COMPUTATIONAL PHYSICS

Shuai Dong

Dice

骰子

Monte Carlo simulations

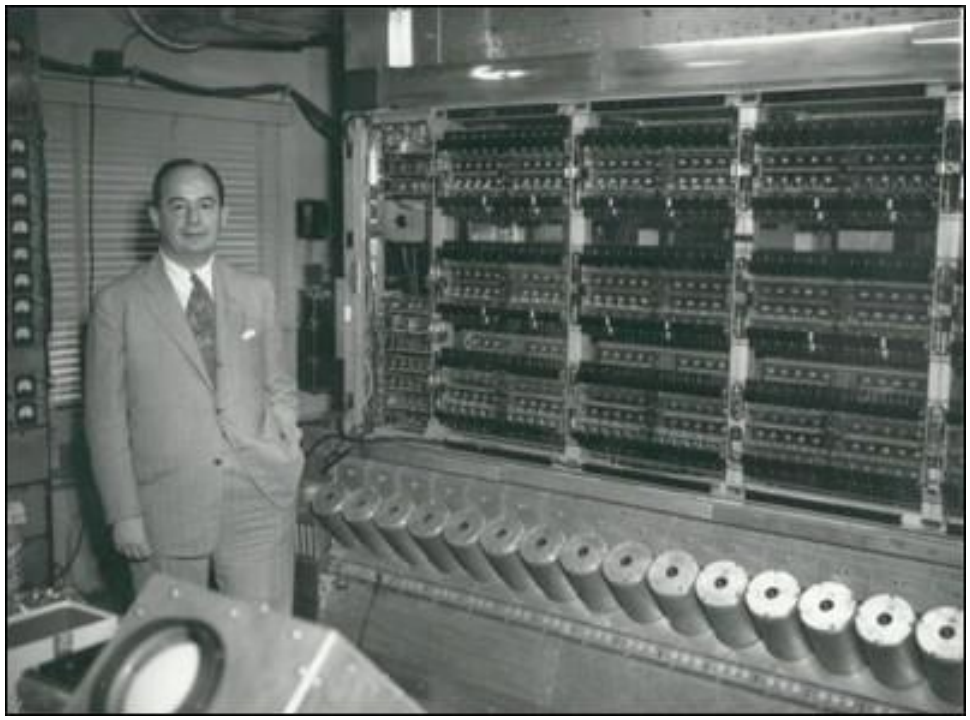
- Sampling and integration
- Darts method to calculate π
- Random-number generators
- The Metropolis algorithm
- Ising model

What is Monte Carlo?

Monte Carlo Casino

Monte-Carlo, Monaco

Monte Carlo method: computational statistics
Birthday: World War II, 1940s
Father: S. Ulam
Birth Place: neutron reaction in atom bomb, Manhattan Project



Von Neumann
(1903-1957)

Stanislaw Ulam
(1909-1984)

Principle

Physical Problems

⇨ Simulated by Random Processes

Why is it called Monte Carlo?

⇨ Numerical solutions

Sampling and integration

If we want to find the numerical value of the integral

$$S = \int_0^1 f(x)\,dx,$$

Divide the region [0, 1] evenly into *M* slices with $x_o = 0$ and $x_M = 1$, and then the integral can be approximated as

$$S = \frac{1}{M}\sum_{n=1}^{M} f(x_n) + O(h^2),$$

which is equivalent to sampling from a set of points $x_1, x_2, \ldots, x_M$ in the region[0, 1] with an equal weight, in this case, 1, at each point.

We can also select x_n with $n = 1,2,\ldots, M$ from a uniform random number generator in the region [0,1] to accomplish the same goal.

If M is very large, we would expect x_n to be a set of numbers uniformly distributed in the region [0, 1] with fluctuations proportional to 1/sqrt(M). Then the integral can be approximated by the average

$$S \simeq \frac{1}{M} \sum_{n=1}^{M} f(x_n)$$

where x_n is a set of M points generated from a uniform random number generator in the region [0,1].

Example

- In order to demonstrate the algorithm clearly, let us take a very simple integrand $f(x) = x^2$ from 0 to 1.

- The exact result of the integral is 1/3.

