

Project 4 - FYS4150

Jostein Brændshøi

Department of Physics
University of Oslo

November 19, 2017

Abstract

We study the evolution of thermal energy and spin interactions in a magnetic material using the 2D Ising model and Monte Carlo simulations. We implement a parallel Ising model and look statistical properties of the system for various lattice sizes and number of Monte Carlo cycles. We find that the convergence towards the most likely state (and the most likely state itself) depends strongly on temperature in correspondence with the underlying Boltzmann distribution. We also find that different initial spin configurations improves or worsen the convergence. Finally we look at phase transitions and see that the Ising model gives rise to second order phase transitions that are more pronounced for larger lattice sizes. Also the associated critical temperature is found to be in fairly good agreement with Lars Onsager (1944) exact results [4].

The address <https://github.com/jostbr/FYS4150-Projects/tree/master/Project4> on GitHub is associated with this project. Here one can find all code used in the project. This includes C++ source files containing the core of the project, but also Python plotting scripts for result visualization. There are also available benchmark results from running the code as well as \LaTeX source for this PDF. The parameters behind the benchmark simulation results are described throughout this report. Also, if one wishes closer plot inspection, running the Python visualization script reproduces all plots in this report directly using the data files in the benchmark directory.

1 Introduction

The use of statistics in science is a continuously growing area and make huge impacts in a range of fields. Stochastic methods are becoming increasingly popular and especially as powerful computers are more and more easily accessible (even as laptops in many cases), the potential for such methods continue to grow. Monte Carlo methods are even the methods of choice in many areas of physics, like for example statistical physics and in particular solid state physics. Hence, gaining experience with Monte Carlo methods may be very valuable for a physicist. This brings us to the contents of this project.

In this project we will employ methods in statistical physics and implement the binary 2D Ising model in order to simulate spin interactions in e.g. a solid metal. In this model each spin in the material can only take on two directions, either $+1$ or -1 , hence it is a binary model. Working with the Ising model, we study the response of a magnetic material to thermal energy. In particular, we look at the evolution of the magnetic spin configuration in the material that undergoes neighbouring spin interactions as response to thermal energy. As is a very popular use of the Ising model, we will apply it to see if we can identify phase transitions [2].

Specifically, the structure of this report involves firstly discussing some analytical results that will help us greatly when implementing the Ising model. Then we move on to some technicalities about the model before describing the verification process undergone to make sure the model behaves as it should. Mentions on code performance improvements (parallelization) are discussed. Then we discuss results with a main focus on the evolution of various statistical expectation values for the system and attempt to apply some of the underlying physics of the Ising model in order to

2 Theory

As an initial exercise it is useful to look at some analytical results for a small sample size lattice of, say, 2×2 . The usefulness in this exercise is mainly to have results to which we can compare the later obtained numerical solutions, but such an exercise also serves as a stepping stone into working with and understanding the Ising model. We consider a 2×2 lattice, meaning a total of four spins, and assume that each spin can only take on the two possible values ± 1 , i.e. a binary system as mentioned in section 1. We also use boundary boundary conditions giving the condition that the neighbour of an edge spin is at the opposite side of the lattice. We want to compute the partition function Z , expectation values of the energy E and absolute value magnetic moment $|\mathcal{M}|$, the specific heat C_v and the magnetic susceptibility χ .

Before we can obtain the expression for any of the mentioned quantities we have to find the energy and magnetic moment of each of the M possible spin configurations offered by the 2×2 lattice. Due to it being four spins each with two possible values (± 1) for the spin, we have a total of $M = 2^4 = 16$ different possible spin configurations in the lattice. From [2] and [3] we have the Ising model formula for computing the energy of configuration i by

$$E_i = -\mathcal{J} \sum_{\langle kl \rangle} s_k^i s_l^i \quad (1)$$

where s_k is the spin value of the k 'th spin, $\langle kl \rangle$ indicates a double sum, but only over nearest neighbours (i.e. e.g. for each k , l loops over the nearest neighbours of k) and \mathcal{J} is a constant

describing the strength of the interaction between neighbouring spins in the lattice. Computing the total magnetic moment of the lattice is a more straight-forward operation where we simply sum up all spin values according to [2]

$$\mathcal{M}_i = \sum_{j=i}^N s_j^i \quad (2)$$

Applying (1) and (2) to the 2×2 lattice with periodic boundary conditions, we get

$$E_i = s_1^i s_2^i + s_1^i s_3^i + s_2^i s_1^i + s_2^i s_4^i + s_3^i s_4^i + s_3^i s_1^i + s_4^i s_3^i + s_4^i s_2^i \quad (3)$$

$$\mathcal{M}_i = s_1^i + s_2^i + s_3^i + s_4^i \quad (4)$$

where in the nearest neighbour sum (for energy) we have only counted each neighbour spin product once (they appear twice due to periodic boundary conditions). If one were to draw out all 16 spin configurations, one would see that one could group the configurations into categories as displayed in table 2.1. In this table, (3) and (4) have been used for each of the $M = 16$ configurations. As can be seen, many of the configurations are duplicates of each other, e.g. the case with one spin equal to $+1$ and the other three equal to -1 gives four possible positions to find the $+1$ value, but they all produce the same energy and magnetic moment. This particular case corresponds with row 5 in the table.

