Project 4

FYS4150 Computational Physics

Even S. Håland

Abstract

In this project we perform Monte Carlo simulations of magnetic systems using the Ising model. We do comparisons between our simulations and a analytically solvable system, as well as studies of equilibration time for larger systems. We also study phase transitions in the model, and estimate the critical temperature in the thermodynamical limit to be $k_B T_C / J = 2.269$, which is in very good agreement with the exact value for this quantity.

1 Introduction

The aim of this project is to do Monte Carlo simulations of magnetic systems described by the two-dimensional Ising model. The Ising model describes binary systems, that is a collection of objects with only two possible states each. Here we will take these two states to be spin-up and spin-down. In particular we will study phase transitions in such systems.

We will start by studying some analytical solutions to the Ising model by considering a 2×2 lattice. Then we will make a code that simulates the evolution of such systems for larger lattices. We will see how the system evolves depending on temperature and the initial state of the system, and we will finally study phase transitions, and extract the critical temperature in the thermodynamical limit.

In our simulations we will use the Metropolis algorithm for accepting or rejecting the Monte Carlo trials. We will also, as these simulations are quite time consuming, parallelize the code, so that we can distribute our jobs to multiple processors.

2 Theoretical framework

2.1 The Ising model

As mentioned in the introduction the Ising model describes a collection of objects, where each object can take two values, i.e. spin-up and spin-down, parametrized by $s = \pm 1$. We will consider a two-dimensional model with the spins organized in a square lattice with dimension L, so the total number of spins is given by $N = L \times L$. The two key quantities for describing our system will be energy and magnetization. The total energy of this system (with no external magnetic field) is given by

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

where J is a coupling constant and $< kl >$ indicates that we sum over neighbouring spins only, i.e. one object is only "talking" to its closest neighbours. The total magnetization of the system is simply given by the sum over all spins:

$$M = \sum_{k}^{N} s_k. \tag{2}$$

A natural next step is to calculate expectation values within this theory, i.e. mean energy, $\langle E \rangle$, and mean magnetization $\langle M \rangle$. To be able to do so we need to introduce some concepts from statistical physics.

2.2 Elements from statistical physics

In order to calculate expectation values we need a probability distribution. In our case this is the Boltzmann distribution, given as

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

where $\beta = 1/k_B T$ (where k_B is Boltzmann's constant and T is temperature), E_i is the energy of state i and Z is the partition function. The partition function is given as

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

where the sum runs over all possible (micro)states of the system. The expectation value of some general quantity A can now be calculated as

$$\langle A \rangle = \sum_{i=1}^{n} A_i P_i(\beta) = \frac{1}{Z} \sum_{i=1}^{n} A_i e^{-\beta E_i},$$

and the variance of A is given as

$$\sigma_A^2 = \langle A^2 \rangle - \langle A \rangle^2 .$$

By considering the variances of E and M we can also calculate specific heat, C_V, and susceptibility, χ, as

$$C_V = \frac{1}{k_B T^2} \sigma_E^2 \quad \text{and} \quad \chi = \frac{1}{k_B T} \sigma_M^2 .$$

2.3 Analytical solutions

Using the above described framework we can study the Ising model for a 2×2 lattice analytically. This means that we have 4 objects which each can have $s = \pm 1$. Both in this and later analysis we will make use of so-called periodic boundary conditions, which means that we let the first and last spins in a lattice row be neighbours when summing up the energy.

We label the spins in the 2×2 lattice with $k \in \{1, 2, 3, 4\}$, and the energy is then given by

$$\begin{aligned} E &= -J(s_1 s_2 + s_2 s_1 + s_1 s_3 + s_3 s_1 + s_2 s_4 + s_4 s_2 + s_3 s_4 + s_4 s_3) \\ &= -2J(s_1 s_2 + s_1 s_3 + s_2 s_4 + s_3 s_4), \end{aligned}$$

and the magnetization as

$$M = s_1 + s_2 + s_3 + s_4 .$$

Table 1: Degeneracy, energy and magnetization for the different spin configurations in the 2×2 lattice.