- The following program implements such a sampling.
  - 8.1.MC.cpp
  - 8.rand.h

A history of π (pie):

Time	Who	Value
~2000-1850 BC	Egyptian 阿美斯纸草书	256/81=3.160493....
~1900 BC	Babylonian	25/8=3.125
~900 BC	Indian (Shatapatha Brahmana)	339/108=3.13888
~600 BC	Hebrew 希伯来圣经	3
~250 BC	Αρχιμήδης	3.14163491
~100 BC ?	Chinese 周髀算经	3 径一而周三

Time	Who	Value
~20 BC	Vitruvius	3.125
~50-23 BC	刘歆	3.1547
~130 AD?	张衡	3.146551
~150 AD	Πτολεμαῖος	3.141666
250 AD	王蕃	3.155555
263 AD	刘徽	$3.141024 < \pi < 3.142104$
400 AD	何承天	111035/35329 = 3.142885…
480 AD	祖冲之	$3.1415926 < \pi < 3.1415927$

A record for one thousand years!!!

Zu Chongzhi

- Born in Nanjing! (南朝刘宋）
- 大明历
- π=355/113 (祖率)
- The record holds for about 1000 years.
- Broken by Jamshid Masud Al Kashi (Arab) in 1424.

World-leading scientists were very rare in the Chinese long history!

The history of science, is a flag of human civilization!

Chinese name in Moon:

石申、张衡、祖冲之、郭守敬、万户

The first rocket man!

Chinese name in Mercury:
李清照

- We have largest population!
- We have one of longest history and unbroken culture!
- We have the 2nd GPD now!
- We have the 1st-3rd golden medals in Olympic games!
- We have the Nobel laureates in Peace, Literature, & Physiology or Medicine!

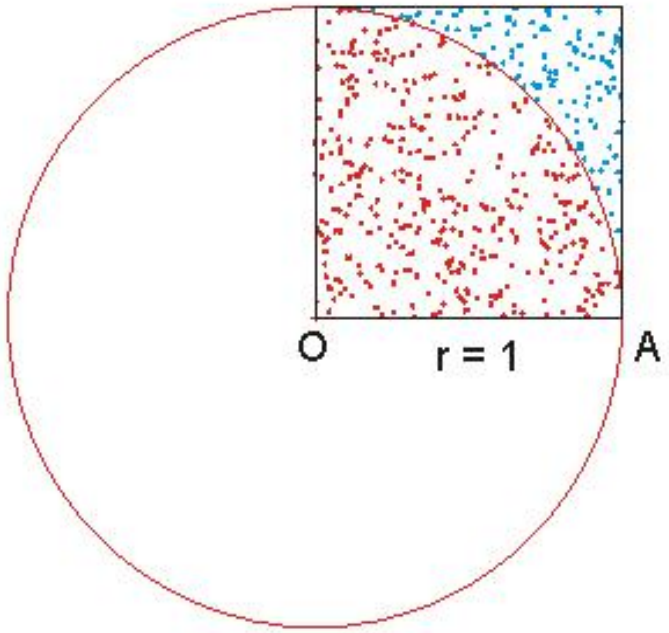
What don't we have?
Scientists & philosophers!

We need some people looking at sky!

