FYS4150
COMPUTATIONAL PHYSICS

PROJECT 4
THE ISING MODEL

KRISTINE GARVIN

NOVEMBER 19, 2018

ABSTRACT

The aim of this project is to study the Ising model in two dimensions. The Ising model is a popular model for simulating phase transitions. The model is implemented by the Metropolis algorithm and we make ensembles by using Monte Carlo simulations. These ensembles can be used to find expectation values, and with large ensembles, 1 million Monte Carlo cycles, we can make good approximations to the analytic results that is possible to calculate for a simple 2×2 square lattice. For a 20×20 lattice we find that one needs approximately $2 \cdot 10^4$ Monte Carlo cycles for a low temperature and $4 \cdot 10^4$ Monte Carlo cycles for a high temperature to reach the equilibrium state situation. We also find an approximation to the Curie temperature, the temperature at which the phase transition from a ferromagnetic to a paramagnetic material occurs. By using the results we get from studying the thermodynamic quantities for a 80×80 and a 100×100 lattice as functions of temperature, we find the Curie temperature to be approximately $T_C = 2.249[kT/J]$.

Contents

1 Introduction

Ferromagnetic materials exhibit the 'long-range ordering phenomenon' at the atomic level. This means that unpaired electron spins line up parallel with each other in domains locking the magnetic moments of neighbouring atoms into a rigid parallel order over a large number of atoms in spite of the thermal agitation which tend to randomize any atomic-level order. This order abruptly disappears at a certain temperature which is called the Curie temperature for the material.

The Ising model, after Ernst Ising, is a mathematical model of magnetism in statistical mechanics, a simple way of describing how a magnetic material reacts to temperature. The model consists of discrete variables that represent the magnetic dipole moments of atomic spins. The spins can be either positive or negative, represented by the values ± 1. The model is a simplified way to study reality and widely used as a tool to identify phase transitions.
For this study the spins are arranged in a two dimensional square lattice with each spin interacting with the four nearest neighbours. At the boundaries, periodic boundary conditions are applied.
By using the theory of the canonical statistical ensemble we are able to derive the systems thermodynamic quantities from this collection of micro-physical systems. We study the magnetic system at constant temperature with no external magnetic field acting on the system.
For a small 2x2 lattice, the expected mean energy, mean momentum, specific heat and susceptibility can be found analytically, but for larger lattices this calculations must be done numerically.
The model will be implemented numerically by the Metropolis algorithm and we will run Monte Carlo simulations to repeat the experiment and create ensembles. To make good approximations of the thermodynamic quantities, we want to find the number of Monte Carlo cycles that is necessary to reach steady state, this depends on the temperature of the system, we will then discard these first cycles. Finally we will look at the thermodynamic quantities as function of temperature, and by studying the phase transition we can make an approximation to the Curie temperature as the critical temperature for a lattice of infinite size.

2 Method

The energy of a micro-state in the Ising model is expressed as follows

$$E = -J \sum_{<kl>}^{N} s_k s_l - \mathcal{B} \sum_{k}^{N} s_k \tag{1}$$

Where J is a coupling constant expressing the strength of the interaction between the neighbouring spins, $< kl >$ indicates that we sum over nearest neighbours only and $\mathcal{B}$ is an external magnetic field.

For this study we assume that we have a ferromagnetic ordering ($J > 0$) and no external magnetic field ($\mathcal{B} = 0$) interacting with the magnetic moments set up by the spins.

2.1 Canonical ensemble

With a canonical ensemble of micro-states it is possible to derive expectation values and thermodynamic properties of the total system, given a probability distribution. In this case we use the Boltzmann distribution,

$$P_i(\beta) = \frac{e^{-\beta E_i}}{Z} \tag{2}$$

where $\beta = 1/k_B T$ being the inverse temperature and k_B the Boltzmann constant.

The energy of a micro state is defined as E_i and thus the partition function for our ensemble is

$$Z = \sum_{i=1}^{M} e^{\beta E_i}, \tag{3}$$

where the sum extends over all microstates M.

As there is no external magnetic field acting on our system, the energy of a micro state or specific configuration is defined by the Ising model as

$$E_i = -J \sum_{<kl>}^{N} s_k s_l \tag{4}$$

At the boundaries we will apply periodic boundary conditions. This means that the *'neighbour'* to the left of the leftmost boundary spin will take the rightmost boundary spin as its value, the same applies at the right side boundary and the top and bottom of the lattice.

As we allow the system to exchange energy with its surroundings the energy is an expectation value or mean value of the micro-states of our ensemble, and can be defined in terms of the probability distribution P_i as follows

$$\langle E\rangle = \sum_{i=1}^{M} E_i P_i(\beta) = \frac{1}{Z}\sum_{i=1}^{M} E_i e^{-\beta E_i} \tag{5}$$

From this we can derive the specific heat of the system as a function of the energy's variance

$$C_v = \frac{\sigma_E^2}{k_B T^2} = \frac{1}{k_B T^2}\left(\langle E^2\rangle - \langle E\rangle^2\right) \tag{6}$$

The magnetization of a specific configuration of the system is defined as the sum over all spins in the lattice

$$\mathcal{M}_i = \sum_{j=1}^{N} s_j \tag{7}$$

We can also calculate the mean magnetization over all possible configurations as a function of the probability distribution

$$\langle \mathcal{M}\rangle = \sum_{i}^{M} \mathcal{M}_i P_i(\beta) = \frac{1}{Z}\sum_{i}^{M} \mathcal{M}_i e^{-\beta E_i} \tag{8}$$

From this we can derive the susceptibility of the system as a function of the magnetization's variability

$$\chi = \frac{\sigma_{\mathcal{M}}^2}{k_B T} = \frac{1}{k_B T}\left(\langle \mathcal{M}^2\rangle - \langle \mathcal{M}\rangle^2\right) \tag{9}$$

Remember that $\beta = \frac{1}{k_B T}$ and that all the above quantities are functions of the temperature.

2.2 A simple 2x2 lattice

Analytic solution

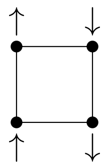

Figure 1: Sample 2 × 2 spin lattice.