# spin-up	Degeneracy	E	M
4	1	$-8J$	4
3	4	0	2
2	2	$8J$	0
2	4	0	0
1	4	0	-2
0	1	$-8J$	-4

Table 1 lists the possible values for these quantities for all possible spin configurations, along with the degeneracy for each spin configuration. Using this we find that the partition function is given as

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

Using this along with the expressions from the previous section we can calculate $\langle E\rangle$, $\langle|M|\rangle$, C_V and χ, and we arrive to the following expressions:

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

$$\langle|M|\rangle = \frac{4+2e^{8\beta J}}{3+\cosh(8\beta J)}$$

$$C_V = \frac{192J^2\sinh(8\beta J)}{k_B T^2\left[3+\cosh(8\beta J)\right]^2}$$

$$\chi = \frac{4\beta\left(1+2e^{8J\beta}\right)}{3+\cosh(8\beta J)}$$

These expressions (with inserted values for T and J) can be used as a check on our simulations.

2.4 Phase transitions

In important part of this project is to study phase transitions in our system. The Ising model with no external magnetic field undergoes a second order phase transition at some critical temperature, T_C, which is such that $\langle M\rangle = 0$ for $T > T_C$. In the thermodynamical limit (i.e. $L \to \infty$) C_V and χ are divergent at T_C. The lattices we study here will (of course) be of finite dimensions, and C_V and χ will not diverge, but rather show a maximum around T_C.

The critical temperature is given by

$$T_C(L) - T_C(L=\infty) = aL^{-1/\nu} \tag{3}$$

where a is a constant and ν is given by

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

where ξ is called the *correlation length*, which defines the length scale at which microscopic variables are correlated. By measuring the critical temperature for finite sized lattices we can extract T_C in the thermodynamical limit ($T_C(L=\infty)$), which is done towards the end. We do however need estimates of T_C for two different L's, since equation 3 contains two unknown quantities ($T_C(L=\infty)$ and a). The exact result (calculated by Lars Onsager) is given by

$$kT_C/J = \frac{2}{\ln(1+\sqrt{2})} \approx 2.269 \tag{4}$$

when $\nu = 1$.

3 Code

All code written for the project can be found in the following git-repository:

https://github.com/evensha/FYS4150/tree/master/Project4/Programs

The three most important scripts in this repository are:

- `main.cpp`
- `Plotting.py`
- `Phase_trans.py`

The first one is the C++ program that performs the Monte Carlo simulation, while the two other are python scripts used for evaluating and plotting the output from the simulation.

The repository also contains the C++ library (`lib.cpp`/`lib.h`) provided for the course, as well as a `makefile` and the executable file `Program.x`.

3.1 The main program

The set-up of the simulation (i.e. the `main.cpp` program) follows very closely the approach from the example programs in Chapter 13 of the lecture notes [1], but with some modifications. Paralleliziation of the program using Open MPI is also implemented as suggested in the lecture notes. A slight difference is however the usage of Armadillo [2] for handling vectors and matrices.

The first thing we do in the main program is to define some useful functions:

- `periodic`: Handles the periodic boundary conditions when summing up the energy of a lattice.
- `initialize`: Initialize the spin matrix and calculates initial energy and magnetization. The spin matrix is initialize either with ordered spins (all spins up), or a random spin configuration. Random spins are generated using the `ran1` function from the library (`lib.cpp`).
- `Metropolis`: Implements the Metropolis algorithm with the Metropolis test. This will be discussed in a bit more detail later.
- `output`: Write final output (e.g. mean E and M, C_V and χ) to file.

The three latter functions are implemented in the end of the program.

Next we enter the main function. This function takes three input arguments: lattice size, number of Monte Carlo cycles and an integer which indicates whether or not we want a random initial spin state. In the beginning of this function we set up all necessary quantities, vectors and matrices, and we also initialize MPI.

Then we start a loop over the temperatures we want to consider. In our analysis we will mostly consider two situations; either a lattice with $L = 20$ and temperatures

$T = 1.0$ and $T = 2.4$, or larger lattices with $L = 40, 60, 80, 100$ and $T \in [2.0, 2.3]$. For this reason the program is tailored so that the temperatures taken into account depends on the lattice size given as input.