Spins = $+1$	Duplicates	Energy	Moment
4	1	$-8\mathcal{J}$	4
3	4	0	4
2	4	0	0
2	2	$+8\mathcal{J}$	0
1	4	0	-2
0	1	$-8\mathcal{J}$	-4

Table 2.1: The energy and magnetic moment for all $M = 16$ spin configurations in a 2×2 lattice.

We are now in a position where we can move on to find the quantities initially mentioned in this (SUB)section. The Ising model is based on the Boltzmann distribution [2]

$$P'(\beta) = e^{-\beta E} \quad (5)$$

where the marked P indicates a non-normalized distribution and $\beta = 1/k_B T$ inversely proportional with the temperature T of the system under study, i.e. the lattice, and k_B is the Boltzmann constant. Firstly we compute the partition function for the distribution (for normalization of the distribution) by summing over all spin configurations [2]

$$Z = \sum_{i=1}^M P'_i(\beta) = \sum_{i=1}^M e^{-\beta E_i} = 2e^{8\beta\mathcal{J}} + 2e^{-8\beta\mathcal{J}} + 12 = 4 \cosh(8\beta\mathcal{J}) + 12 \quad (6)$$

Then we find the expectation values of the energy and magnetization as well as their square expectation values through the standard statistical expressions by summing over spin configurations and using the values in table 2.1:

$$\langle E \rangle = \frac{1}{Z} \sum_{i=1}^M E_i e^{-\beta E_i} = -\frac{32\mathcal{J} \sinh(8\beta\mathcal{J})}{Z} \quad (7)$$

$$\langle E^2 \rangle = \frac{1}{Z} \sum_{i=1}^M E_i^2 e^{-\beta E_i} = \frac{256\mathcal{J}^2 \sinh(8\beta\mathcal{J})}{Z} \quad (8)$$

$$\langle |\mathcal{M}| \rangle = \frac{1}{Z} \sum_{i=1}^M |\mathcal{M}_i| e^{-\beta E_i} = \frac{1}{Z} (8e^{8\beta\mathcal{J}} + 16) \quad (9)$$

$$\langle |\mathcal{M}|^2 \rangle = \frac{1}{Z} \sum_{i=1}^M |\mathcal{M}_i|^2 e^{-\beta E_i} = \frac{32}{Z} (e^{8\beta\mathcal{J}} + 1) \quad (10)$$

which further allows us to compute the heat capacity C_v (at constant volume) through [2]

$$C_v = \frac{1}{k_B T^2} (\langle E^2 \rangle - \langle E \rangle^2) = -\frac{1}{k_B T^2 Z} \left\{ 256\mathcal{J}^2 \cosh(8\beta\mathcal{J}) - \frac{[32\mathcal{J} \sinh(8\beta\mathcal{J})]^2}{Z} \right\} \quad (11)$$

and the magnetic susceptibility χ as [2]

$$\chi = \frac{1}{k_B T} (\langle |\mathcal{M}|^2 \rangle - \langle |\mathcal{M}| \rangle^2) = -\frac{1}{k_B T Z} \left[32(e^{32\beta\mathcal{J}} + 1) - \frac{(8e^{8\beta\mathcal{J}} + 16)^2}{Z} \right] \quad (12)$$

It's worth remembering that all these quantities are functions of the temperature T through $\beta = 1/k_B T$. We now have some analytical results for the 2×2 lattice case and this will serve as useful comparison and validation material for the implementation of the Ising model below.

3 Methods

3.1 Ising model

The methods used in this project are mainly related to the Ising model and the Metropolis algorithm [2]. We use stochastic methods, in particular Monte Carlo methods, in order to simulate the spin interactions in a 2D lattice as a response to the thermal energy of the system. the main idea with the method is that we start the lattice in some spin configuration (every spin is either $+1$ or -1), and then we start flipping spins and have a sophisticated rule for what spin flips to accept in order to go to a new spin configuration. Then we calculate the energy and magnetic moment for the new configuration and update the expectation values for these quantities. Then we repeat this for a large number of Monte Carlo cycles in order to obtain decent statistics. The above mentioned sophisticated rule is the Metropolis algorithm and defines the sampling rule for accepting a move to a new spin configuration. The code on GitHub is an implementation of the algorithm described in pseudo code on page 435 in [2]. It is attempted to have frequent comments in the code and more about the Ising model and Metropolis algorithm can be seen from there.

3.2 Model verification/testing

To make sure an implementation of a numerical model is reasonable, it is invaluable and very useful to have analytical results to which one can compare the numerical ones. Here, we are in a position where we have known analytical expressions, in the 2×2 lattice, for quantities of interest in equations (7) through (12). One might argue that the discussion in this particular subsection also could have been grouped together with the other results below in section 4, but primary use of the 2×2 case was under development and verification of the model. It lays the foundation before one could, with confidence, extend the simulations to the larger lattices where analytical expressions are harder to come by. Therefore the author decides it fits slightly better here.

One can view Table 3.1 as a summary of the verification tests of the implementation of the Ising model. Here we compute analytical values for the expectation value of the energy $\langle E \rangle$ and absolute magnetization $\langle |M| \rangle$ as well as heat capacity C_v and magnetic susceptibility χ , all using the equations (7) through (12). These are then the values to which we compare the numerical results produced by the model for various choices of the total number of Monte Carlo cycles/sweeps.