Darts method to calculate Pi

Circle: $\pi r^2/4=\pi/4$

Square: $r^2=1$



$$\frac{\text{Darts in Circle}}{\text{Total Darts}} = \frac{\pi}{4}$$

8.2.Darts.cpp

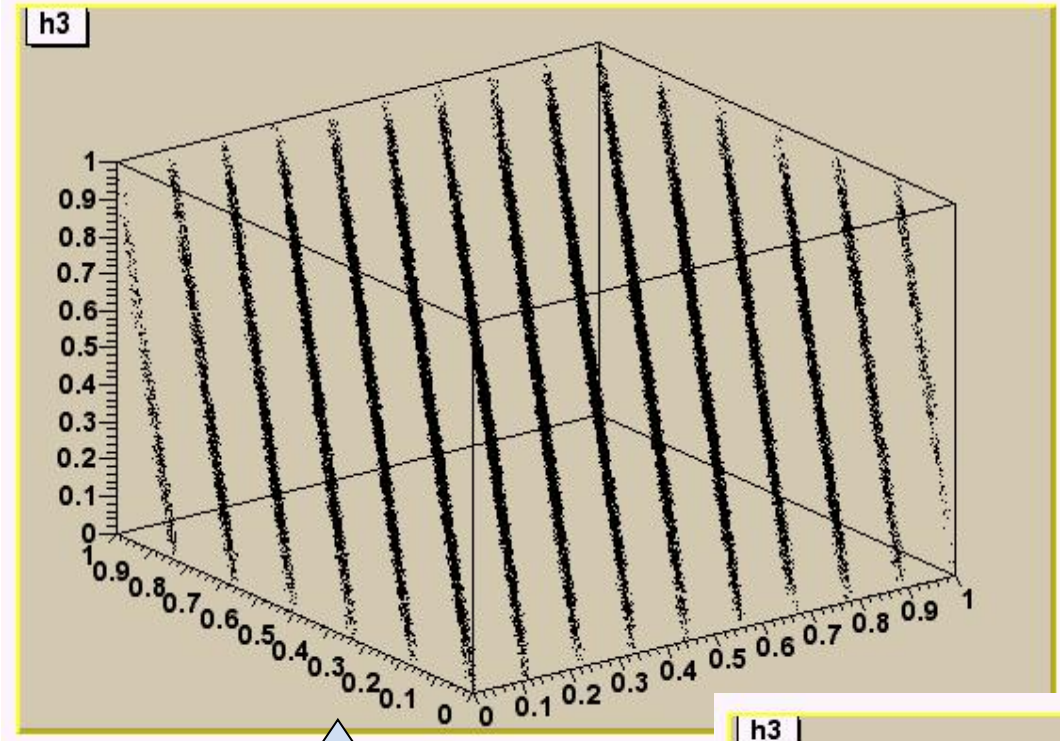
Random-number generators

Not a number but a sequence with random numbers

Random number generators:

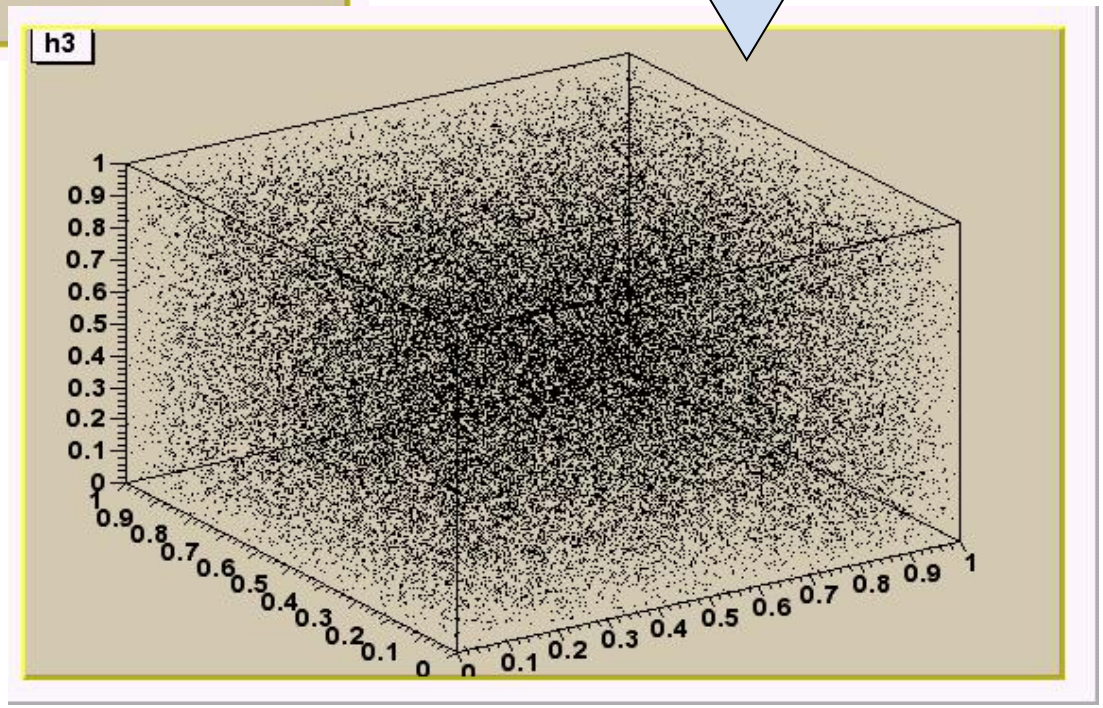
- Real random number (RRN) generators: Quantum processes like nuclear decay

- Pseudo-random number (PRN) generators: Lottery, computational PRN: depends on the initial one, algorithm, and word length



Good PRN

Bad PRN



PRN function in C/C++

void srand(unsigned int *seed*);
int rand(void)

The rand function returns a pseudorandom integer in the range 0 to RAND_MAX.