For each new temperature we call the `initialize` function, and we also set up an array containing the five possible energy differences (or rather the ration between probabilities for different energies, $\exp(-\beta\Delta E)$), and an array for storing the final average values for a given temperature. Then we call the initialize function which calculates initial energy and magnetization. For the $L = 20$ case we would like to study the evolution of the mean values, so in this case we make an output file to which we write updated average values after each MC cycle.

Finally we are ready to start running the Monte Carlo cycles. For each cycle we call the `Metropolis` function, update the average values and write to file (if required). When the MC loop is finished we write the final average values to file, and move to the next temperature.

3.2 The Metropolis algorithm

As mentioned the Metropolis algorithm [3] is implemented in the `Metropolis` function. The idea of this algorithm is briefly explained in the following.

We start by establishing an initial state with some energy, E_0, which is done in the `initialize` function. Then we loop over the total number of spins. For each turn in the loop we pick out a random lattice position, and try to flip this spin. Then we compute the new energy, E_1, and look at the energy difference between the new and the previous state, $\Delta E = E_1 - E_0$. If $\Delta E \leq 0$ we accept the new spin configuration, while if $\Delta E > 0$ we pick a random number r, and compare it to the probability ration $w = \exp(-\beta\Delta E)$. If $r \leq w$ we accept the configuration.

Whenever a configuration is accepted we need to update energies and magnetization. The algorithm is then repeated until we get the desired precision in our simulation. Each loop over total number of spins is called a *Monte Carlo cycle*.

4 Results

We previously calculated analytical expressions for the expectation values of E and $|M|$, as well as the heat capacity, C_V, and the susceptibility, χ. By inserting $T = J = k_B = 1.0$ in these expressions we can compare the values with those obtained in our MC simulation. This is done in Table 2 for various number of MC cycles. We see that by using a million MC cycles we get good agreement between analytical and simulated values.

Now that we have established that our algorithm is able to produce reasonable results we would like to extend the lattice. We start by considering $L = 20$, and want to see how many MC cycles we need for the system reach equilibrium by studying the evolution of the mean energy and the mean magnetization. Results for two different temperatures ($T = 1.0$ and $T = 2.4$) are shown in Figure 1 (all spins up initially) and 2 (random initial spins).

When starting with all spins up we see that the averages stabilizes quite fast when $T = 1.0$ ($\sim 3,000$ MC cycles), while it takes considerably longer when $T = 2.4$

Table 2: Comparison between analytically calculated and simulated expectation values for the various thermodynamical quantities for the 2×2 lattice.

		# MC cycles		
Parameter	Analytical value	10^4	10^5	10^6
$\langle E \rangle$	-7.984	-7.967	-7.982	-7.984
$\langle \lvert M \rvert \rangle$	3.995	3.989	3.994	3.995
C_V	0.128	0.261	0.141	0.129
χ	15.97	15.93	15.95	15.97