	MC cycles	$\langle E \rangle$	$\langle M \rangle$	C_v	χ	Exec time [s]
Numerical	10^2	-1.98000	0.995000	0.1584000	0.00990000	0.00025
Numerical	10^3	-1.98200	0.993500	0.1427040	0.02083100	0.00055
Numerical	10^4	-1.99100	0.997150	0.0716760	0.00806751	0.00240
Numerical	10^5	-1.99558	0.998460	0.0352819	0.00481051	0.03709
Numerical	10^6	-1.99614	0.998701	0.0308204	0.00393026	0.14306
Numerical	10^7	-1.99595	0.998651	0.0323185	0.00404142	1.37764
Numerical	10^8	-1.99596	0.998656	0.0322280	0.00401855	7.65786
Numerical	10^9	-1.99598	0.998660	0.0320946	0.00401503	69.3750
Analytical	N/A	-1.99598	0.998661	0.0320787	0.00401074	N/A

Table 3.1: Comparing numerical and analytical results for $\langle E \rangle$, $\langle |M| \rangle$, C_v and χ in the 2×2 lattice case with temperature $T = 1.0$ J and $\mathcal{J} = k_B = 1$. Also the lattice was initialized with a random spin configuration. Note that all values are "per-spin" (divided by total number of spins (4 in this case)). Numerical results are computed by 4 parallel MPI processes as well as compiled with optimization flags.

One can see that there are fairly good correspondence between the numerical and analytical values for $\langle E \rangle$ and $\langle |M| \rangle$ already at very low number of Monte Carlo cycles for this 2×2 case. However the heat capacity and susceptibility are not satisfactory in agreement with their analytical counterparts this early on. When we reach around 10^6 Monte Carlo cycles, all four quantities seem to be within acceptable limits. Although, what's acceptable here can be very subjective, but at least for 10^6 cycles, all quantities are within 96% of the analytical value. However, if one is only interested in the mean energy and mean magnetization, far fewer cycles are needed to achieve good agreement.

3.3 Parallelization

The computations in this project use a lot of CPU time due to the vast number of Monte Carlo cycles we perform, and thus, emphasis has been put on the speed of the program. In particular, the computationally heavy nature of the project motivates implementation of parallel execution of the code. To achieve this the Message-Passing-Interface (MPI) [1] has been used. The workings of the Monte Carlo simulations in the Ising model makes parallelization quite accessible. The main idea is that divide the total number of Monte Carlo cycles we wish to run between all the MPI process and let each run its own set of Monte Carlo cycles and locally keep track of the computations of the expectation values. Then when we e.g. wish to write out the expectation values to file (or the simulation is done. i.e. each process is done with their Monte Carlo cycles) we sum up all local expectation values from each MPI process and collect the total expectation values at the master process. In practice this operation is done by using the `MPI_Reduce()` function. See next section for performance gains resulting from utilising the discussed parallelization.

3.4 Timing

Here we will briefly discuss the time gains made by two performance enhancements done to the code. One of the being the parallelization discussed in the above subsection and the other being use of optimization compiler flags for the C++ compiler in order to enable features such as vectorization of loops. Standard compiled C++ programs does not have optimization features turned on and creates a full size direct translation of the C++ code into machine code. This means the program will not run as fast as it potentially can with enabled optimization flags for the compiler. Indeed this was experienced during running of the simulations in this project and a speedup of a factor 10 was achieved using optimization flags and vectorization. This saved significant amounts of time. In addition we have the speedup we gain from parallelizing the code as described above. The laptop used for running the simulations described in this report had 4 available CPU cores. Thus the logical choice for number of MPI processes to use, was 4. Observing execution times of a simulation with 10^6 Monte Carlo cycles for a 20×20 lattice gave an average of 25 seconds with 1 process while 4 MPI processes gave an average of 7.5 seconds execution time, yielding a speedup of 3.3. While we would expect to get a speedup closer to 4, this is almost impossible due to the fact that when you add MPI there are also associated startup costs for the processes as well as time used in communication between the processes (although there isn't that much inter-process communication in this project, only one reduce call). There is no point in trying to run on more than 4 processes in this case as then MPI would just spawn virtual more virtual processes per CPU core which would instead slow down the program (due to more overhead) instead of giving additional speedup.

To summarize, combining the speedup from optimization flags and parallelizing with MPI gave a speedup of roughly factor 33 compared to the original non-vectorized serial code. Some additional timing data can be seen in the rightmost column in table 3.1. Here it appears that the execution time scales linearly with number of Monte Carlo cycles, which is intuitively reasonable. However the execution time scales much worse with the lattice size considering the double loop over the lattice.

4 Results

We now move to look at some results from running the Ising model for various lattice sizes and. Key features we will be looking at include equilibration time, i.e. how many Monte Carlo cycles are required before the system reaches its most likely state, state acceptance dependence on temperature, probability distribution for the Ising model energy, signs of phase transitions and closer looks at the critical temperature.