For a simple $2x2$ lattice, each spin sees four neighbouring spins. As each spin can have two states, the number of configurations are $2^4 = 16$. The energy in each configuration i is given by the sum of the product of all neighbouring spin values.

$$E_i = -J \sum_{j=1}^{N} s_j s_{j+1} = s_1 s_2 + s_1 s_3 + s_2 s_1 + s_2 s_4 + s_4 s_2 + s_4 s_3 + s_3 s_4 + s_3 s_1.$$

The only time the energy will be non-zero, is when all spins are in the same direction, or if the spins on the diagonals share direction.

↑ ↑	↓ ↓	↑ ↓	↓ ↑
↑ ↑	↓ ↓	↓ ↑	↑ ↓
E=-8J	E=-8J	E=8J	E=8J

This means that twelve configurations results in zero energy, for the remaining four, $E = \pm 8J$.

The magnetization of each configuration is given by

$$\mathcal{M}_i = \sum_{j=1}^{N} s_j = s_1 + s_2 + s_3 + s_4$$

This results in a magnetization of ±4 for all spins equal, ±2 for all except one spin equal and zero otherwise. The results together with number of configurations of each case is collected in the table below.

Configs	Spins up	Energy	Magnetization
1	4	-8J	4
4	3	0	2
4	2	0	0
2	2	8J	0
4	1	0	-2
1	0	-8J	-4

Table 1: Energy and magnetization for Ising model with 2x2 lattice with periodic boundary conditions

The partition function can be found as follows

$$Z = \sum_{i=1}^{M} e^{-\beta E_i} = 12e^{-\beta 0J} + 2e^{8\beta J} + 2e^{-8\beta J} = 12 + 2(e^{8\beta J} + e^{-8\beta J})$$

Rewritten in terms of $e^x + e^{-x} = 2cosh(x)$, the partition function becomes

$$Z = 12 + 4cosh\left(8\beta J\right)$$

With the partition function we can now calculate the Energy's expectation value $\langle E \rangle$, given by

$$\begin{aligned}\langle E \rangle &= \sum_{i=1}^{M} E_i P_i(\beta) = \frac{1}{Z}\sum_{i=1}^{M} E_i e^{-\beta E_i} \\ &= \frac{1}{Z}(2 \cdot 8Je^{8\beta J} + 2 \cdot 8Je^{-8\beta J}) \\ &= \frac{16J}{Z}(-e^{8\beta J} + e^{-8\beta J})\end{aligned}$$

Rewriting in terms of $2sinh(x) = e^x - e^{-x}$ together with our expression for Z, the mean energy becomes

$$\langle E \rangle = -\frac{8Jsinh\left(8\beta J\right)}{3 + cosh\left(8\beta J\right)}$$

The specific heat can be calculated from the variance of the expected energy as follows

$$C_v = \frac{\sigma_E^2}{k_B T^2} = \frac{1}{k_B T^2}\left(\langle E^2\rangle - \langle E\rangle^2\right)$$

We calculate the variance of the expectation value for the energy in two steps,

$$\begin{aligned}\langle E^2\rangle &= \sum_{i=1}^{M} E_i^2 P_i(\beta) = \frac{1}{Z}\sum_{i=1}^{M} E_i^2 e^{-\beta E_i} \\ &= \frac{1}{Z}\left(2\cdot(8J)^2 e^{8\beta J} + 2\cdot(8J)^2 e^{-8\beta J}\right) \\ &= \frac{64J^2 cosh\left(8\beta J\right)}{\left(3 + cosh(8\beta J)\right)}\end{aligned}$$

$$\langle E\rangle^2 = \frac{64J^2 sinh^2\left(8\beta J\right)}{\left(3 + 2cosh(8\beta J)\right)^2},$$

After some rearrangement of the expression, the specific heat can be expressed as

$$C_v = \frac{64J^2}{k_B T^2\left(3 + cosh(8\beta J)\right)}\left[cosh\left(8\beta J\right) - \frac{sinh^2\left(\beta J\right)}{3 + cosh\left(8\beta J\right)}\right]$$

The expected magnetization can also be calculated as a function of the probability distribution,

$$\begin{aligned}\langle \mathcal{M}\rangle &= \sum_{i}^{M} \mathcal{M}_i P_i(\beta) = \frac{1}{Z}\sum_{i}^{M} \mathcal{M}_i e^{-\beta E_i} \\ &= \frac{1}{Z}\left[4e^{8\beta J} + (-4)e^{8\beta J} + 4\cdot(-2)e^0 + 4\cdot 2e^0\right] = 0.\end{aligned}$$

Analytically, this is not a very interesting result and will always be the result when you sum up the magnetization for all possible configurations.

The absolute mean magnetization on the other hand, is given by

$$\begin{aligned}\langle|\mathcal{M}|\rangle &= \sum_{i}^{M}|\mathcal{M}_i|P_i(\beta) = \frac{1}{Z}\sum_{i}^{M}|\mathcal{M}_i|e^{-\beta E_i}\\ &= \frac{1}{Z}\left[2\cdot 4e^{8\beta J} + 8\cdot 2e^{0}\right] = 16 + 8e^{-8\beta J}\end{aligned}$$

The susceptibility of the system is a function of the variability of the magnetization

$$\chi = \frac{\sigma_{\mathcal{M}}^2}{k_B T} = \frac{1}{k_B T}\left(\langle\mathcal{M}^2\rangle - \langle\mathcal{M}\rangle^2\right)$$

Again we calculate the variance in two steps,

$$\begin{aligned}\langle\mathcal{M}^2\rangle &= \sum_{i=1}^{M}\mathcal{M}_i^2 P_i(\beta) = \frac{1}{Z}\sum_{i=1}^{M}\mathcal{M}_i^2 e^{-\beta E_i}\\ &= \frac{1}{Z}\left[4^2 e^{8\beta J} + (-4)^2 e^{8\beta J} + 4\cdot 2^2 e^0 + 4\cdot(-2)^2 e^0\right]\\ &= \frac{1}{Z}\left[32e^{8\beta J} + 32\right]\end{aligned}$$

$$\langle\mathcal{M}\rangle^2 = 0,$$

The susceptibility thus become,

$$\chi = \frac{8}{k_B T}\left[\frac{e^{8\beta J}+1}{3 + cosh(8\beta J)}\right]$$

These expressions will be used to compare our numerical approach to a 2 × 2 lattice with analytic solutions.

