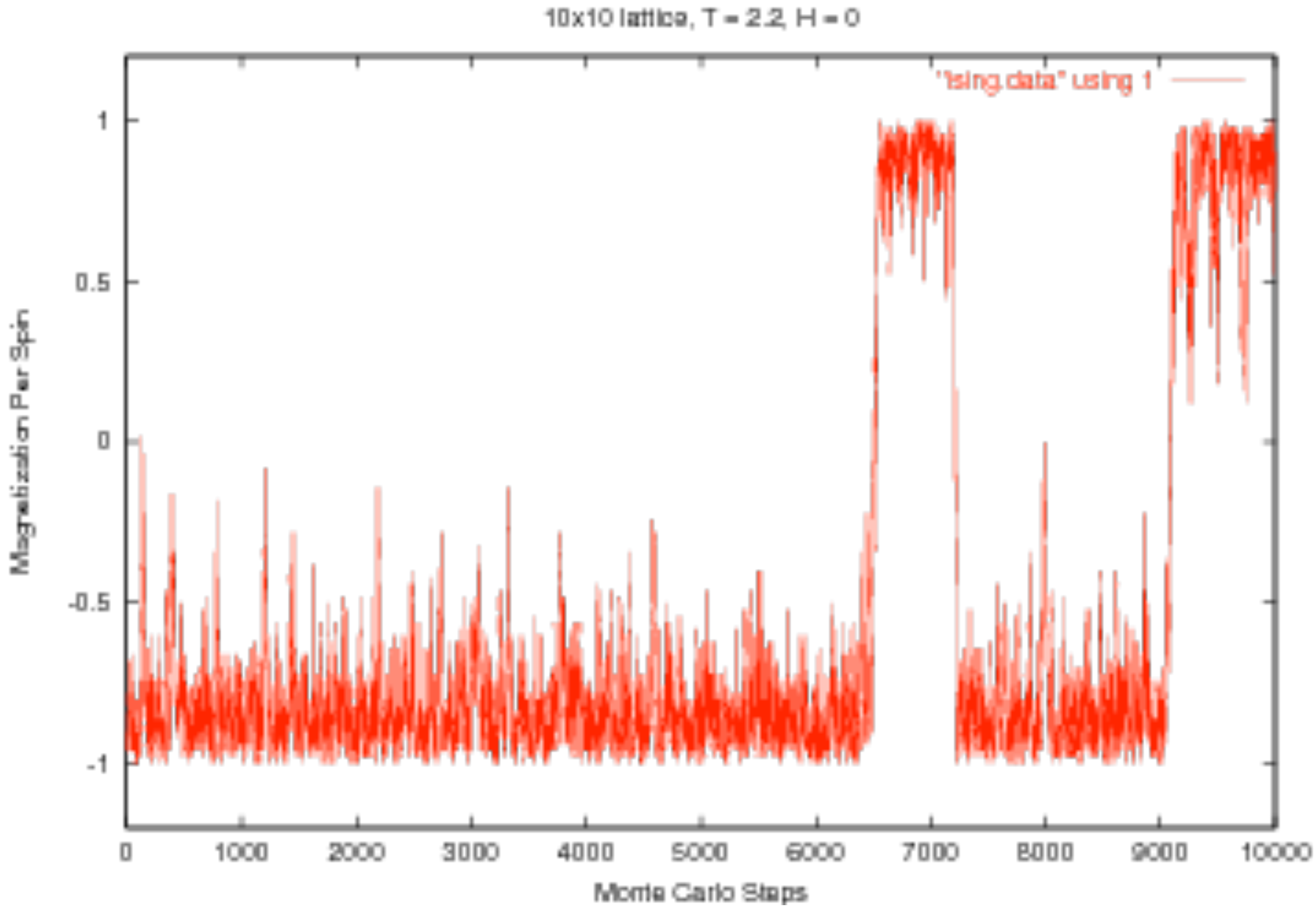
- Near the Curie temperature

$$m \sim (T_c - T)^\beta \ ,$$

- Beta = 1/8 for the 2-d Ising model

Probabilistic methods

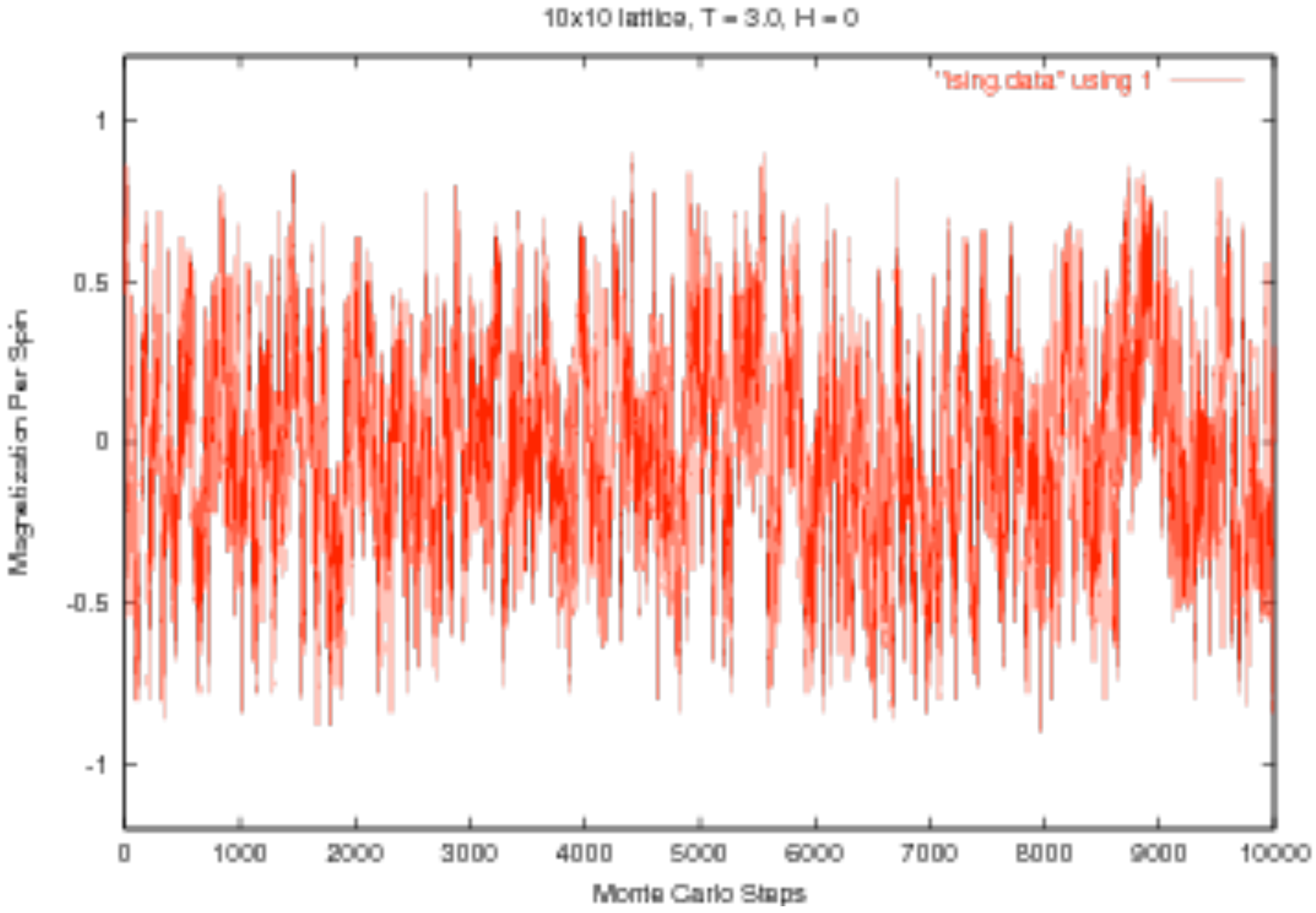
- Here are some numerical results for T = 2.2, below Tc



- Note : for finite systems, spins can flip from time to time!

Probabilistic methods

- And now for T=3.0, above Tc:


10x10 lattice, T = 3.0, H = 0
"ising.data" using 1
Magnetization Per Spin
Monte Carlo Steps