4.1 Reaching the most likely state

Let us look a 20×20 lattice and study the amount of Monte Carlo simulations needed in order to reach the most likely state. We set up a simulation using $L = 20$ spins in each direction and a total of 10^8 Monte Carlo cycles. We perform the computations both for $T = 1.0$ J and for $T = 2.4$ J. We repeat this for both an ordered (all spins pointing up) and random initial spin configuration. We are interested in looking at the development of the mean energy $\langle E \rangle$ and mean magnetization $\langle |M| \rangle$ as functions of number of Monte Carlo cycles. The results from the simulations are visualized in figure 4.1. The plots are zoomed in on to get a better impression of the convergence towards the equilibrium state (the initial values for $\langle E \rangle$ and $\langle |M| \rangle$ are far outside the scope of the y -axis in the plots, but showing these initial values would make everything else look flat).

Firstly, we may look note from figure 4.1 that both the mean energy and mean magnetization move towards what looks like equilibrium states, for both temperatures and for both initial spin configurations. This is promising and reasonable based on the workings of the Metropolis sampling algorithm, but what are the number of cycles needed before the most likely state is reached and what are the factors that impact it? Looking at the graph we may read values of Monte Carlo cycles in the range $[1.1 \cdot 10^6, 5.0 \cdot 10^6]$ for all four panels in the figure. There is, though, significant differences for the initial spin configurations within each of the temperature runs. In particular, for the $T = 1.0$ case, the most likely state is reached a lot quicker (both for $\langle E \rangle$ and $\langle |M| \rangle$) starting with an ordered configuration versus the random configuration. Especially the mean magnetization uses noteworthy more "time". A reason behind this could be related to the fact that when starting in an ordered configuration (all spins up) we have the minimum energy in the system in accordance with equation (1) and as illustrated in table 2.1 for the 2×2 case. And for the lower $T = 1.0$ of the two temperatures, this minimum energy would be closer to the most likely state (lower temperatures having lower energy in the most likely state. This can be seen in equation (7) for the energy expectation value using the Boltzmann distribution. A lower temperature gives a smaller exponent and thus smaller overall expectation value. We also see this on the y -axis in the plots) than for the upper $T = 2.4$. This argument is also help explain why the ordered initial configuration does not appear that beneficial in the $T = 2.4$ case. Here the random one even seem to be slightly better, although both are quite similar and produce about the same equilibration time.

Next we may note that for $T = 2.4$, the system in general uses more time towards the most likely state, there are significantly larger oscillations (see scale of y -axis for the two temperature cases) about the most likely state. This can also be linked to the Boltzmann distribution and the Metropolis algorithm itself: In the Metropolis algorithm we have two options to accept a new configuration; either if the computed energy difference ΔE from the previous configuration is less