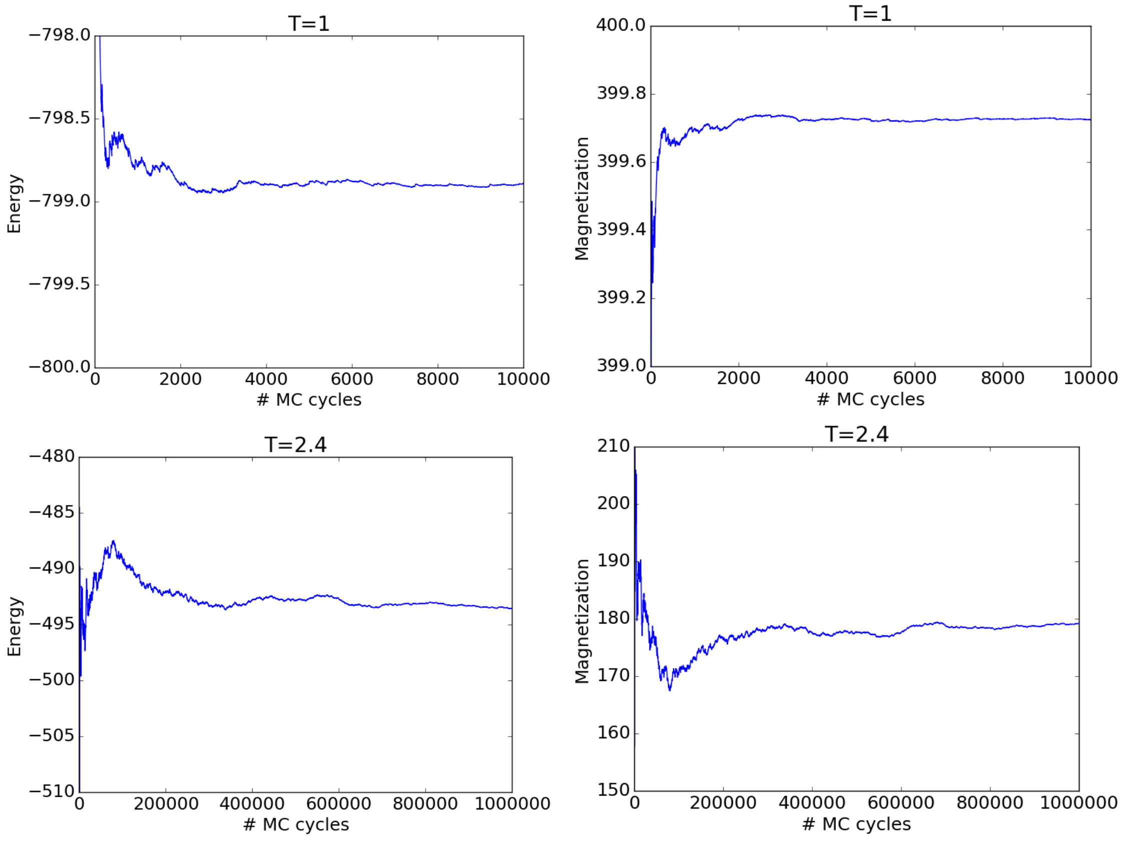

Figure 1: Mean energy and magnetization as function of number of MC cycles for $L = 20$ for two different temperatures with all spins up initially.

($\sim 300,000$ MC cycles). We also see that for $T = 2.4$ there are more fluctuations, also after the stable state is reached With random initial spins we see that we get very smooth curves for $T = 1.0$. This is probably because the most likely states are reached quite quickly, so the energy/magnetization stays more or less the same, so that the average is smoothly converging. However, for $T = 2.4$ we again see more fluctuations, but the equilibrium is actually reached faster than with ordered initial spins.

The behaviour we observe is governed by the Boltzmann distribution, $e^{E/k_B T}$.

