

Ising Model

Danny Goldstein

April 15, 2022

1 Introduction

The 2D Ising model is one of the simplest statistical models to show a phase transition¹. The purpose of this simulation is to investigate how some properties of this model change across the phase transition. The Ising model is a simple model for studying magnetism in materials. We are writing this simulation to study the how the magnetism, magnetic susceptibility, heat capacity and hysteresis change across the phase transition.

2 Mathematical Model

This model consists of a lattice of spins. Spins represent magnetic dipole moments, and the lattice of these spins represents a magnetic material. See Figure 1 for a cartoon.

We define a spin on lattice site i as $\sigma_i \in \{+1, -1\}$. The Hamiltonian for this system is then

$$H = - \sum_{\langle i j \rangle} J_{i,j} \sigma_i \sigma_j - \sum_i \sigma_i h_i$$

Where $J_{i,j}$ is the interaction strength of lattice sites i and j and h_i is an external magnetic field at site i . The first term gives a preference for neighboring spins to align with each other, the second terms gives spins a preference for aligning with an external field.

We then make the simplifying assumptions (that we may want to lift later) that there is uniform interaction strength and external magnetic field so $J_{i,j} = J$ and $h_i = h$. For this simulation $J = 1$

The sum over $\langle i j \rangle$ denotes a sum over all nearest neighbor pairs on a square lattice.

2.1 What Might Change

There are a few assumptions from above that we might want to relax or change:

¹Gallavotti, G. (1999), Statistical mechanics, Texts and Monographs in Physics, Berlin: Springer-Verlag, doi:10.1007/978-3-662-03952-6, ISBN 978-3-540-64883-3, MR 1707309

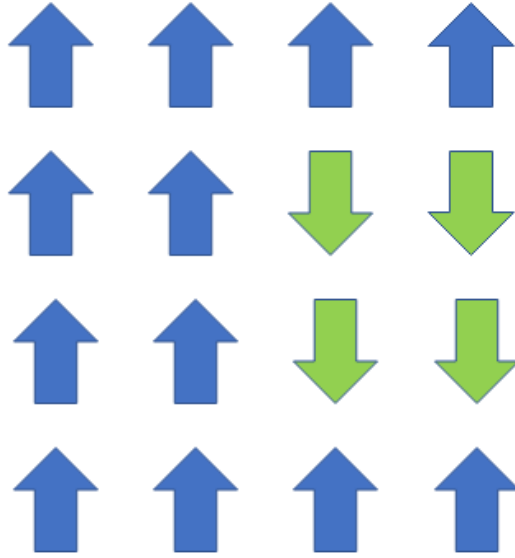


Figure 1: Cartoon of a 4x4 lattice of spins

- We might want a spatially inhomogeneous magnetic field (for a random field Ising model)
- We might want a triangular lattice instead of a square one.

3 Computational Model

To sample states of this Hamiltonian at a temperature T we use the Metropolis-Hastings algorithm. The basic process is we pick a random spin, and calculate the energy change ΔH_i by flipping the spin (from $-1 \rightarrow +1$ or $+1 \rightarrow -1$). If the energy change is negative (that is flipping the spin lowers the energy) we go ahead and do it. If the energy change is positive, we flip the spin with a probability $\exp(-\frac{\Delta H_i}{k_b T})$. We repeat this process of picking a random spin a number of times equal to the number of spins in the system. We call this the “Monte-Carlo Step” of our system.

A choice that must be made here is take measurements. The two options that are most obvious here are doing a set number of steps or waiting until some macroscopic observable has reached a steady state. The first method gives a fairly constant run time with the need to tune tolerances, but can lead to taking measurements while the system is not yet in equilibrium. Instead we will go with the second method and watch the slope of the energy to determine if we are in equilibrium.

Note that we can shortcut having to calculate the energy twice by noting that flipping

the spin negates both terms in the Hamiltonian so

$$\Delta H_i = -H_i = 2J \sum_{\langle j \rangle} \sigma_i \sigma_j + 2h\sigma_i$$

To do one “Monte-Carlo Step” in our simulation we do:

```

for the number of spins do
  Pick a random spin  $i$ 
  \\Calculate the energy change of flipping site  $i$ 
  Get a list of the nearest neighbors of spin  $i$  call this  $nn_i$ 
  \\Accumulate the change in energy for each nearest neighbor
  Initialize the energy to 0  $\Delta H_i = 0$ 
  for  $j$  in  $nn_i$  do
     $\Delta H_i = \Delta H_i + 2J\sigma_i\sigma_j - 2h\sigma_i$ 
  end
  if  $\Delta H_i < 0$  then
     $\sigma_i = -\sigma_i$  \\flip spin  $i$ 
  else
    Pick a random number  $r \in [0, 1]$ 
    if  $T > 0$  and  $r < \exp(\frac{\Delta H_i}{k_b T})$  then
       $\sigma_i = -\sigma_i$  \\flip spin  $i$ 
    end
  end
end

```

We want to run the above algorithm until the system reaches equilibrium before we make measurements. This will be done here by looking at averages of the derivative of energy of the system. After each Monte-Carlo Step we calculate the energy of the entire system and keep a log of the past 100 energy values. We then run a linear regression on the system. When the slope of the line is below a threshold (we’ll start with 1% of the average energy) we will claim the system is equilibrated.

