

FYS3150 - Project 4

Gunnar Lange

November 15, 2016

Abstract

HEI

Contents

1	Introduction	1
2	Theoretical model	2
2.1	Thermodynamic quantities and the Ising model	2
2.2	Periodic boundary condition	3
2.3	Phase transitions in the Ising model	3
2.4	The Metropolis Algorithm	3
2.5	Equilibrium of the system	4
2.6	Analytic solution for the 2x2 case with periodic boundary conditions	4
3	Methods	7
3.1	Energy in our system	7
3.2	Implementing periodic boundary conditions	7
3.3	Implementing the Metropolis algorithm	7
3.4	Investigating the time taken to arrive at the most likely state and the underlying probability distribution	8
3.5	Investigating phase transitions	8
3.6	Parallelizing our code	8
4	Results	8
5	Discussion	8
6	Conclusion	8

1 Introduction

The Ising model is a simple but very popular model for modelling phase transitions (see e.g. **HERE**). The model consists of a lattice of atomic spins which can have one of two possible values ("up" or "down"). The energy of each spin in their configuration is determined solely by their nearest neighbors. This model is widely used in statistical physics, in particular to study ferromagnetic phenomena.

We will investigate how multiple interesting thermodynamic quantities, including heat capacity, magnetic susceptibility, mean magnetization and mean energy, behave over time in the Ising model. We will investigate both ordered and disordered initial states of the lattice, checking the equilibration time in each case. Finally, we will also investigate phase transitions in the Ising model. We implement periodic boundary conditions and simulate temporal progression in the lattice by means of the Metropolis algorithm.

We begin with a discussion of the thermodynamic quantities that we will study, and how they are manifested in the Ising model. We then briefly discuss our chosen boundary conditions. Subsequently, we present the theory of phase transitions and the Metropolis algorithm, and discuss how we could identify the equilibrium of our system. We then discuss some technicalities relating to our implementation of the physical model, before presenting and discussing our results..

2 Theoretical model

2.1 Thermodynamic quantities and the Ising model

We study a square lattice consisting of $N \times N$ magnetic dipole moments (atomic spins) that can be in one of two states (+1 or -1). We wish to study how energy, magnetization, heat capacity and magnetic susceptibility develop in this system. This system, the Ising model, is a microcanonical system, i.e. we keep the temperature fixed. Then it is known from basic thermodynamics that this system can be described by Boltzmann statistics, i.e. by a distribution function given by:

$$P(E) = \frac{1}{Z} e^{-\frac{E}{k_B T}} \quad (1)$$

Where E is the energy of a microstate, k_B is Boltzmann's constant and T is the temperature of the system. We define $\beta = 1/(k_B T)$. Then Z , the partition function, is a normalization constant defined by:

$$Z = \sum_i e^{-\beta E_i} \quad (2)$$

Where the sum is over all microstates, i . As $P(E)$ is a probability function, it may be used to define moments of a quantity X according to the general scheme:

$$\langle X^n \rangle = \frac{1}{Z} \sum_i X_i^n e^{-\beta E_i} \quad (3)$$

Where X_i is a quantity associated with the microstate i . This will be useful later, when computing the heat capacity and the magnetic susceptibility.

From this probability distribution, and the partition function, we can compute all thermodynamic quantities of interest. In the simplest model of the Ising system, the energy is given by:

$$E = -J \sum_{\langle kl \rangle} s_k s_l \quad (4)$$

Where the sum is over all the nearest neighbors in the lattice. J is a constant which is determined by the quantum mechanical details of the system. We will simply set it to 1. The magnetization, on the other hand, is given as:

$$\mathcal{M} = \sum_i s_i \quad (5)$$

Where the sum goes over all spins, i .