Use the srand function to seed the pseudorandom-number generator before calling rand.

Or you will always get the same number sequence!

1.How to generate a real PRN in 0-1?

- rand()/32767
- 1.0*rand()/32767
- 1.0*rand()/RAND_MAX

8.3.Rand.cpp

2.How to generate an integer PRN in 0-6?

3.How to generate a binary PRN -1/1?

Importance sampling

- The accuracy of the integral evaluated from the Monte Carlo quadrature is very low.

- Is there any way to increase the accuracy for some specific types of integrands?

- If f (x) is smooth and close to constant, the accuracy from the Monte Carlo quadrature would be much higher.

The $3N$-dimensional integral is written as

$$S = \int_D F(\mathbf{R})\, d\mathbf{R}$$

where D is the domain of the integral, $\mathbf{R} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$, with each $\mathbf{r}_i$ for i = 1, 2, . . . , N being a three-dimensional vector.

In many cases, the function $F(\mathbf{R})$ is not a smooth function. The idea of importance sampling introduced by Metropolis *et al.* (1953) is to sample the points from a nonuniform distribution.

If a distribution function $W(\mathbf{R})$ can mimic the drastic changes in $F(\mathbf{R})$, we should expect a much faster convergence with

$$S \approx \frac{1}{M} \sum_{i=1}^{M} \frac{F(R_i)}{W(R_i)} W(R_i)$$

where M is the total number of points sampled according to the distribution function $W(\mathbf{R})$. Rewrite the integral as

$$S = \int \mathcal{W}(\mathbf{R}) G(\mathbf{R})\, d\mathbf{R}$$

where $W(\mathbf{R})$ is positive definite and satisfies the normalization condition.

$$\int \mathcal{W}(\mathbf{R})\, d\mathbf{R} = 1$$

From the two expressions of the integral we have $G(\mathbf{R}) = F(\mathbf{R})/W(\mathbf{R})$.

We can imagine a statistical process that leads to an equilibrium distribution $W(\mathbf{R})$ and the integral S is merely a statistical average of $G(\mathbf{R})$.

This can be compared with the canonical ensemble average.

$$\langle A \rangle = \int A(\mathbf{R}) \mathcal{W}(\mathbf{R})\, d\mathbf{R}$$

where $A(\mathbf{R})$ denotes the physical quantities to be averaged.

The probability or distribution function $W(\mathbf{R})$ is given by

$$\mathcal{W}(\mathbf{R}) = \frac{e^{-U(\mathbf{R})/k_{\mathrm{B}}T}}{\int e^{-U(\mathbf{R}')/k_{\mathrm{B}}T}\, d\mathbf{R}'}$$

with $U(\mathbf{R})$ being the potential energy of the system for a given configuration $\mathbf{R}$. Here k_{B} is the Boltzmann constant and T is the temperature of the system.

A procedure introduced by Metropolis *et al.* (1953) is extremely powerful in the evaluation of the multidimensional integral defined above.

Metropolis algorithm

Nicholas Metropolis
(1915-1999)

The algorithm by Metropolis (and A Rosenbluth, M Rosenbluth, A Teller and E Teller, 1953) has been cited as among the top 10 algorithms having the "greatest influence on the development and practice of science and engineering in the 20th century."

Great algorithms are the poetry of computation

Says Francis Sullivan of the Institute for Defense Analyses. He and other folks from the University of Tennessee and Oak Ridge National Laboratory have put together a list of 10 algorithms having "the greatest influence on the development and practice of science and engineering in the 20th century".

List in Computing in Science & Engineering

- 1946: The Metropolis Algorithm for Monte Carlo.
- 1947: Simplex Method for Linear Programming.
- 1950: Krylov Subspace Iteration Method.
- 1951: The Decompositional Approach to Matrix Computations.
- 1957: The Fortran Optimizing Compiler.
- 1959: QR Algorithm for Computing Eigenvalues.
- 1962: Quicksort Algorithms for Sorting.
- 1965: Fast Fourier Transform.
- 1977: Integer Relation Detection.
- 1987: Fast Multipole Method.