2.3 Studies of phase transition

As mentioned above we are looking for the phase transition of our system. A phase transition is marked by abrupt macroscopic changes due to increase of temperature. Near the critical temperature, the Curie temperature, we can characterize the behaviour of many physical quantities by power laws. For the Ising model, the mean magnetization is given by

$$\langle \mathcal{M}(T) \rangle \sim (T - T_C)^{\beta} \tag{10}$$

here $\beta = 1/8$ is called the critical exponent.
Similarily, the heat capacity and susceptibility is given by

$$C_v(T) \sim |T_C - T|^{\alpha} \tag{11}$$

$$\chi(T) \sim |T_C - T|^{\gamma} \tag{12}$$

where $\alpha = 0$ and $\gamma = 7/4$.
As T approaches the critical temperature T_C the spins become more and more correlated and the correlation length ξ diverges as

$$\xi(T) = |T - T_C|^{\nu} \tag{13}$$

For a second-order phase transition the correlation length spans the whole system. As we are always limited to a finite lattice, the correlation length will be proportional to the size of the lattice. It is possible to relate the behavior of a finite lattice to a infinitely large lattice through finite size scaling relations. The critical temperature then scales as

$$T_C(L) - T_C(L = \infty) \propto aL^{-1/\nu} \tag{14}$$

And the correlation length can be shown to be proportional to

$$\xi(T) \propto L \sim |T_C - T|^{-\nu} \tag{15}$$

The constant a is unknown, but can be found if we have the critical temperature of two different lattices. In this study we simulate the critical temperatures for lattices of size 80×80 and 100×100 and calculate an approximation of the critical temperature for a infinite system.

With only one function and two unknowns, we approximate the constant a by simply subtracting the output from two runs.

$$T_C(L_1) - T_C(L = \infty) \propto aL_1^{-1/\nu}$$
$$T_C(L_2) - T_C(L = \infty) \propto aL_2^{-1/\nu}$$

$$T_C(L_1) - T_C(L = \infty) - T_C(L_2) + T_C(L = \infty) \propto aL_1^{-1/\nu} - aL_2^{-1/\nu}$$
$$\frac{T_C(L_1) - T_C(L_2)}{(L_1^{-1/\nu} - L_2^{-1/\nu})} \propto a$$

This lead to the approximation of the critical temperature as

$$T_C(L = \infty) \propto T_C(L_2) - \left(\frac{T_C(L_1) - T_C(L_2)}{(L_1^{-1/\nu} - L_2^{-1/\nu})}\right) * L_2^{-1/\nu}. \tag{16}$$

2.4 Algorithm

The algorithm we have used to solve the Ising model, is the Metropolis algorithm. The path to achieving a steady state for the Ising model can be seen as a Markov chain and the use of the Metropolis algorithm can be seen as one cycle in a Monte Carlo solver, with the probability of finding the system in a given state as sampling function.

In this study we need random samples from our probability distribution

$$P_i(\beta) = \frac{e^{-\beta E_i}}{Z} \tag{17}$$

The problem is that we need to compute the partition function

$$Z = \sum_{i=1}^{M} e^{\beta E_i}, \tag{18}$$

This is difficult to compute as we need the all states. Fortunately the Metropolis algorithm only need a function which is proportional to the probability distribution, so we do not need to compute the partition function at all. New configurations of the system is generated from the previous one using this transition probability that depends on the energy difference between the initial and final states. In this study, Markov chains are used repeatedly in Monte Carlo simulations to simulate how the system evolves towards steady state.

A Markov chain is a good tool to describe a system moving towards a steady state, with a given initial configuration. A Markov process is a random walk with a set of probabilities for making a set of moves.

The Metropolis algorithm can be described by the following steps

The system is given an initial configuration, with energy E_b at a random position in the lattice. We will later compare the difference between random initial configuration and ground state initial configuration.

1. The configuration is changed by flipping the spin. The energy of this trial state, E_t is calculated.
2. Then the energy difference between the new and previous energy states, $\Delta E = E_t - E_b$ is calculated. In two dimensions ΔE is limited to five values as you could see from the table above.
3. If $\Delta E < 0$, the new configuration is accepted. This means that the energy is lowered and the system is progressing towards the energy minimum for the given temperature.
4. If $\Delta E > 0$, $w = e^{-\beta \Delta E}$ is computed.
5. w is compared to a random number r. If $r \leq w$, the new configuration is accepted. If not, the initial configuration is kept.
6. The expectation values are updated.

These steps are repeated as many times as there is spins in the lattice, i.e. $L \times L$ times. For each loop the energy and magnetization is added to the previous value. These steps represents one Monte Carlo cycle, a simulated measurement of the energy and magnetization of one configuration. The energy and magnetization are stored in arrays and summed and divided by number of cycles when the calculations are done. These calculations gives us the expectation values as defined above and the necessary mean values to further study the phase transition of the system.

3 Implementation

Implementation of the Metropolis algorithm and the Monte Carlo simulation can be found in the script called *ising_model.py*.

To speed things up, Numbas "just-in-time" decorator @*jit* has been applied to the functions in the script marked for optimization by numbas jit compiler.

