A Numerical Study of Phase Transitions in the Two-Dimensional Ising Model

Caspar William Bruenech

November 19, 2019

Abstract

A numerical implementation of the two dimensional Ising model is presented. The model produced unsatisfactory results for many of the experiments, including increasing errors in the expectation values and magnetic susceptibility for an increasing number of Monte Carlo cycles when compared to the analytical values of a small spin lattice of size 2×2 at $T = 1.0$. The relaxation time of the Ising model was observed for varying temperatures and initial configurations. A phase transition was observed and a critical temperature was extracted, but a too small step size in the temperature resulted in having to fit a polynomial to the produced data. Nevertheless, the resulting critical temperatures produced a good approximation of the critical temperature at the thermodynamic limit (infinite lattice size), with an error of only 0.3%. Parallelization was also implemented and produced a speedup of 3 with 4 cores used to run the program.

1 Introduction

The two-dimensional square-lattice Ising model is one of the simplest statistical models to show a phase transition [1], however the analytical expressions requires the computation of significantly large numbers in order to produce realistic results. In this article we will look at how we can estimate the interesting quantities in the model by using Monte Carlo algorithms combined with Metropolis sampling. We will first make sure the implemented program which simulated the Ising model produces accurate results when compared the analytical solution of a small example of the Ising model, before observing how the model reaches a thermodynamic steady state (equilibrium) after a certain number of iterations. We will also be observing how the Metropolis algorithm samples values which should allow us to estimate expectation values as the mean over many Monte Carlo cycles, before finally studying phase transitions in the Ising model by observing the heat capacity and magnetic susceptibility for varying values of the temperature and number of particles in the model.

2 Method

Note: In this article we utilized natural units such that the Boltzmann constant $k_b = 1$. We have also used a value for $J = 1$ in addition to scaling the temperature T such that T has units $[k_b T/J]$.

2.1 The Ising Model and General Idea

The two-dimensional square Ising model contains L particles in each dimension, whereas each particle has possible spin value of 1 or -1, indicating spin up and spin down. There are two main quantities which we are interested in computing, first of which is the expected energy of the Ising model, which is given as

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

where Z is the so-called partition function and $\beta = \frac{1}{k_B T}$ with k_B as the Boltzmann constant. The partition function is a physical quantity used to describe the statistical properties in a system which is in thermodynamic equilibrium. It is mathematically defined as

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

and where M goes over all possible microstates. The second quantity we will be studying is the mean magnetization $\langle \mathscr{M} \rangle$. However, due to the magnetization oscillating around zero, this can sometimes produce unexpected results. Therefore, we will be utilizing the absolute mean magnetization, which is defined as

$$\langle |\mathscr{M}| \rangle = \frac{1}{Z} \sum_{i}^{M} |\mathscr{M}|_i e^{-\beta E_i}.$$

Due to Using these two quantities, the specific heat capacity and magnetic susceptibility can be computed using

$$C_v = \frac{\sigma_E^2}{kT^2 N}, \qquad \chi = \frac{\sigma_m^2}{kT},$$

where σ_G^2 denotes the variance of the quantity G, such that

$$\sigma_G^2 = \langle G^2 \rangle - \langle G \rangle^2.$$

The specific heat capacity tells us how much energy is required to increase the temperature of a substance per mass, while the susceptibility quantifies how much a material will become magnetized when exposed to a magnetic field. Both of these quantities are related to phase transitions, which we will be studying later.

The problem with computing the mean energy and magnetization for the two dimensional Ising model is the fact that even for a relatively small amount of particles (spins) in each dimensions, L, the number of possible configurations is of a significant magnitude, as the number of microstates goes as 2^{L^2}. For example with only 100 spins in each direction, there are a total of 2^{10000} microstates, or approximately 9.9×10^{30102}. This is a number which is numerically impossible to compute within a realistic time span. As a result of this, other methods for computing the expectations values for the energy and magnetization have to be deployed. One such method, which is the one we will be implementing and analysing in this experiment, is the use of Monte

Carlo methods with Metropolis sampling. In the Ising model without an external magnetic field, the energy in a specific configuration is defined as

$$E_i = -J \sum_{\langle k,l \rangle} s_k s_l, \tag{1}$$

where J is a constant indicating the strength of interaction between neighbour particles, and the sum goes over the nearest neighbours of each spin. Equivalently, the magnetisation for each configuration can be computed with

$$\mathscr{M}_i = \sum_{j}^{L^2} s_j. \tag{2}$$

As mentioned, the partition function is defined for a system in thermal equilibrium. Thermal equilibrium is also defined as the most likely state, which means that when we simulate the Ising model at a given temperate T, the spin lattice (a $L \times L$ matrix with values of -1 or 1) will eventually reach a steady-state, where the energy oscillates around the true expectation value of the energy, $\langle E \rangle$. By then performing M Monte Carlo cycles, the true expectation values $\langle E \rangle$ and $\langle \mathscr{M} \rangle$, can be approximated as

$$\langle \hat{E} \rangle = \frac{1}{M} \sum_{i}^{M} E_i \quad \text{and} \quad \langle \hat{\mathscr{M}} \rangle = \frac{1}{M} \sum_{i}^{M} \mathscr{M}_i,$$

where E_i and $\mathscr{M}_i$ are computed, respectively, from equation 1 and 2.