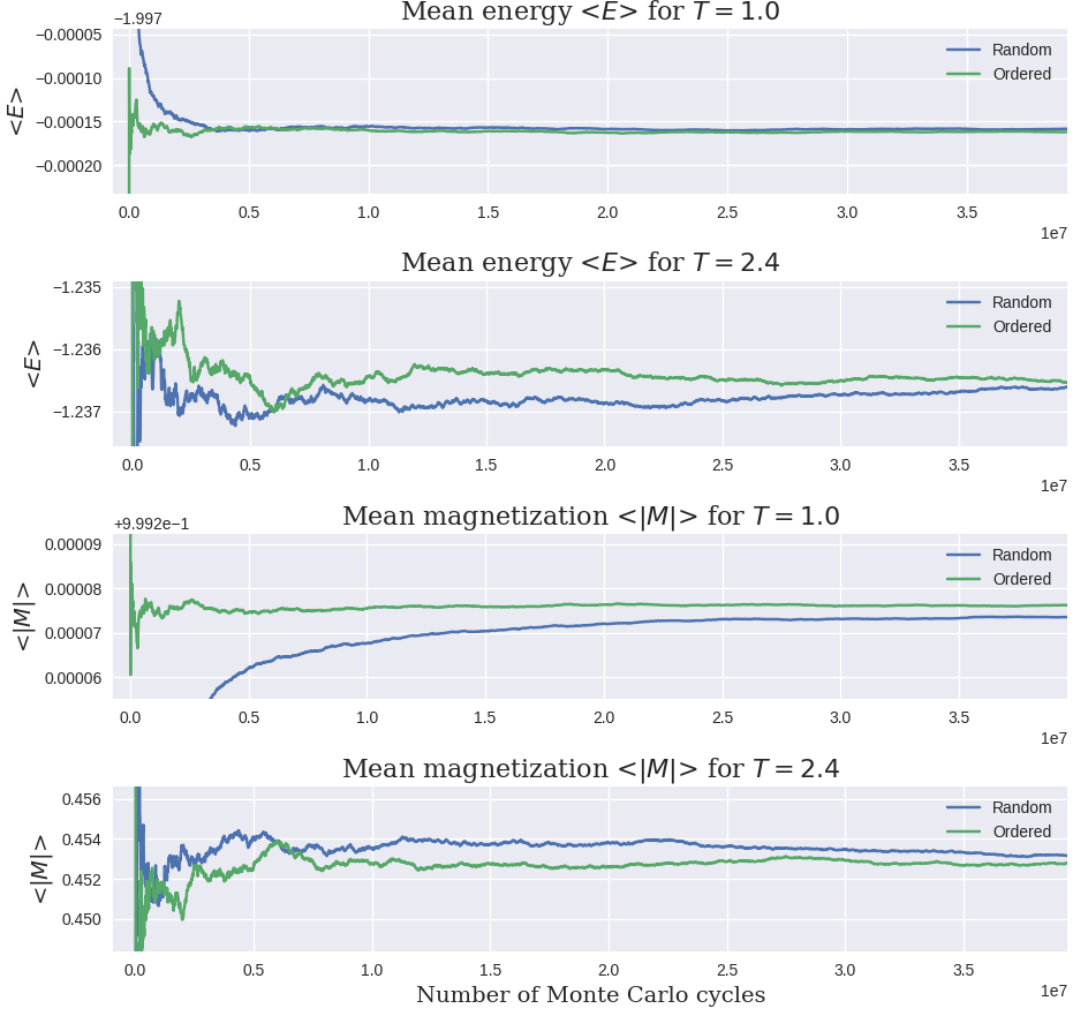


Figure 4.1: Results for the 20×20 lattice for mean energy $\langle E \rangle$ and mean magnetization $\langle |M| \rangle$ as a function of number of Monte Carlo cycles. Simulations are done for both a temperature $T = 1.0$ J and for $T = 2.4$ J and with 10^8 Monte Carlo cycles for each temperature.

than or equal to zero, i.e. $\Delta E \leq 0$, or if a random number $r \in [0, 1]$ satisfies

$$r \leq e^{-\beta \Delta E} = e^{-\Delta E/T} \quad (13)$$

with $\beta = 1/T$ in our case (scaled temperature $k_B T \rightarrow T$). We can then see that if T is larger (as in e.g. 2.4 versus 1.0), we get a larger exponent in (13) and thus a higher chance for r to actually be smaller than $e^{-\Delta E/T}$ which further implies a higher chance for the system to accept the new configuration. In summary, and in other words, this reflects systems of higher temperatures having more "available" configurations/states to "jump" in and out of. This also seem reasonable with physical intuition and seems to be the reason behind the less "ordered" move towards the most likely state for the $T = 2.4$ case.

Another manifestation of this effect can be seen in figure 4.2. Here the 20×20 lattice simulation is run over a range of $T \in [1.0, 2.4]$ with a temperature step $\Delta T = 0.05$ and 10^6 Monte Carlo cycles for each temperature. What's plotted is the total number of accepted configurations (at total Monte Carlo cycles) as a function of temperature. As the title suggest, this corresponds to

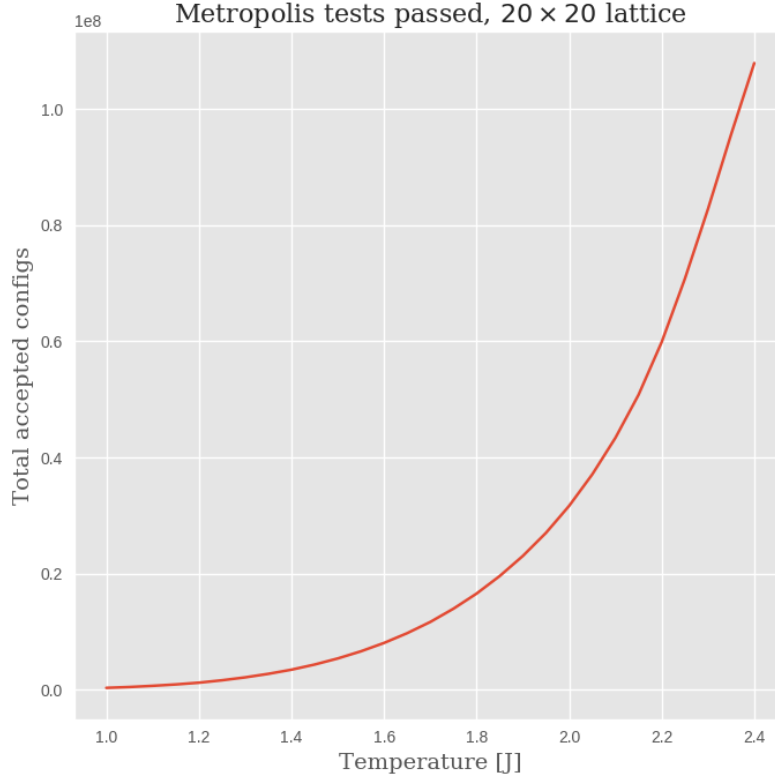


Figure 4.2: Plot showing the total number of accepted states (at total number of Monte Carlo cycles (10^6 for this plot)) as a function of temperature T (with units energy due to scaling). For this figure, a random starting configuration was used for all temperatures.

plotting the number of passed Metropolis tests as a function of temperature. We observe that the number of accepted states exhibits an exponential-like behaviour on T . This brings us back to the above discussed specifics of the Metropolis test and the Boltzmann distribution. As T increases the system becomes exponentially more prone to accept the new suggested configuration. This provides us with a link between figure 4.1 and figure 4.2.