It is shown **here** that this gives the following relation for thermodynamical quantities:

Heat capacity, C_V

$$C_V = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2} \quad (6)$$

Magnetic susceptibility, χ :

$$\chi = \frac{\langle \mathcal{M}^2 \rangle - \langle \mathcal{M} \rangle^2}{k_B T} \quad (7)$$

The first moment of the energy, $\langle E \rangle$, can also be computed in another way, which will be useful later, namely:

$$\langle E \rangle = -\frac{\partial \ln Z}{\partial \beta} \quad (8)$$

This is proved **here**.

2.2 Periodic boundary condition

From the above description, it is not obvious how to treat the boundaries of the lattice. We will assume the lattice (representing for example a crystalline structure in a solid) to be essentially infinite in all directions. In this case, we can assume the boundary to have no effect on the behavior of our crystal. Therefore, we let our crystal "loop around" itself, that is, we let neighbor of the leftmost crystals be the rightmost crystals, simulating a "continuation" of the crystals to the left. We adapt the same approach for the other edges of the crystals.

2.3 Phase transitions in the Ising model

It is well known from literature **REFERENCE** that there exists a critical temperature, T_C , at which phase transitions can be observed in the Ising model. At this temperature, one can observe spikes in the thermodynamic quantities. Near T_C , our thermodynamic quantities can be modelled as simple power laws. As shown **HERE**, these laws take the following forms:

$$\begin{aligned}\langle \mathcal{M}(T) \rangle &\sim (T - T_C)^\beta \\ C_V(T) &\sim |T_c - T|^{-\gamma} \\ \mathcal{X}(T) &\sim |T_C - T|^{-\alpha}\end{aligned}\tag{9}$$

Where α, β and γ are critical exponents, given by: $\alpha = 0, \beta = 1/8, \gamma = 7/4$

The temperature T_C will depend upon the number of spins in our lattice, N . Ideally, the number of spins should be close to infinite. However, our computational capacities limit us to systems with a maximum size of about $N = 140$. Luckily, however, we can estimate the critical temperature for $N = \infty$, $T_C(N = \infty)$ from the critical temperature at a finite N , $T_C(N)$ by the equation shown **HERE**, which we reproduce:

$$T_C(N) - T_C(N = \infty) = aN^{-1/\nu}\tag{10}$$

Where a is a constant, which can be determined from:

$$a = \frac{T_C(N_1) - T_C(N_2)}{N_1^{-1/\nu} - N_2^{-1/\nu}}\tag{11}$$

ν is another critical temperature, which has the exact result $\nu = 1$. Thus we can estimate the critical temperature by computing a from equation 11 for two different lattice sizes, N_1 and N_2 , and then solving equation 10 for $T_C(N = \infty)$. Note that we can also use the equations in 9 near the critical temperature as analytic solutions to which we can compare our numerical results.

2.4 The Metropolis Algorithm

We will use Monte-Carlo simulations to model the time development of our spin system. Assume that we have an initial random configuration of spins, with energy E_b (which we have calculated from equation 4). We employ the famous Metropolis algorithm to achieve this. The Metropolis algorithm is described in detail **HERE**, but it can be briefly summarized in the following steps:

- Randomly select N^2 spins from the $N \times N$ lattice.
- For each spin, calculate the change in energy, ΔE , that the system would experience if we were to flip it
- Draw a random number, ζ uniformly between 0 and 1, and compare it to $e^{-\beta\Delta E}$.
- If $\zeta \leq e^{-\beta\Delta E}$, flip the spin, else do not flip the spin (note that any spin flip with $\Delta E < 0$ will always be performed).

- Update the thermodynamic quantities.

It can be shown (**REFERENCE**) that this simple scheme will evolve the system towards its equilibrium state (dictated by the Helmholtz free energy). Thus, repeating the above steps many times, which corresponds to performing many Monte-Carlo cycles, will evolve our system towards equilibrium. We will therefore use the number of Monte-Carlo cycles synonymously with time in our simulation.