Metropolis algorithm

- The selection of the sampling points is viewed as a Markov process.
- In equilibrium, the values of the distribution function at different points of the phase space are related by

$$\mathcal{W}(\mathbf{R})T(\mathbf{R} \rightarrow \mathbf{R}') = \mathcal{W}(\mathbf{R}')T(\mathbf{R}' \rightarrow \mathbf{R})$$

where $T(\mathbf{R} \rightarrow \mathbf{R}')$ is the transition rate from the state characterized by **R** to the state characterized by **R'**.
This is usually referred to as detailed balance in statistical mechanics.

Now the points are no longer sampled randomly but rather by following the Markov chain.

The transition from one point **R** to another point **R'** is accepted if the ratio of the transition rates satisfies

$$\frac{T(\mathbf{R} \to \mathbf{R}')}{T(\mathbf{R}' \to \mathbf{R})} = \frac{\mathcal{W}(\mathbf{R}')}{\mathcal{W}(\mathbf{R})} \geq w_i$$

where w_i is a uniform random number in the region [0, 1].

Outline of the steps

We first randomly select a configuration R_0 inside the specified domain *D*.

Then $W(R_0)$ is evaluated. A new configuration R_1 is tried with $R_1=R_0+\Delta R$, where ΔR is between $[-h, h]$. The actual value of *h* is determined from the desired accepting rate.
In practice, *h* is commonly chosen so that the accepting rate of moves is around 50%.

We can evaluate the probability $$p = \frac{\mathcal{W}(\mathbf{R}_1)}{\mathcal{W}(\mathbf{R}_0)}$$

Comparing p with a uniform random number w_i in the region [0, 1]. If $p \geq w_i$, the new configuration is accepted; otherwise, the old configuration is assumed to be the new configuration.

This procedure is repeated and the physical quantity $A(R_k)$ for $k = n_1, n_1 + n_0, \ldots, n_1 + (M - 1)n_0$ is evaluated. The numerical result of the integral is then given by

$$\langle A \rangle \simeq \frac{1}{M} \sum_{l=0}^{M-1} A(\mathbf{R}_{n_1+n_0 l})$$

Note that the first n_1 steps are used to remove the influence of the initial selection. The data are taken n_o steps apart to avoid high correlation between the data points.

In most cases, the distribution function $W(\mathbf{R})$ varies by several orders of magnitude, whereas $A(\mathbf{R})$ stays smooth or nearly constant.

This sampling by importance is much more efficient than the direct, random sampling presented in the preceding section.

Now we would like to compare this procedure numerically with the direct, random sampling. We still consider the integral

$$S = \int_0^1 f(x)\,dx = \int_0^1 \mathcal{W}(x) g(x)\,dx$$

with $f(x) = x^2$. We can choose the distribution function as

$$\mathcal{W}(x) = \frac{1}{\mathcal{Z}}\left(e^{x^2} - 1\right)$$

which is positive definite.

The normalization constant Z is given by

$$\mathcal{Z} = \int_0^1 \left(e^{x^2} - 1\right) dx = 0.462\,651\,67$$

which is calculated from another numerical scheme for convenience. Then the corresponding function $g(x) = f(x)/W(x)$ is given by

$$g(x) = \mathcal{Z} \frac{x^2}{e^{x^2} - 1}$$

Now we are ready to put all of these into a program that is a realization of the Metropolis algorithm for the integral specified.

Code example

8.4.Metropolis.cpp

Homework

- 1. Generate random integer numbers: 0, 1, 2 with the possibility 25%, 50%, 25%, respectively.

- 2. Generate random numbers from 0 to 2 with the given distribution: $\sim \exp[-(x-1)^2]$