4.2 Probability distribution

In light of the all above talk about most likely states etc., we now move on to looking at the likelihood function for the energy, i.e. the probability distribution $P(E)$ for the energy in the 20×20 case. We are interested in seeing how the probability of the system is distributed (hopefully) around the expectation values discussed in the above section. To compute the distribution we count the occurrences of energies after each sweep through the lattice. However as the energy of the initial spin configuration is generally not very close to the most likely state, the move towards the most likely state would somewhat disturb our computations of the probability distribution. Thus we start counting the energy occurrences after we have reached the most likely state. The chosen value for this is based on the discussion in section 4.1. To be well within limits of when the most likely state is reached, we choose to start the distribution counting after $6 \cdot 10^6$ Monte Carlo cycles. Then we count store the each energy for the next million Monte Carlo cycles and create a histogram of the distribution of the energy when oscillation about the most likely state.

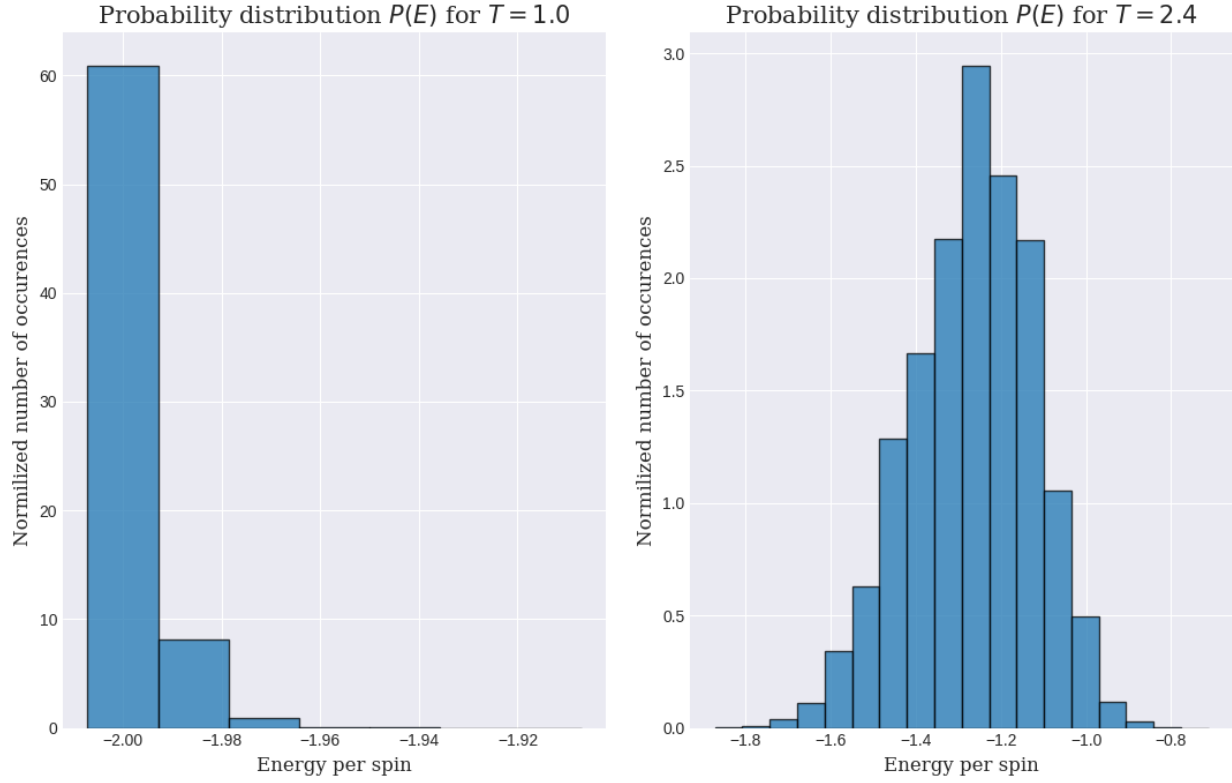


Figure 4.3: Computed energy distribution $P(E)$ for the $20 \times$ lattice at temperatures $T = 1.0$ and $T = 2.4$ by counting number of times various energies occur in the simulation. The counting was initiated after $6 \cdot 10^6$ cycles and 10^6 sequential energy values were counted. A random initial configuration was used in both cases.

This approach is done both for $T = 1.0$ and for $T = 2.4$ and the results can be seen in figure 4.3. Note that this histogram is normalized such that the area under the distributions integrate to 1. First of all we observe that peak of the distribution (for both temperatures) correspond well with the expectation values illustrated in figure 4.1. Next we observe a significant difference in the spread of the distributions. The spread is roughly one order of magnitude larger for the $T = 2.4$ case. This is consistent with the discussion in section 4.1 and again reflecting the larger tendency of accessing more states at higher temperatures in correspondence with the Boltzmann distribution behind the Ising model. Conversely for the lower temperature case, the system rarely jumps out of the ground state (all spins up) and almost all energy counts are for this energy. For further studying the spread, the variance σ_E^2 was calculated for the two simulations. For $T = 1.0$, the variance was 0.027 which by visual inspection of the left panel in figure 4.3 seems to correspond pretty well. However for the $T = 2.4$ simulation, the variance was found to be 8.12 which is a lot more than what the right panel in figure 4.3 suggests. This caused confusion and one might have thought that the reason behind the anomaly was related to the fact that we compute the variance from the very start of the simulation and that the initial move towards the most likely state disturbed the value significantly. This was checked in the form of basing the variance computations only in the same Monte Carlo cycle interval $[6 \cdot 10^6, 7 \cdot 10^7]$ as the distribution counting. This resulted in a variance of 8.06. Although slightly smaller than 8.12, the author concludes that the initial move towards the most likely state did not significantly disturb

the proper value for the variance, and thus, the reason behind the discrepancy remains unclear.