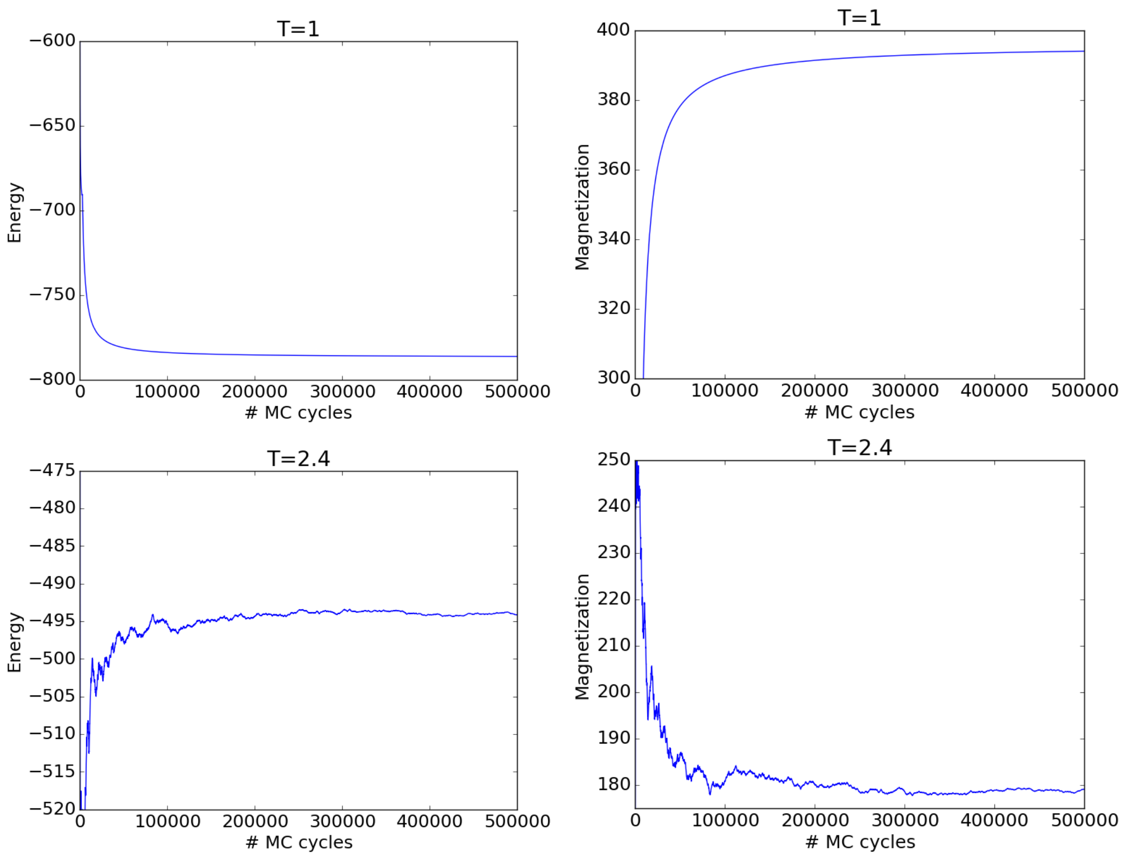

Figure 2: Mean energy and magnetization as function of number of MC cycles for $L = 20$ for two different temperatures with random initial spins.

This basically tells us that by increasing the temperature we increase the likelihood for entering a larger number of states. This is also supported by Figure 3, which shows how the number of accepted states in our simulation evolves with number of MC cycles, as well as Figure 4, which shows the energy probability distribution. The variances for the energies is calculated to be $\sigma_E^2(T = 1.0) = 8.6$ and $\sigma_E^2(T = 2.4) = 3219$, which is also in good correspondence with Figure 4, as energies varies much more when the temperature is increased.

4.1 Phase transitions

Finally we will have a look at phase transitions in the Ising model. In Figure 5 the quantities $\langle E \rangle$, $\langle |M| \rangle$, C_V and χ are plotted as function of temperature for four different lattices sizes, $L = 40, 60, 80, 100$. The simulations are done with a temperature step of $\Delta T = 0.01$, and with $5 \cdot 10^5$ MC cycles for $L = 40, 60, 80$ and 10^6 MC cycles for $L = 100$. Both C_V and χ peaks towards the end of the temperature range, while the magnetization falls towards zero, indicating a phase transition.

However, the curves are not exactly beautiful, which could probably be improved by increasing the number of MC cycles. This was not done due to the time consumption of these simulations, in particular for the largest lattices, even though

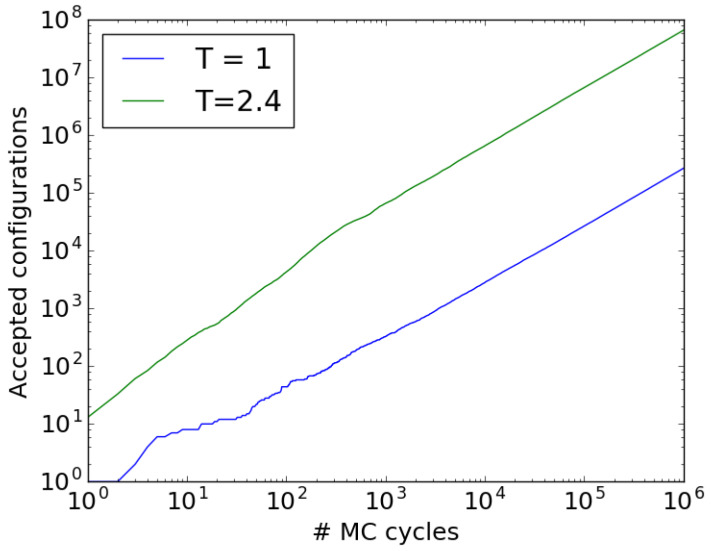

Figure 3: Number of accepted spin configurations for $T = 1.0$ and $T = 2.4$.