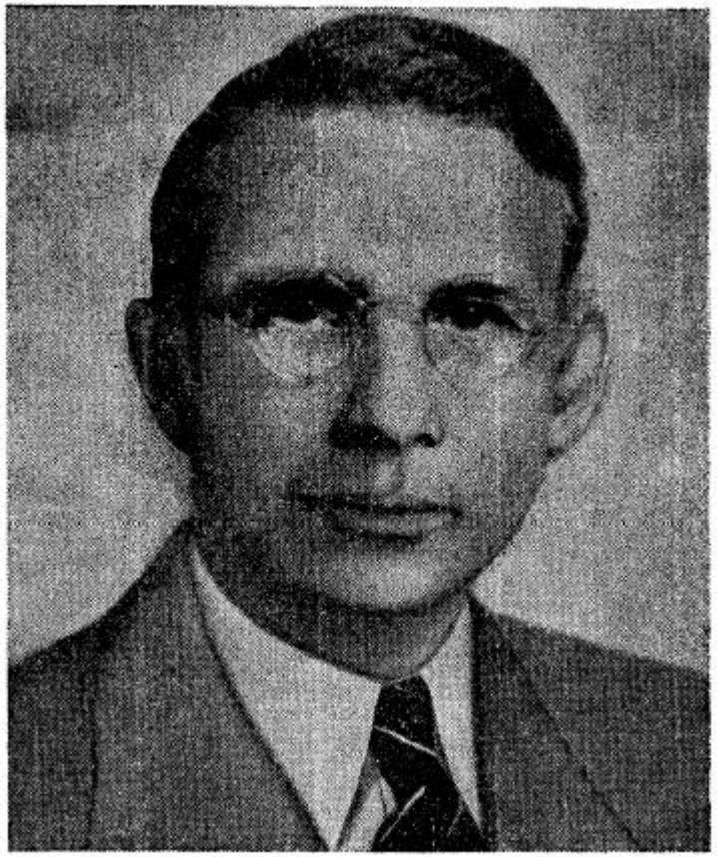
Applications in statistical physics

In this section, we discuss some applications of the Metropolis algorithm in statistical physics.
We will use the Ising model as the illustrative example.
The Ising model is a discrete lattice system.

Ising model

Wilhelm Lenz 1888-1957

1920

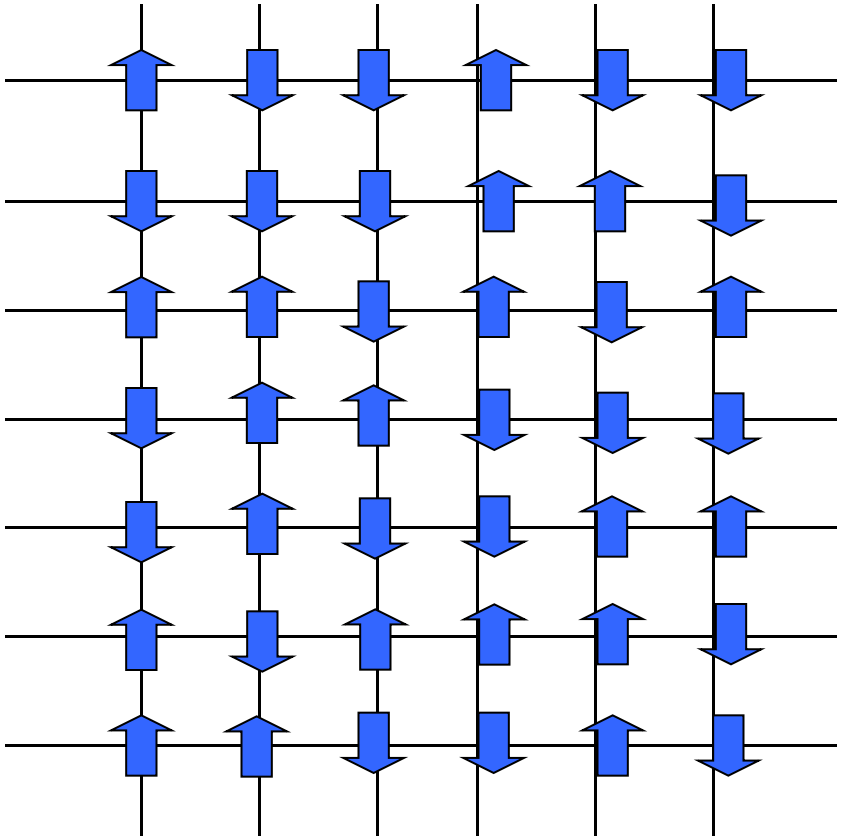


Ernst Ising, 1900–1998

1924

- Ernst Ising was born in Cologne in 1900. After school, he studied physics and mathematics at the University of Göttingen and University of Hamburg. In 1922, he began researching ferromagnetism under the guidance of Wilhelm Lenz. He earned a Ph.D in physics from the University of Hamburg in 1924 when he published his doctoral thesis (an excerpt or a summary of his doctoral thesis was published as an article in a scientific journal in 1925 and this has led many to believe that he published his full thesis in 1925). His doctoral thesis studied a problem suggested by his teacher, Wilhelm Lenz. He investigated the special case of a linear chain of magnetic moments, which are only able to take two positions, "up" and "down," and which are coupled by interactions between nearest neighbors. Mainly through following studies by Rudolf Peierls, Hendrik Kramers, Gregory Wannier and Lars Onsager the model proved to be successful explaining phase transitions between ferromagnetic and paramagnetic states.