The function MC takes 3 variables

1. spin_matrix: The initial lattice configuration
2. num_cycles: The number of Monte Carlo cycles
3. temperature: For the calculation of $w = e^{\beta \Delta E} = e^{\Delta E / k_B T}$

The function picks out a random position in the lattice and evaluates it according to the description of the Metropolis algorithm above. This is repeated for a number equal to the number of lattice elements, and the energy, magnetization and their squared values, together with number of accepted configurations, are stored as a row in the output matrix. The whole operation is repeated for each Monte Carlo cycles and the output is a matrix where the columns contain the evolution of the thermodynamic quantities in the system. These columns can later be extracted, summed up and normalized and the the specific heat and susceptibility can be calculated from the variance.

```
for i in range(1, num_cycles+1):
        for j in range(num_spins*num_spins):
            ix = np.random.randint(num_spins)
            iy = np.random.randint(num_spins)

            # Boundary conditions
            left = spin_matrix[ix - 1, iy] if ix > 0 \
            else spin_matrix[num_spins - 1, iy]
            right = spin_matrix[ix + 1, iy] if ix < (num_spins - 1) \
            else spin_matrix[0, iy]

            above = spin_matrix[ix, iy - 1] if iy > 0 \
            else spin_matrix[ix, num_spins - 1]
            below = spin_matrix[ix, iy + 1] if iy < (num_spins - 1) \
            else spin_matrix[ix, 0]

            # Change of energy
            delta_energy = (2 * spin_matrix[ix, iy] * \
            (left + right + above + below))

            # Condition for accepting the new configuration
            if np.random.random() <= np.exp(-delta_energy / temperature):
                spin_matrix[ix, iy] *= -1.0
                E = E + delta_energy
                M = M + 2*spin_matrix[ix, iy]
                accepted += 1
        exp_values[i-1,0] = E
        exp_values[i-1,1] = M
        exp_values[i-1,2] = E**2
        exp_values[i-1,3] = M**2
        exp_values[i-1,4] = np.abs(M)
        exp_values[i-1,5] = accepted

return exp_values
```

4 Results

4.1 Analytic solution vs. numerical simulation

To test the algorithm and implementation we have found the analytic solution for a simple 2×2 lattice, for temperature, $T = 1.0$ conveniently scaled as $[kT/J]$. The calculations are implemented in the script *Analytic2x2.py*.

Monte Carlo cycles	$\langle E \rangle$	C_v	χ	$\langle \mathcal{M} \rangle$	$\langle \lvert \mathcal{M} \rvert \rangle$
100	-2.0	0.0	0.0	1.0	1.0
1000	-1.998	0.015984	3.262551	0.4285	0.9995
10000	-1.9972	0.02236864	3.97301984	-0.0748	0.9991
100000	-1.9958	0.03352944	3.98213984	-0.05201	0.99858
1000000	-1.995746	0.03399161	3.98706858	-0.038293	0.998593
10000000	-1.9959762	0.03212884	3.992483	0.0142267	0.9986581
Analytic	-1.99598	0.03208	3.9933	0.0	0.9986607

Table 2: Caption

From Table 2 one can see that the agreement between numerical and analytic results increases with number of Monte Carlo cycles. As the computations of large number of cycles are heavy, it is desirable to limit this number and still get sufficient agreement.

4.2 Equilibrium state

We are interested in finding the steady state, or equilibrium state. The time it takes for the system to reach steady state can be found as the number of Monte Carlo cycles needed, for the mean energy and magnetization, to stabilize around their mean values. When this thermalization time is reached, we will be able to get meaningful results of the expected values. To find out when the equilibrium situation is reached, we evaluate the mean energy and mean absolute magnetization as a function of Monte Carlo cycles. The results are shown in figure 2 and 3 for the temperature $T = 1.0$, and in figure 4 and 5 for temperature $T = 2.4$.

All figures show calculations for a 20×20 lattice, with two different starting configurations for the orientation of the spins - a random spin orientation and a ordered spin orientation (all spins initialized as positive). The calculations can be found in the script *Ordered_Random.py*.

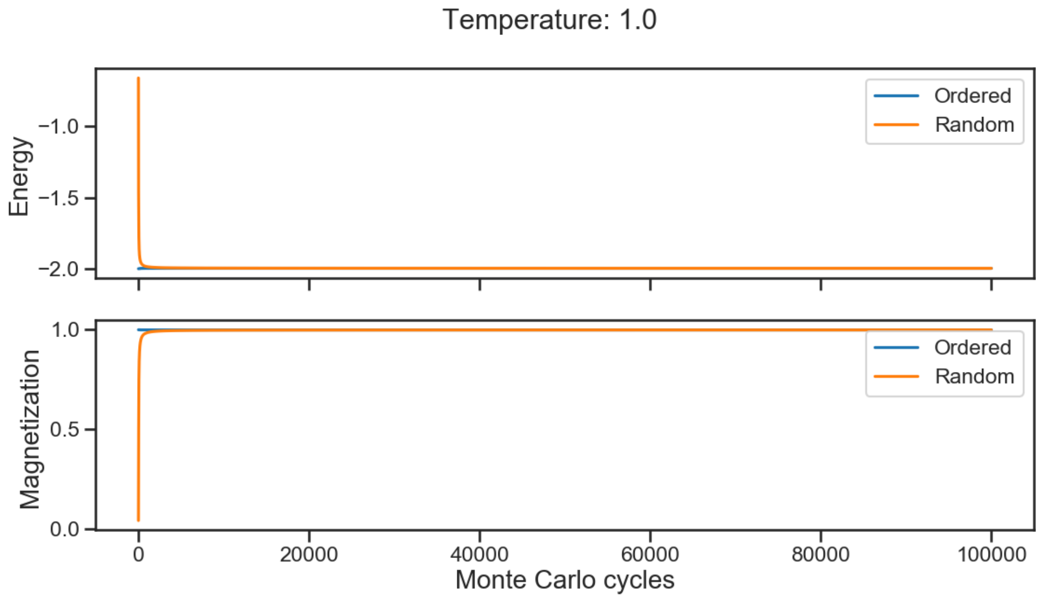

Figure 2: Expected energy and magnetization as function of Monte Carlo cycles.