4.3 Phase transitions

Next we wish to see if we can identify a phase transition using the Ising model. In accordance with [2], the Ising model is expected to exhibit second order phase transitions. Our quest in this subsection is to try and identify them in our simulations. Our setup for this exercise is as suggested [3], i.e. we run simulations for lattices of sizes 40×40 , 60×60 and 80×80 . Unfortunately, due to long computation time for the 100×100 lattice, there was not enough time to include the results for this particular case. However we add the 20×20 lattice instead. For each of the lattices we simulate over a temperature range $T \in [2.2, 2.34]$ (presumably close around the critical temperature for phase transitions) with a step of $\Delta T = 0.02$ and use $4 \cdot 10^6$ Monte Carlo cycles (based on the equilibration times discussed in section 4.1. However, this is probably a bit too few cycles for the larger lattices here and the author wishes there were time for simulations with more Monte Carlo cycles) and compute the expectation values for the energy $\langle E \rangle$ and magnetization $\langle |M| \rangle$ as well as the specific heat C_v and magnetic susceptibility χ . Recall that the two latter quantities are calculated with the left formula in equations (11) and (12) after the simulation for each temperature.

In figure 4.4 we can see how the four mentioned quantities evolve as functions of temperature for each of the simulated lattices. In accordance with [2] we have that, at the critical temperature the magnetic susceptibility will diverge. However in our case we have a finite sized lattice and are not in the thermodynamic limit where the lattice size $L \rightarrow \infty$. Thus we rather expect to find a large maxima in the susceptibility. Indeed this is what we see in the bottom panel in figure 4.4. And so it seems like we are able to identify the signs of a phase transition. Also noteworthy is that this effect becomes more prominent for the larger lattices, i.e. the lattices further towards the thermodynamic limit where the diverging susceptibility takes place. In that sense the difference in the graphs for the various lattice sizes seem reasonable.

4.4 Critical temperature

How does the critical temperatures at which the phase transitions seem to occur in 4.4 compare with Lars Onsager exact results from 1944 [4]? In order to answer this we utilize equation (3) in [3], which relates the critical temperature $T_C(L)$ for a finite lattice of size $L \times L$ with the corresponding one for an infinitely large lattice $T_C(L = \infty)$ i.e.

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

where a is a constant and the exponent constant ν that in accordance with [2] and [3] have an exact value of $\nu = 1$. Applying (14) for two lattices with sizes $L_1 \times L_1$ and $L_2 \times L_2$ gives us the opportunity to write

$$\begin{aligned} T_C(L_1) - T_C(L = \infty) &= \frac{a}{L_1} \\ T_C(L_2) - T_C(L = \infty) &= \frac{a}{L_2} \end{aligned}$$