2.5 Equilibrium of the system

We choose multiple different starting configurations of our lattice, including a homogeneous configuration (all spins initially point the same way), and a random configuration (spins drawn from a uniform distribution). We expect that our system should approach a stationary state after some time, T . There are multiple indicators of this state. First of all, we expect the number of accepted spins to reach a low, constant value, i.e. the total number of accepted spins should be a curve with a steadily decreasing slope. Most spins will be in their stable configuration, and therefore few flips will be favorable, energy-wise. We also expect all thermodynamical quantities to reach an equilibrium state, where the variations are small. Both of these will be good indicators for equilibrium.

2.6 Analytic solution for the 2×2 case with periodic boundary conditions

It is not too difficult to find an analytic solution for a small lattice consisting only of 2×2 spins. This analytic solution provides an important consistency check for our algorithm. Assume therefore that we have a lattice consisting of 4 magnetic dipole moments, labelled as shown in figure 1:

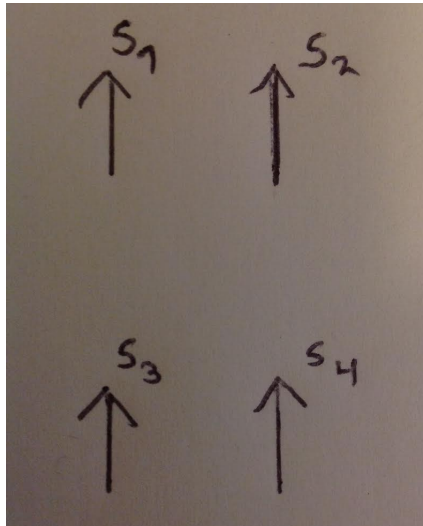


Figure 1: One microstate of our 2×2 lattice, illustrating how we label the spins.

We notice first there are only 5 distinct macrostates, namely 0 to 4 spins up. There should be a total of $2^4 = 16$ microstates. It is clear that the extremes (0 spins up and 4 spins up), have only one microstate associated with them. The states 1 spin up and 3 spins up each have 4 associated microstates (you can flip one of the four spins). This implies that there are a total of 6 ways to have the state 2 up and 2 down. This is summarized in the table below:

Spins up	Number of microstates
4	1
3	4
2	6
1	4
0	1

Now we must investigate the energy of each of these states. In general, the energy with periodic boundary condition for a 2×2 lattice can be written as:

$$E = -J[s_1s_2 + s_2s_1 + s_1s_3 + s_3s_1 + s_2s_4 + s_4s_2 + s_3s_4 + s_4s_3]$$

From this it follows immediately that the energy for the states for 4 spins up or 4 spins down is $-8J$. If we flip one spin, we will flip the sign of exactly half the terms (because each spin occurs in four of the eight pairs). Thus the energy will be zero. If we flip two spins, we can either flip spins that are nearest neighbors (such as s_1 and s_2 , s_1 and s_3 , s_2 and s_4 or s_3 and s_4), or we can flip spins that are not nearest neighbors (such as s_1 and s_4 or s_2 and s_3). In the first case, we flip the sign of four of the terms in the sum (namely all those that contain one of the two spins we flip). Thus the total energy will be zero. In the other case, we flip the sign of all terms. Thus the total energy will be $8J$.

To find the magnetization, we must simply sum the total number of spin up and subtract the number of spin down. This is all summarized in the table below:

Spins up	Energy [J]	Magnetization[J/T]	Number of microstates
4	-8J	4	1
3	0	2	4
2	0	0	4
2	8J	0	2
1	0	-2	4
0	-8J	-4	1

This enables us to calculate the partition function as:

$$Z = \sum_{i=1}^{16} e^{-\beta E_i}$$

Where $\beta = 1/k_B T$. As most of the energies are zero, this is relatively straightforward and gives:

$$Z = 2e^{8\beta J} + 2e^{-8\beta J} + 12 = 4 \cosh(8\beta J) + 12$$

We can now compute the expected value of the energy as:

$$\langle E \rangle = -\frac{\partial \ln Z}{\partial \beta} = -\frac{\partial}{\partial \beta} \ln (2e^{8\beta J} + 2e^{-8\beta J} + 12) = \frac{-16Je^{8\beta J} + 16Je^{-8\beta J}}{2e^{8\beta J} + 2e^{-8\beta J} + 12} = -\frac{32J \sinh(8\beta J)}{4 \cosh(8\beta J) + 12} \quad (12)$$