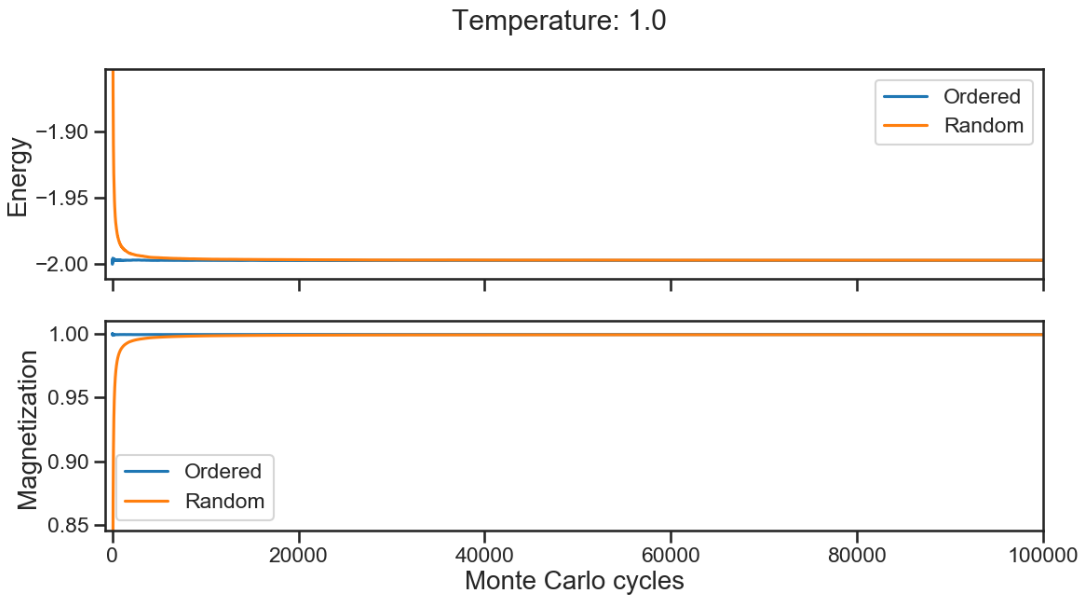

Figure 3: Zoomed. Expected energy and magnetization as function of Monte Carlo cycles.

Figure 2 and 3 show the evolution of $\langle E\rangle$ and $\langle|\mathcal{M}|\rangle$ as function of Monte Carlo cycles at temperature $T = 1.0$, with both ordered and random initial configuration. It is clear that both these variables reaches a equilibrium state on fewer Monte Carlo cycles when the start configuration is ordered.

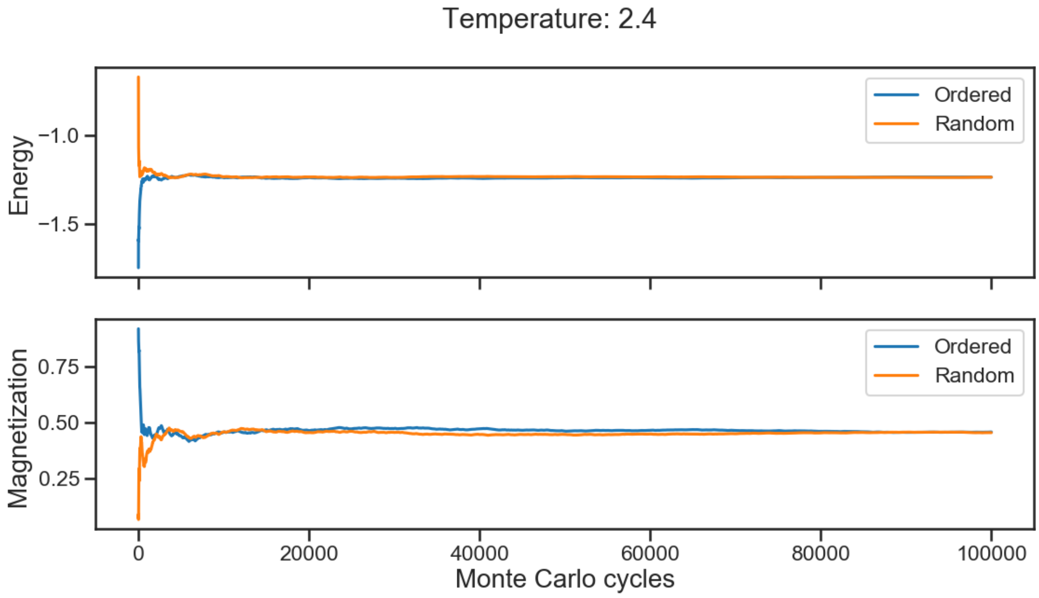

Figure 4: Expected energy and magnetization as function of Monte Carlo cycles.

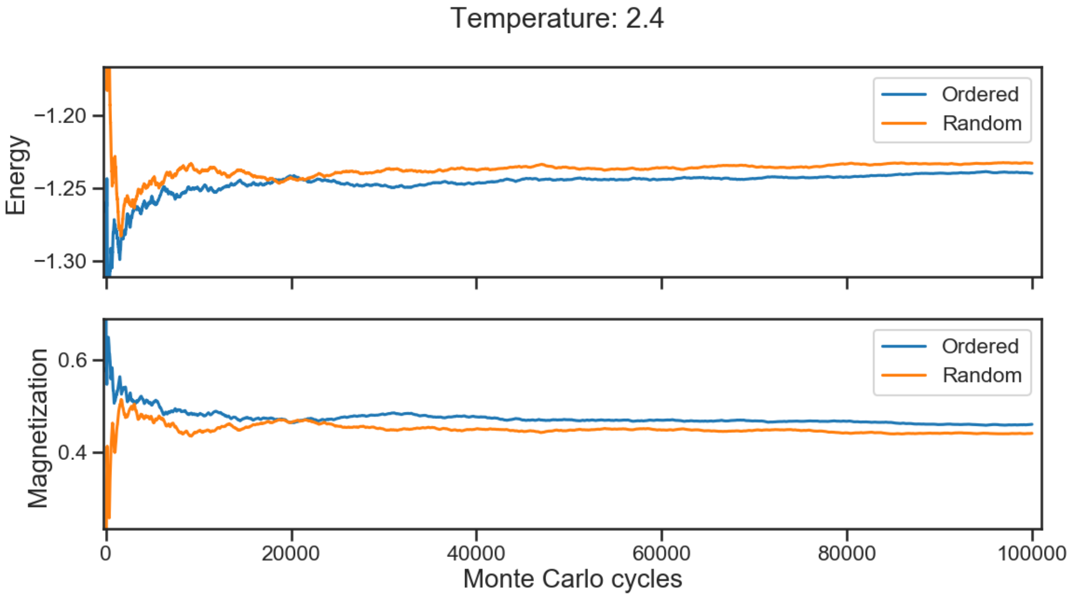

Figure 5: Zoomed. Expected energy and magnetization as function of Monte Carlo cycles.