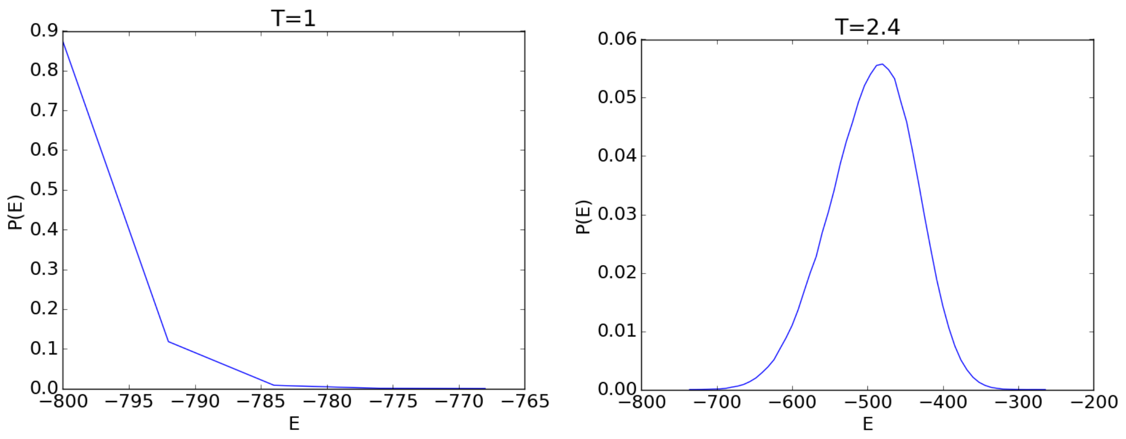

Figure 4: Probability distribution for energy states with $T = 1.0$ and $T = 2.4$.