Similarly, we can compute the heat capacity, C_V , as:

$$C_V = \frac{d\langle E \rangle}{dT} = \frac{d}{dT} \left(\frac{-16Je^{\frac{8J}{k_B T}} + 16Je^{-\frac{8J}{k_B T}}}{2e^{\frac{8J}{k_B T}} + 2e^{-\frac{8J}{k_B T}} + 12} \right)$$

However, this is an ugly differentiation. Therefore, we will instead use the relation:

$$C_V = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2}$$

The square of the expected value of the energy is given by:

$$\langle E^2 \rangle = \frac{1}{Z} \sum_{i=1}^{16} E_i^2 e^{-\beta E_i}$$

The sum is easy to compute:

$$\sum_{i=1}^{16} E_i^2 e^{-\beta E_i} = 64J^2 e^{8J\beta} + 128J^2 e^{-8J\beta} + 64J^2 e^{8J\beta} = 128J^2 (e^{8J\beta} + e^{-8J\beta})$$

Which gives:

$$\langle E^2 \rangle = \frac{128J^2 (e^{8J\beta} + e^{-8J\beta})}{2e^{8J\beta} + 2e^{-8J\beta} + 12} = \frac{256J^2 \cosh(8J\beta)}{4 \cosh(8J\beta) + 12}$$

From which it follows that:

$$C_V = \frac{1}{k_B T^2} \left(\frac{128J^2 (e^{8J\beta} + e^{-8J\beta})}{2e^{8J\beta} + 2e^{-8J\beta} + 12} - \left(\frac{-16J e^{8\beta J} + 16J e^{-8\beta J}}{2e^{8\beta J} + 2e^{-8\beta J} + 12} \right)^2 \right)$$

$$C_V = \frac{1}{k_B T} \left(\frac{256J^2 \cosh(8J\beta)}{\cosh(8J\beta) + 12} - \left(\frac{-32J \sinh(8J\beta)}{4 \cosh(8J\beta) + 12} \right)^2 \right) \quad (13)$$

Magnetization can be computed as:

$$\langle M \rangle = \frac{1}{Z} \sum_{i=1}^{16} M_i e^{-\beta E_i}$$

The sum can be computed as:

$$\sum_{i=1}^{16} M_i e^{-\beta E_i} = 4e^{8J\beta} + 8 - 8 - 4e^{8J\beta} = 0$$

Thus the expected value of the magnetization is zero. The expected value of the absolute value of the magnetization on the other hand is:

$$\langle |M| \rangle = \frac{1}{Z} \sum_{i=1}^{16} |M_i| e^{-\beta E_i} = 8e^{8J\beta} + 16 \quad (14)$$

The square of expected value of the magnetization can be computed as:

$$\langle M^2 \rangle = \frac{1}{Z} \sum_{i=1}^{16} M_i^2 e^{-\beta E_i} = \frac{1}{Z} (16e^{8\beta J} + 8 + 8 + 16e^{8\beta J}) = \frac{8e^{8\beta J} + 4}{\cosh(8J\beta) + 3}$$

Now it is easy to compute the susceptibility:

$$\chi = \frac{\langle M^2 \rangle - \langle M \rangle^2}{k_B T} = \frac{1}{k_B T} \left(\frac{8e^{8\beta J} + 4}{\cosh(8\beta J) + 3} \right) \quad (15)$$

We can now simulate a 2×2 lattice, and check if equations 12, 13, 14 and 15 hold true, which gives us a consistency check for our methods.