In figure 4 and 5 you can see the evolution of $\langle E \rangle$ and $\langle |\mathcal{M}| \rangle$ as function of Monte Carlo cycles at temperature $T = 2.4$. At this temperature it is not as evident for which start configuration steady state is reached first, they both oscillate quite a bit for a lower number of Monte Carlo cycles.

4.3 Accepted configurations

Figure 6 show the number of accepted configurations as a function of Monte Carlo cycles for the two temperatures of interest, plotted by initial configuration, for a 20×20 lattice.

When a configuration is accepted, the total energy is supposed to be lowered, eventually reaching the energy minimum for a given temperature.

By plotting the number of accepted configurations on a logarithmic scale, we see that the acceptations evolve towards a linear state, for both temperatures and both initial configurations. That means that when the Monte Carlo simulation has *'spinned up'*, the number of accepted configurations evolve proportional to the number of Monte Carlo cycles.

Generally there are fewer accepted configurations when the initial configuration is ordered. This is due to the fact that the metropolis algorithm only accepts

energy change $\Delta E \leq 0$ and $\Delta E < e^{-\beta \Delta E}$, and the first case does not occur in the beginning for a initial configuration set to ground state.
The number of accepted configurations at $T = 2.4$ are generally higher than for $T = 1.0$. With a random initial configuration we notice that we have fewer accepted configurations for $T = 2.4$ than $T = 1.0$ in the first 1000 Monte Carlo cycles. Another thing worth noticing is that the linear growth in accepted configurations is reached faster with random start configuration when the temperature is 2.4, while for $T = 1$ this is achieved faster with ordered start configuration.

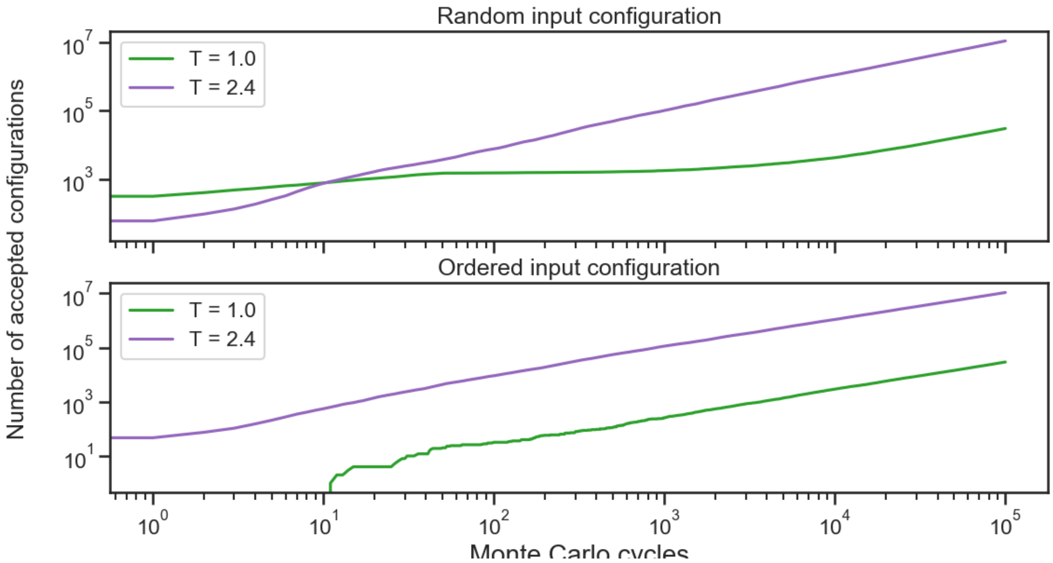

Figure 6: Number of accepted configurations as function of Monte Carlo cycles. Plotted for T = 1.0 and T = 2.4 with both random and ordered initial configuration.

The code for this calculation can be found in the script *Accepted.py*.

4.4 Analyzing the probability distribution

By computing the probability distribution $P(E)$, i.e. the probability of finding the energy of the system at a given level, we get figure 7 and 8, for the temperature $T = 1.0$ and $T = 2.4$ respectively. Both plots are results of computations with a 20×20 lattice after the system has reached steady state. For $T = 1.0$ the results are produced by choosing the Monte Carlo cycles between 20000 and 120000, and between 100000 and 200000 for $T = 2.4$.