Explicitly if we have $N = 100$ samples of energy H_n . we calculate the slope m to be:

$$m = \frac{N \sum_{n=1}^N n H_n - \sum_{n=1}^N H_n \sum_{n=1}^N n}{N \sum_{n=1}^N H_n^2 - (\sum_{n=1}^N H_n)^2}$$

This will be carried out on a square lattice with $M = 100 \times 100$ spins.

4 Initial Conditions and Boundary Conditions

4.1 Initial conditions

The case we are interested in is random initial conditions, such that each spin has a 50% chance of being in the +1 state and 50% chance of being in the -1 state.

It is easy to extend this to a case where we can define a probability $p \in [0, 1]$ where each spin has a probability of p to be in state $+1$ and $1 - p$ to be in state -1 . This can give us the two ground states by setting $p = 1$ or $p = 0$.

If later we want spatial variation in our initial conditions, we can define one or more domains Ω_k with each their own p_k to achieve arbitrary initial conditions.

4.2 Boundary conditions

We will use periodic boundary conditions. These are implemented where the neighbor lists of a site are defined.

5 Observables

5.1 Physical Observables

- Magnetism $M = \sum_i \sigma_i$.
- Energy $H = - \sum_{\langle i j \rangle} J_{i,j} \sigma_i \sigma_j - \sum_i \sigma_i$
- Specific heat $C = d\langle H \rangle / dT = (\langle H \rangle^2 - \langle H^2 \rangle) / K_b T$. This result is obtained using the fluctuation-dissipation theorem.
- Magnetic susceptibility $\chi = \delta M / \delta h = (\langle M \rangle^2 - \langle M^2 \rangle) / K_b T$. This result is obtained using the fluctuation-dissipation theorem.

5.2 Computational Observables

The energy and the magnetism are the same but now we will be explicit with how Specific heat and Magnetic susceptibility are calculated.

- Magnetism $M = \sum_i \sigma_i$.
- Energy $H = - \sum_{\langle i j \rangle} J_{i,j} \sigma_i \sigma_j - h \sum_i \sigma_i$
- Specific heat $C = (\langle H \rangle^2 - \langle H^2 \rangle) / K_b T$ We can calculate $\langle H \rangle^2$ and $\langle H^2 \rangle$ by waiting for the system to reach equilibrium as described above, and then collecting a large ($N = 1000$) number of energy measurements H_n and this $\langle H \rangle = \sum_{n=1}^N H_n / N$ and $\langle H^2 \rangle = \sum_{n=1}^N H_n^2 / N$
- Magnetic susceptibility $\chi = (\langle M \rangle^2 - \langle M^2 \rangle) / K_b T$. Where $\langle M \rangle = \sum_{n=1}^N M_n / N$ and $\langle M^2 \rangle = \sum_{n=1}^N M_n^2 / N$.

5.3 What might change

Its easy to imagine we might want to add more measurements to these. The next one I will implement is the Hilbert Dimension of the clusters of spins to measure how the shape of the clusters of spins change.

6 Experiments

1. **Phase Transition:** Measure $|M|$, H , C and χ as $K_b T$ goes from 5 to 0 in 100 evenly spaced steps called T_k . Starting from random initial conditions the system equilibrates to T_k , then some $N = 1000$ measurements of magnetism and energy are made. Then C and χ are calculated and plotted against $k_b T$.
2. **Hysteresis:** Measure $|M|$, H , C and χ as h goes from -1 to 1 in 100 evenly spaced steps called h_k . Repeat this for $k_b T_k = 0, 1, 2, 3, 4, 5$. Starting from random initial conditions the system equilibrates to T_k at $h = h_0$, then some $N = 100$ measurements of magnetism and energy are made. Then the external magnetic field changes to h_1 . We equilibrate the system and make another N measurement. This process repeats for each value of h_k until $h_k = 1$ then we reverse the path decreasing the magnetic field until $h_k = -1$.

7 Simulation Interface

This simulation will be written in C++ and Morpho with different interfaces for each. Data will be saved as comma separated values.

7.1 C++ code

The C++ code will be called by

```
./isingmodel experiment-name
```

Where experiment-name will be either "PhaseTransition" or "Hysteresis"

7.2 Morpho Code

The Morpho code will have a short script for each experiment that will be called like:

```
morpho5 IsingPhaseTransition.morpho
```

```
morpho5 IsingHysteresis.morpho
```

8 Code Structure

...

9 Test Plan

...