- After earning his doctorate, Ernst Ising worked for a short time in business before becoming a teacher, in Salem, Strausberg and Crossen, among other places. In 1930, he married the economist Dr. Johanna Ehmer (February 2, 1902 - February 2, 2012). As a young German-Jewish scientist, Ising was barred from teaching and researching when Hitler came to power in 1933. In 1934, he found a position, first as a teacher and then as headmaster, at a Jewish school in Caputh near Potsdam for Jewish students who had been thrown out of public schools. Ernst and his wife Dr. Johanna Ising, lived in Caputh near the famous summer residence of the Einstein family. In 1938, the school in Caputh was destroyed by the Nazis, and in 1939 the Isings fled to Luxembourg, where Ising earned money as a shepherd and railroad worker. After the German Wehrmacht occupied Luxembourg, Ernst Ising was forced to work for the army. In 1947, the Ising family emigrated to the United States. Though he became Professor of Physics at Bradley University in Peoria, Illinois, he never published again. Ising died at his home in Peoria in 1998, just one day after his 98th birthday.



Spin: $s_i = \pm 1$ $s = \{s_1, s_2, \ldots, s_i, \ldots\}$

Ising model Hamiltonian

$$H_{\text{Ising}} = -\sum_{i,j} J_{ij} S_i S_j - h \sum_i S_i$$

Total states: 2^N

Grains on chessboard

Any physical observation

$$< A(s) >_T = \frac{\sum_s \exp[-H(s)/k_B T] A(s)}{\sum_s \exp[-H(s)/k_B T]}$$

canonical ensemble

partition function

The Ising model was used historically to study magnetic phase transitions. The magnetization is defined as

$$m \simeq \frac{1}{M} \sum_{\sigma=1}^{M} s_\sigma$$

which is a function of the temperature and external magnetic field.

For the $h = 0$ case, there is a critical temperature T_C that separates different phases of the system. For example, the system is ferromagnetic if $T < Tc$, paramagnetic if $T > Tc$, and unstable if $T = Tc$. The complete plot of T , m, and h forms the so-called phase diagram.

Another interesting application of the Ising model is that it is also a generic model for binary lattices: that is, two types of particles can occupy the lattice sites with two on-site energies which differ by $2h$. So the results obtained from the study of the Ising model apply also to other systems in its class, such as binary lattices.

Other quantities of interest include, but are not limited to, the total energy $E = \langle H \rangle$ and the specific heat

$$C = \frac{\langle \mathcal{H}^2 \rangle - \langle \mathcal{H} \rangle^2}{N k_B T^2}$$

In order to simulate the Ising model, for example, in the calculation of m:

$$m = \frac{1}{\mathcal{Z}} \sum_{\sigma} s_{\sigma} e^{-\mathcal{H}_{\sigma}/k_{\mathrm{B}}T}$$

where $s_\sigma = S_\sigma / N$, with $S_\sigma = \Sigma s_i$ being the total spin of a specific configuration labeled by σ and H_σ being the corresponding Hamiltonian (energy).

The above summation is over all the possible configurations. Here Z is the partition function given by

$$\mathcal{Z} = \sum_{\sigma} e^{-\mathcal{H}_{\sigma}/k_{\mathrm{B}}T}$$

The average of a physical quantity, such as the magnetization, can be obtained from

$$m \simeq \frac{1}{M} \sum_{\sigma=1}^{M} s_\sigma$$

with $\sigma = 1, 2, \ldots, M$ indicating the configurations sampled according to the distribution function

$$\mathcal{W}(S_\sigma) = \frac{\exp[-\mathcal{H}_\sigma / k_B T]}{\mathcal{Z}}$$

Outline of the steps

- Start from a given spin configuration.
- Choose a spin.
- Change it.
- Calculate the energy change ΔE=new-old. Compare the possibility $p=\exp[-\Delta E/k_B T]$ with a PRN. If $p<$PRN, reject the change, else, accept the new configuration.
- Repeat the above processes for many steps.
- Perform measurements and calculate average.