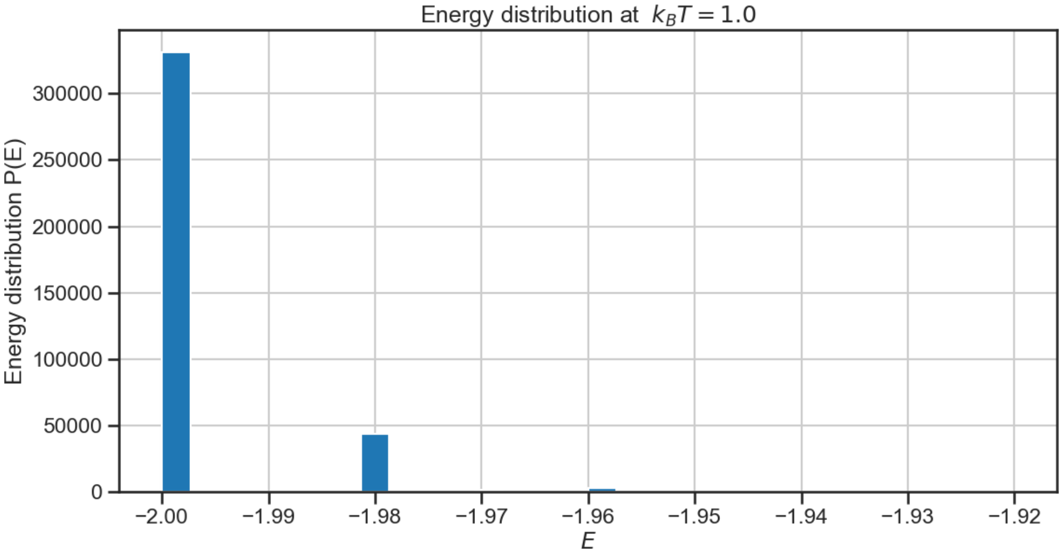

Figure 7: Probability distribution of the Energy, Temperature = 1.0

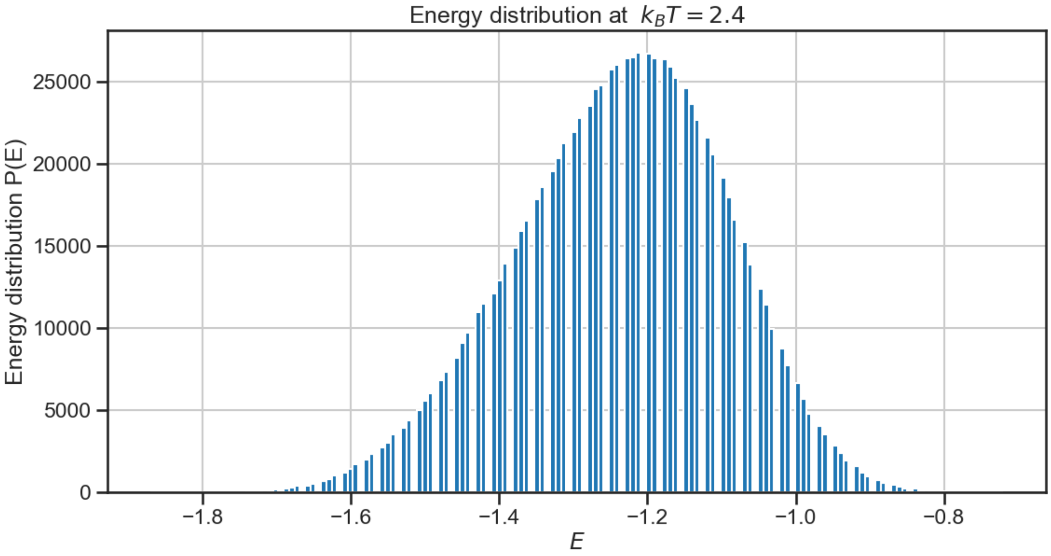

Figure 8: Probability distribution of the Energy, Temperature = 1.0

The caculations are done by the script *Probability_Distribution.py*

The program also calculates the variance of the energy for the two choices of temperature and the results are given below.

```
Variance T=1.0: 0.01946897479327163
Variance T=2.4: 8.099512708997135
```

4.5 Phase transitions

To study the behaviour of the Ising model when we get close to the critical temperature, we use the program *Phase_Transition.py*, which produces plots of the expectation values $\langle E \rangle$ and $\langle |M| \rangle$, as well as plots of the specific heat C_V and the susceptibility χ. All of these thermodynamic quantities are plotted as functions of the temperature. This is done for lattices of sizes 40×40, 60×60, 80×80 and 100×100. The temperature range is $T \in [2.0, 2.5]$ with a step length $\Delta T = 0.01$. The number of Monte Carlo cycles is 10^6, and the results from the first 40000 cycles are discarded since we want to have reached steady state before collecting results. The results are shown in Figure 9.

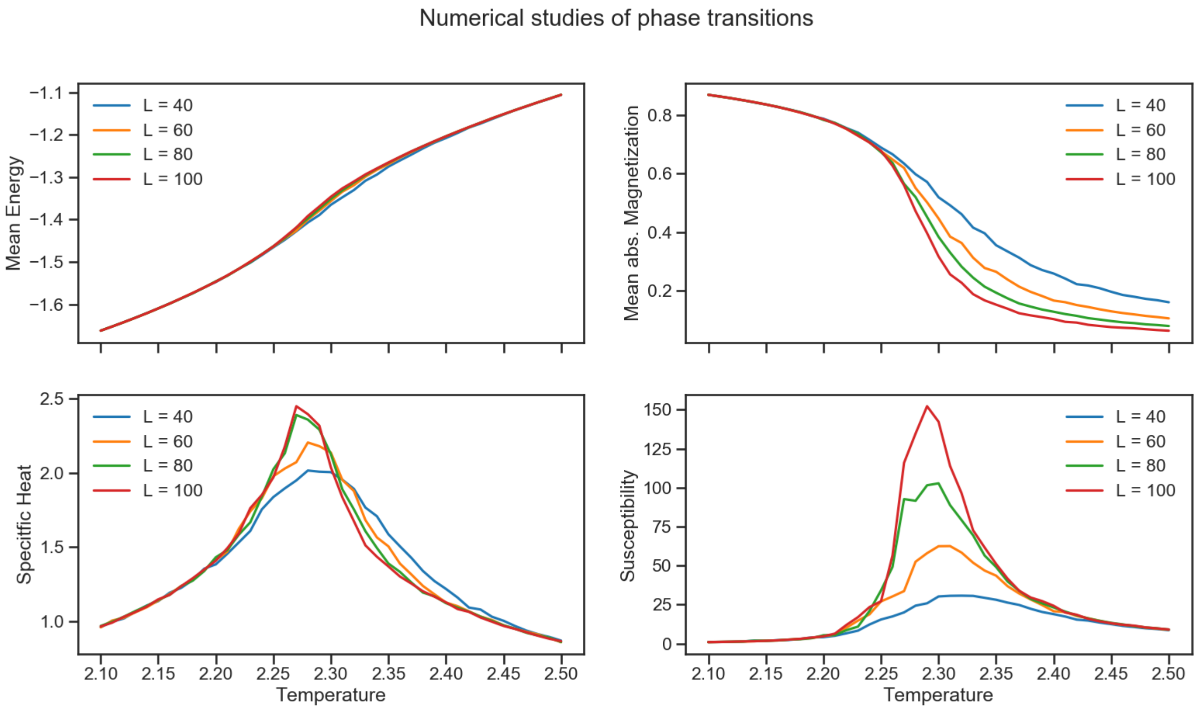

Figure 9: $\langle E \rangle, \langle \mathcal{M} \rangle, C_v$ and χ as functions of temperature.