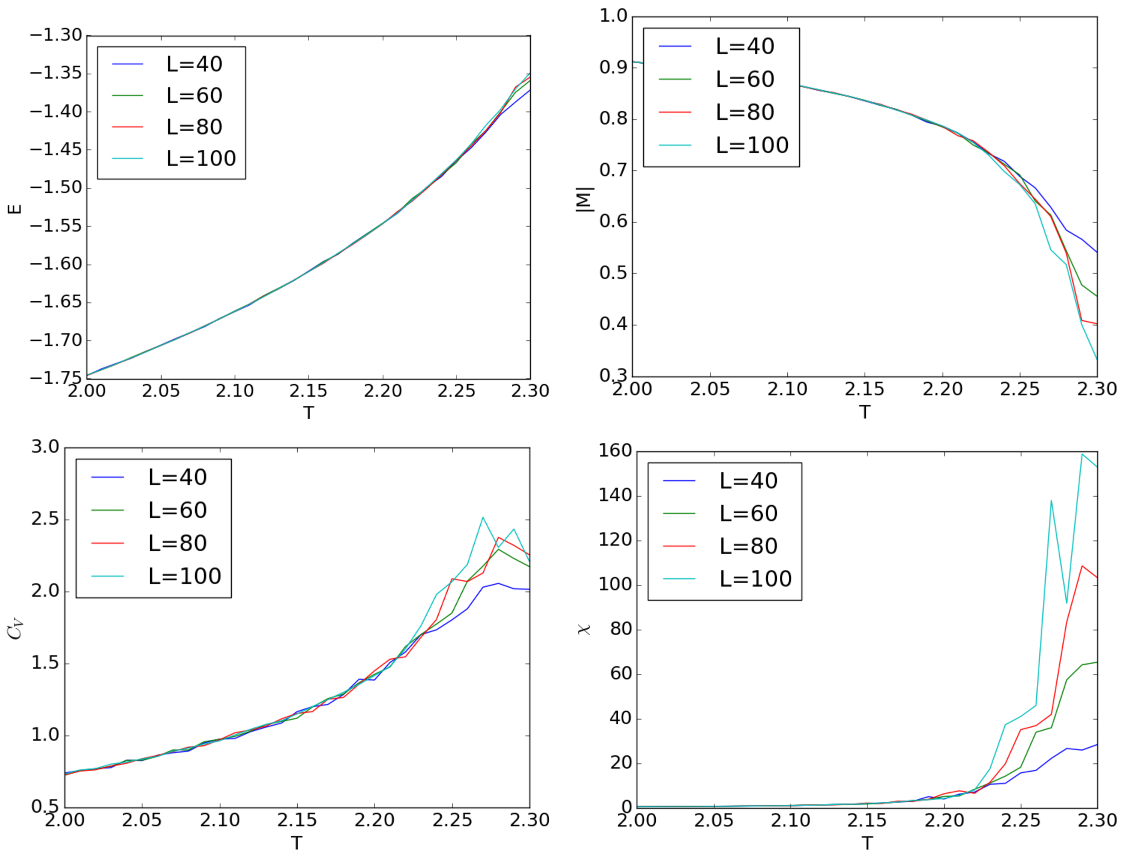

Figure 5: $\langle E\rangle$, $\langle|M|\rangle$, C_V and χ plotted as function of temperature for various lattice sizes.

Table 3: Time consumption for simulation of 500, 00 MC cycles of a $L = 40$ lattice, using different number of processors.

# processors	Time
1	24.6 min
2	12.4 min
4	9.8 min

our program is parallelized. Time consumption for simulation of a $L = 40$ lattice with 500, 000 MC cycles is shown in Table 3, using 1, 2 and 4 processors. When moving from one to two processors we see that we get a very nice speed up, i.e. the time is decreased by a factor of ~ 2. When moving to four processors we are able to speed it up a little bit more. However, four is the total number of processors on my laptop, meaning that the full capacity of all four processors will not be used for the MC simulation.

The final step is to extract the critical temperature in the thermodynamical limit, as discussed in section 2.4. When doing these calculations we need somewhat higher precision in the estimates of T_C, so (since we can established the approximate values of T_C from Figure 5), the simulations has been run again with the same number of MC cycles, but with $T \in [2.65, 2.85]$ and $\Delta T = 0.001$. By looking at C_V in the output files for $L = 60$ and $L = 100$ (found as `Output_L_60_dt0.001.txt` and `Output_L_60_dt0.001.txt` in the Output folder in the previously mentioned git repository) we find that $T_C(L = 60) = 2.279$ and $T_C(L = 100) = 2.275$. Using these values together with Equation 3 gives us

$$T_C(L = \infty) = \frac{2.279 - 100/60 \cdot 2.275}{1 - 100/60} = 2.269,$$

as critical temperature in the thermodynamical limit, which is precisely the same as the exact result. (I'm a bit surprised that the estimation was *that* accurate. It is not clear to me if this is some sort of magic coincidence, or if the simulation actually is that good.)

5 Summary and conclusions

In this project we have done Monte Carlo simulations of magnetic systems by using the Ising model. We found that we are able to obtain good precision in our simulations, compared to analytical values for a 2×2 lattice. We have also seen that the time need to reach equilibrium depends on both initial spin configuration and temperature, and that by increasing the temperature the energy will fluctuate more, as more states are available, as described by the Boltzmann distribution.

A key point of the project was the studies phase transitions in our systems, and we have extracted the critical temperature for such in the thermodynamical limit with surprisingly good precision. Finally, we have also seen that Monte Carlo simulation can be quite time consuming, meaning that parallelization of the program is of great value.

References

[1] M. Hjort-Jensen (2015), *Computational Physics - Lecture Notes Fall 2015*, Department of Physics, University of Oslo.
https://github.com/CompPhysics/ComputationalPhysics/blob/master
/doc/Lectures/lectures2015.pdf

[2] C. Sanderson, R. Curtin (2016), Armadillo: a template-based C++ library for linear algebra, *Journal of Open Source Software*, Vol. 1, p. 26.

[3] N. Metropolis et al. (1953), Equation of State Calculations by Fast Computing Machines, *The Journal of Chemical Physics*, Vol. 21(6), p. 1087.
https://doi.org/10.1063/1.1699114