Lars Onsager (November 27, 1903 – October 5, 1976) was a Norwegian-born American physical chemist and theoretical physicist, winner of the 1968 Nobel Prize in Chemistry. He held the Gibbs Professorship of Theoretical Chemistry at Yale University.

During the 1940s, Onsager studied the statistical-mechanical theory of phase transitions in solids. He obtained the exact solution for the two dimensional Ising model in zero field in 1944.

for 1D chain: $T_C=0$ (Ising proved)
for 2D square lattice:
$K_B T_C/J=2/\ln(1+\sqrt{2})=2.26918531421$ (Onsager proved)
for 3D lattice: ????

Code example

- 8.5.Ising.cpp
- 8.6.Flip.java

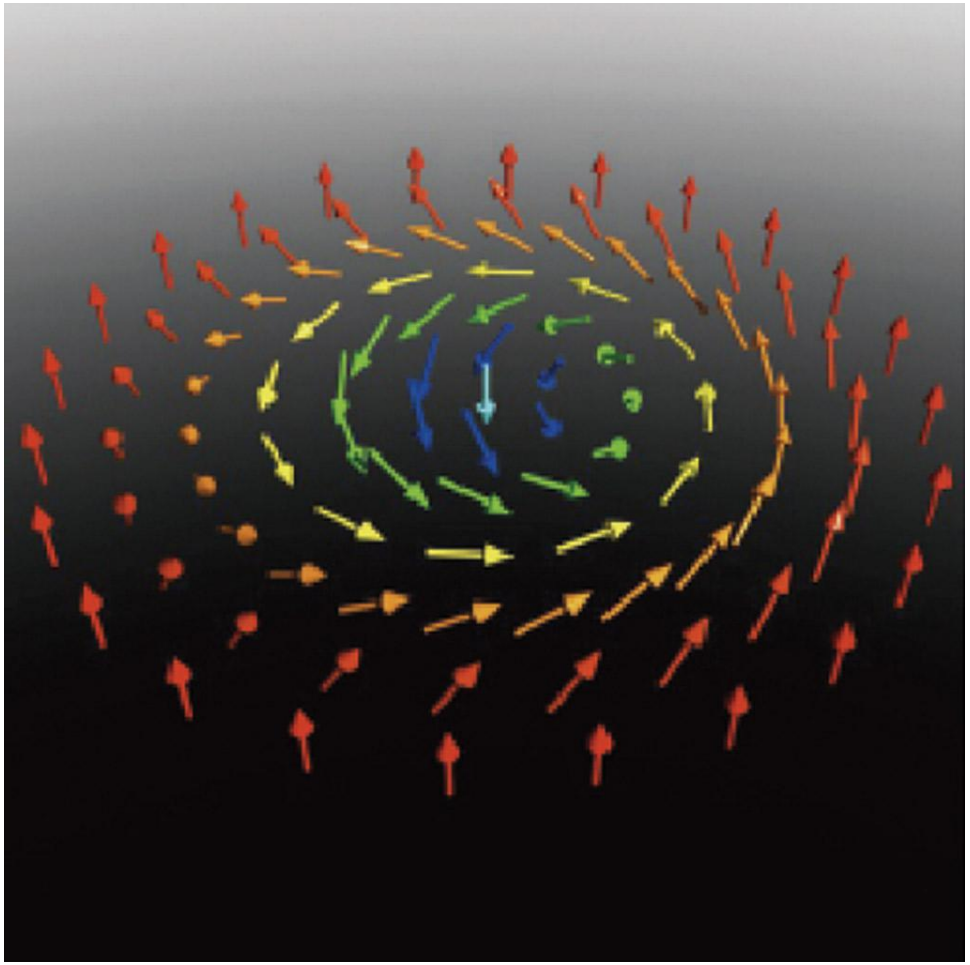
Homework

- Write a MC code for a two-dimensional Heisenberg spin lattice with perioidc boundary conditions.

$$H = J\sum_{<ij>} S_i \cdot S_j + D_{ij} \cdot \sum_{<ij>} S_i \times S_j + h^z \sum_i S_i^z$$

The vector D_{ij} points from i to j.

Skyrmion



A magnetic votex with a topological monopole