We clearly see the phase transition in the plots of specific heat and susceptibility as they peak around $T = 2.3$. We can also see that the energy and magnetization changes character around this temperature.

4.6 Critical temperature in the thermodynamic limit

By implementing the approximation of the critical temperature for a infinite lattice (that we found above), to the script *Phase_Transition.py*, we can use the results from our long run to make this aproximation. The implementation was done as follows

```python
#Calculating the critical temperature for an infinite lattice
max_2 = np.argmax(X[2, :])
max_3 = np.argmax(X[3, :])
Tc_L2 = T[max_2]
Tc_L3 = T[max_3]
# pre-calculating the exponent
#v = 1.0
#exponent = -(1./v)
exponent = -1.0
Tc_infinity = Tc_L2 - ((Tc_L2-Tc_L3)/(L[2]**exponent-L[3]**exponent))*(L[2]**exponent)
print(Tc_infinity)
```

Here we found the index of the maximum value of the susceptibility from the 80×80 and 100×100 lattices and used that to find the critical temperatures for the two systems, necessary for our approximation. The result is the following

```
Approximated Curie Temperature: 2.2499999999999996
```

4.7 Compilation by Numba

To make sure that using Numba in the Python script *ising_model.py* speeds up the code, we tested the program *Ordered_Random.py* which imports the main function *MC* from the *ising_model.py*. We ran the code for 1000 Monte Carlo cycles. By using Numba the run time was 0.032258 seconds, and by using pure Python the run time was 2.503138 seconds. We have gained CPU-time by several orders of magnitude.

5 Discussion

To test the algorithm and its implementations, we compare the analytic results for a 2×2 lattice to numerical simulations for different number of Monte Carlo cycles. We clearly see that the approximation gets better for larger number of cycles and that the approximation we get by 1 million cycles is sufficiently good.
With increased number of cycles we also get increased computation time so it is debatable if a number of the order of 10^5 is sufficient for our purpose.

For a larger 20×20 lattice, we are interested in the number of cycles needed to reach an equilibrium state.
We found, from figure 2 that the ordered initial configuration clearly leads to steady state first, and that 2×10^4 is sufficient to reach steady state for $T = 1.0$, for both initial configurations.
In the study of the number of accepted configurations, figure 6, we see that for $T = 1.0$ and ground state initial configuration, the energy can not be lowered before the lattice is 'spinned up'. The number of accepted configurations then start to follow a linear evolution, proportional to the number of Monte Carlo cycles.

For larger temperatures, figure 4 it is not so evident for which initial configuration the system reaches steady state first. After 40000 Monte Carlo cycles we say that the steady state is sufficiently established for both initial configurations, at this temperature.
Again comparing to the accepted configurations in figure 6 we see that the number is smaller, and it reaches a linear fit earlier, when the initial configuration is ordered.

We note that what type of initial configuration we choose is not important, as we start the calculations of expectation values after steady state is reached, by cutting away the first $2 - 400000$ cycles and thus discarding the unwanted contributions to the expectation values.

Figure 7 and 8 show the probability distribution of the energy, after equilibrium state is reached, for $T = 1.0$ and $T = 2.4$ respectively. When the temperature is 1.0 the energy takes few different values, $E = -2$ for most states. For $T = 2.4$ the energy is more evenly distributed and we can clearly recognize the Boltzmann distribution.
There is, in other words, not much happening in the system when the temperature is low. This fits well with basic intuition on thermodynamics as well as with the results for the accepted configurations.
The variance that we got from this calculation does not seem to be correct for

$T = 1.0$. The value seems to be too large, as this temperature allows only a few energy configurations.

In figure 9 we find our study of the phase transition. The transition can clearly be seen in the plots of the susceptibility $\chi(T)$ and specific heat $C_v(T)$, where we have the characteristic peak.
In the plots of mean energy $\langle E(T)\rangle$ and mean absolute magnetization $\langle|\mathcal{M}|(T)\rangle$ we can also see a change of character around the same temperature.
When evaluating the importance of the size of the lattice, it is evident that the larger the lattice gets, the more pronounced the phase transition is.
This makes sense, since the lattice size should be an approximation to infinity, where there would be a discontinuity or divergence in the point of the phase transition.

The exact result for the critical temperature, found by Lars Onsager, is
$kT_c/J = 2/ln(1+\sqrt{2}) \approx 2.269$. Our approximation to this temperature is
$T_C[kT/J] = 2.24999 \approx 2.25$, which is not so bad.

6 Conclusion

In this project we employed the Ising model in two dimensions and the Metropolis algorithm to model the phase transition from ferromagnetic to paramagnetic material.
We found that we need a certain number of Monte Carlo cycles to reach steady state, before collecting the variables for our ensemble and from that calculate the expectation values of energy and magnetization. This number of Monte Carlo cycles can be seen as a measure of the time needed to reach a steady state situation. It seems like the time needed to reach steady state is longer when the material has gone through the phase transition.

In our studies of the phase transition we got some promising results and a relatively good approximation to the Curie temperature for a lattice of infinite size. This result could probably have been better by choosing a even smaller temperature step size, larger number of Monte Carlo cycles and larger lattice sizes.

Final remarks

The project has been developed in collaboration with Silje Christine Iversen and Dina Stabell.

Link to the repository

Programs and plots can be found at the GitHub repository:
https://github.com/krisgarv/FYS4150/tree/master/Problemsets/4th

The folder named *CODES* contains all of the codes used in this project, while the folder *PLOTS* contains all of the plots used in this report.

References

[1] Hjorth-Jensen, M. (2015). *Computational Physics - Lecture notes Fall 2015.* University of Oslo.

[2] Canonical ensemble. (n.d.). Britannica. Retrieved November 10, 2018, from https://www.britannica.com/science/canonical-ensemble

[3] Metropolis–Hastings algorithm. (n.d.). Wikipedia. Retrived November 17, 2018, from https://en.wikipedia.org/wiki/Metropolis–Hastings_algorithm

[4] Ising model. (n.d.). Wikipedia. Retrived November 17, 2018, from https://en.wikipedia.org/wiki/Ising_model