3 Methods

3.1 Energy in our system

Note that the sum in equation 4 is only over the nearest neighbors. Therefore, the change in energy from the flip of a single spin (as in the Metropolis algorithm), depends only on the configuration of the nearest neighbors (which are only four spins). This means that there are only a limited number of possible energy differences, ΔE . As shown **HERE**, the only possible values of ΔE are:

$$\Delta E = \{-8J, -4J, 0, 4J, 8J\}$$

These can all be precomputed. Thus, for each spin flip in the Metropolis algorithm, we can simply check the configuration of the neighboring spins, and then assign an energy change, ΔE , by checking the state of the neighboring spins, as described in more detail in section 3.3.

Not that this works well for all but the initial configuration. For the initial configuration, we will have to compute the total energy. This can be done by a nested for-loop construction, where we sum over each row and column of the lattice. We start in the top-left corner, and always sum

FINISH THIS

3.2 Implementing periodic boundary conditions

The periodic boundary conditions can be implemented by using modular division. Each spin will have four nearest neighbors in the lattice. Assume our selected spin is at x_i, y_i . This spin will be affected by the spin at position x_{i+1}, y_i and x_{i-1}, y_i (as well as the corresponding spins in the y-direction). If x_i is on the boundary of our lattice, however, the point to the left or to the right may not exist. Therefore, we find adjacent points by using the following formula:

$$x_{i+1} = (x_{i+1} + N) \bmod N \quad (16)$$

Note that, if x_{i+1} happens to be element number N , this equation gives element 0, and if $x_i - 1$ happens to be element 0, the equation gives element number $N - 1$. Thus, this implements the periodic boundary conditions.

3.3 Implementing the Metropolis algorithm

The Metropolis algorithm is also implemented as two nested for-loops: the outer one, where we sum over Monte Carlo steps, and the inner one where we sum over $N \times N$ randomly selected spins.

As we mentioned in section 3.1, we can precompute all possible energy differences. Thus, when we propose a spin flip, we must simply compute the energy change due to the flip, which can be found from:

$$\Delta E = 2J \sum_k s_i s_k$$

Where the sum extends over all neighbors of the spin s_i . We can then associate this ΔE with the correct, precomputed, Boltzmann factor, and do the comparison described in section 2.4. This is shown in Pseudo-code below. Note that we here disregard the periodic boundary conditions, for improved readability.

```
for i in range (MCSteps):
    for k in range (N*N):
        x=int(random.uniform(0,1)*N)
        y=int(random.uniform(0,1)*N)
        dEnergy=2*grid(x,y)*(grid(x+1,y)+grid(x-1,y)+grid(x,y+1)+grid(x,y-1))
        boltzmann=boltzmann_factor(dEnergy) //Precomputed factors
        if random.uniform(0,1)<=boltzmann: //Metropolis test
```

```

lattice(x,y)*=-1
energy +=dEnergy
magnetic_moment=2*lattice(x,y)

```

3.4 Investigating the time taken to arrive at the most likely state and the underlying probability distribution

We wish to investigate how the equilibration time (time to reach most likely state) of our system depends on the temperature of our system, and on the initial configuration. We adapt a plain graphical approach - we simply plot the time development of the mean energy and the mean absolute value of the magnetization, to find when the (initially large) deviations die out. An alternative way to characterize equilibrium is, as discussed in section 2.5, with the total number of accepted configurations. Therefore, we include a plot of this as well.

Additionally, we would also like to characterize the probability distribution governing the equilibrium state. Therefore, we start our simulation after the time which we determined to be the equilibration time, and compute the number of times a given energy E appears.

3.5 Investigating phase transitions

To investigate the phase transitions that occur in the Ising system, we employ the fact that the analytic result for $T_C(N = \infty)$ from section 2.3 is approximately 2.269 **REFERENCE**. Thus we choose temperatures close to this, and find the critical temperature (the temperature at which a divergent behavior is observed for the heat capacity and the magnetic susceptibility). Then we compare with the equations from section 2.3.

3.6 Parallelizing our code

Notice how the algorithm presented schematically in section 3.3 has no dependence on i or k - they are simply counting variables. This algorithm is therefore classified as embarrassingly simple to parallelize.

4 Results

5 Discussion

6 Conclusion