and solving this set of equations for $T_C(L = \infty)$ by eliminating a gives

$$T_C(L = \infty) = \frac{L_2 T_C(L_2) - L_1 T_C(L_1)}{L_2 - L_1} \quad (15)$$

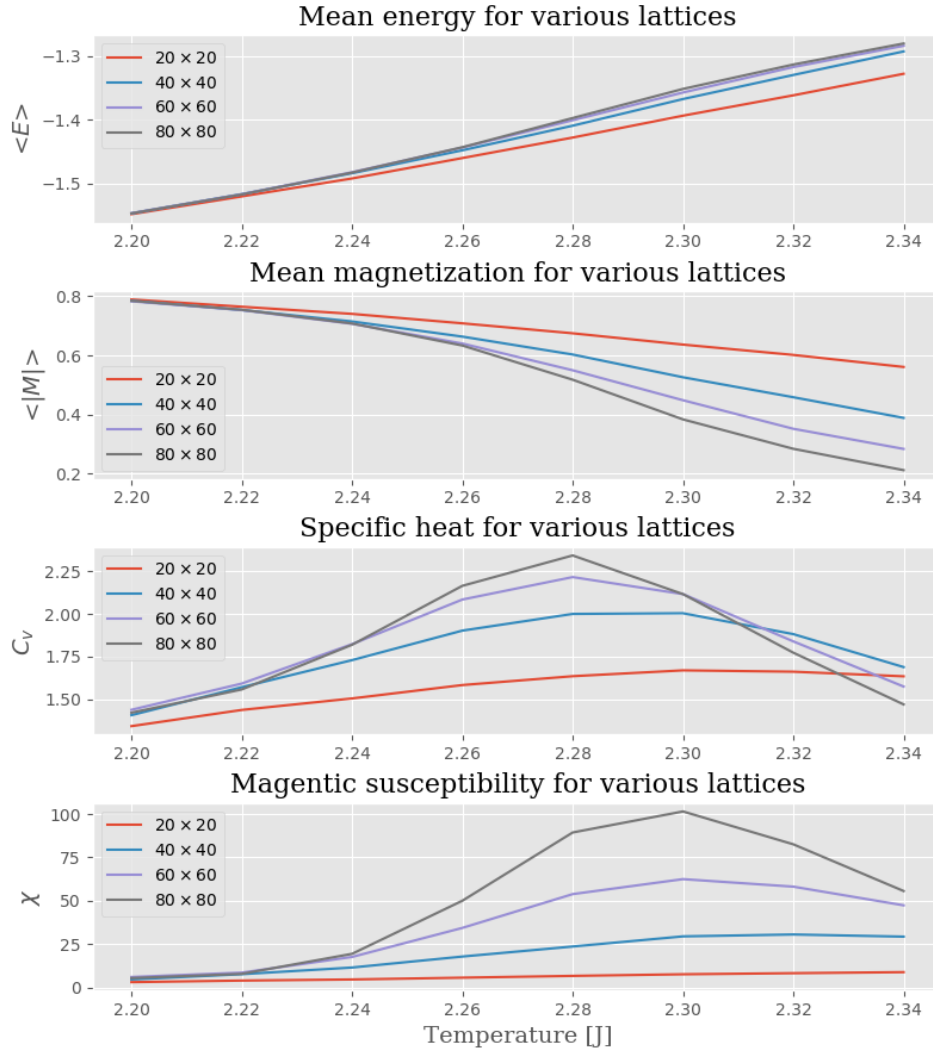


Figure 4.4: Computations of mean energy $\langle E \rangle$ and mean magnetization $\langle |M| \rangle$ as well as the specific heat C_v and magnetic susceptibility χ in simulations for $T \in [2.2, 2.34]$ with a step of $\Delta T = 0.02$ and $4 \cdot 10^6$ Monte Carlo cycles. All repeated for 4 different lattice sizes.

Reading the critical temperature at the susceptibility maxima in figure 4.4 lets us then use (15) to compare our simulation results with the exact value found by Onsager. First we list the values we read for χ maxima in figure 4.4. These values can be seen in table 4.1

Using for instance the critical temperatures for $L_1 = 80$ and $L_2 = 40$ and inserting into (15) gives $T_C(L = \infty) = 2.28$ Joules which corresponds fairly well with Onsager's exact result of $2/\ln(1 + \sqrt{2}) \approx 2.269$ [4]. Similar numbers can be found by using other combinations of L_1 and L_2 from table 4.1. It's worth noting that, as can be seen in table 4.1 and figure 4.4, the critical temperature identified for $L = 80$ is the one that itself is closest to the exact result, further reinforcing the reasoning from section 4.3 that the larger lattices will better approximate the results for the thermodynamic limit $L \rightarrow \infty$. This is also intuitively reasonable.

Areas for improvement in the above calculations include the fairly moderate amount of Monte Carlo cycles used and the resolution in temperature. In order to gain higher accuracy we would have liked to have used even more cycles and a smaller temperature step to properly resolve the

L	20	40	60	80
$T_C(L)$ [J]	2.34	2.32	2.30	2.30

Table 4.1: Values of critical temperature (units energy) read of from the magnetic susceptibility (bottom panel) in figure 4.4.

susceptibility maxima. Doing this we could probably have identified the critical temperature with somewhat higher accuracy. Regardless, phase transitions at reasonable critical temperatures seem to have been identified in the Ising model.

5 Conclusions

In this project we have studied spin interactions in a magnetic material using the binary 2D Ising model. We have implemented the Ising model, and with it, the Metropolis algorithm for sampling the randomly generated data. We have reinforced the value of having available analytical results in order to verify a numerical model during code development. We have seen that vectorization of the C++ code (through the use of optimization flags) and parallelizing the code yields significant performance enhancements. In particular, compared to the non-vectorized and non-parallelized version of the code we can achieve a speedup of almost factor 40, which is very useful when doing computationally tasks like the ones in this project. Furthermore we have observed how expectation values of energy $\langle E \rangle$ and magnetization $\langle |M| \rangle$ of the lattice system in the Ising model behaves as functions of the number of Monte Carlo cycles in the simulation as well as the temperature of the system. Systems with larger temperature are found to be able to access more spin configurations and thus exhibits a more oscillating convergence paths towards their most likely state. This temperature dependence can be traced back to the nature of the underlying physics, i.e. Boltzmann distribution (5). Additionally we have seen reinforced signals of this through analysing the probability functions for systems of various temperatures. Finally we have studied phase transitions through running simulations, for different lattice sizes, around the critical temperature, and observed how the mean energy, mean magnetization, specific heat and magnetic susceptibility behaves as functions of temperature. Signs of a phase transition is visible and is more prominent for the larger lattices closer to the thermodynamic limit. The phase transitions were found to happen at critical temperatures corresponding well with Lars Onsagers exact result [4].

Personal notes:

In yet another project the use of unit tests saved the author potentially several hours of debugging during development of the code for this project. Again an excellent advertisement for embracing the world of unit tests. Also the author found it particularly positive to have a computational project that was sufficiently computationally heavy that implementing parallel code significantly speed up the process of producing results. Hearing about how parallel computing is faster is different from actually having the experience first hand.

References

- [1] Message-passing-interface. <https://computing.llnl.gov/tutorials/mpi/>.
- [2] Morten Hjorth-Jensen. Computational physics. FYS3150/4150 Lecture Notes, 2015.
- [3] Morten Hjorth-Jensen. Project 4. FYS3150/4150 Project Description, 2017.
- [4] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944.