For the implementation used in this experiment, periodic boundary conditions will be enforced. One way of implementing this numerically is to add so-called ghost points to the lattice, which involves adding the end rows and columns to the opposite side of the lattice. This means that the original spin matrix of size $L \times L$ becomes a lattice of size $(L+2) \times (L+2)$. For an $L = 2$ spin-matrix, the expanded lattice then becomes

$$\begin{vmatrix} -1 & 1 \\ 1 & -1 \end{vmatrix} \rightarrow \begin{vmatrix} 0 & 1 & -1 & 0 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{vmatrix}.$$

As a result, the energy can be computed by looping over the inner rows and columns of the lattice. It should be noted that a $2x2$ matrix is not the best example for implementing ghost points, as all necessary neighbours are already accessible, but the idea is generalized for larger matrices.

While we can not calculate the analytical partition function and therefore the expectation values for larger lattices, we can do it for the smallest examples, such as for $L = 2$. In this case there are a total of 2^{L^2} possible configurations (microstates), and the energy for any configuration the energy is calculated as

$$E_i = -J \left[s_{1,1}(s_{1,2} + s_{2,1}) + s_{1,2}(s_{1,1} + s_{2,2}) + s_{2,1}(s_{1,1} + s_{2,2}) + s_{2,2}(s_{2,1} + s_{1,2}) \right].$$

Note that not all four neighbours of each spin is accounted for. This is because we do not want to count each pair of neighbouring spins twice when summing to obtain the total energy. Another

No. spins up	Degeneracy	Energy	Magnetization
4	1	-8J	4
3	4	0	2
2	4	0	0
2	2	8J	0
1	4	0	-2
0	1	-8J	-4

Table 1: All possible configurations for a two dimensional Ising model with $L = 2$ and periodic boundary conditions.

way of dealing with this is to include the term $\frac{1}{2}$ when computing the sum, and including all neighbours. Similarly, the magnetization for a given configuration is computed;

$$\mathscr{M}_i = \sum_j^{16} s_j.$$

For example, with all spins up, we get an energy of $E = -8J$ and $\mathscr{M} = 4$. There is only one possible ordering of the spins to produce this state, and it therefore has a degeneracy of 1. All other possible configurations with a given number of spins up and consequently the degeneracy, energy and magnetization is given in table 1. [2]

Using these values, we can then calculate the analytical values for the partition function, expectations values for both the energy and magnetization, and subsequently the specific heat capacity and magnetic susceptibility. The partition function is then given as

$$Z = \sum_i^{16} e^{\beta E_i} = e^{8J\beta} + 4e^0 + 4^0 + 2e^{-8J\beta} + 4e^0 + e^{8J\beta} = 12 + 2e^{8J\beta} + 2^{-8J\beta},$$

which gives us the expected energy as

$$\langle E \rangle = \frac{1}{Z} \sum_{i=1}^{16} E_i e^{-\beta E_i} = \frac{1}{Z} \left(-8Je^{8J\beta} + 0 + 0 + 2 \times 8Je^{-8J\beta} + 0 + (-8J)e^{8J\beta} \right)$$

$$= \frac{16Je^{-8J\beta} - 16e^{8J\beta}}{Z}$$

Which also gives us the expectation value for the energy squared as

$$\langle E^2 \rangle = \frac{128Je^{8J\beta} + 128e^{-8J\beta}}{Z}$$

Similarly, we can compute the expectation values for $|\mathscr{M}|$ and $|\mathscr{M}|^2$

$$\langle |\mathscr{M}| \rangle = \frac{8(2 + e^{8J\beta})}{Z},$$

$$\langle \mathscr{M}^2 \rangle = \frac{8(2 + e^{8J\beta})}{Z}.$$

This quantities will later be calculated and used as benchmarks for the program which performs the Monte Carlo simulation.

2.2 Monte Carlo algorithm with Metropolis Sampling

The algorithm for the Monte Carlo simulation of the two dimensional Ising model with Metropolis sampling goes as follows

1. Initialize the spin lattice by randomly generating values of either -1 and 1 for each lattice index, and compute the energy and magnetization of the initial configuration
2. Enter the Monte Carlo loop
3. Loop through all spins in the lattice, and for each iteration
   (a) Flip one random spin in the lattice
   (b) Compute the change in energy $\Delta E = E_1 - E_0$ where E_1 is the energy in the new configuration
   (c) Accept the new configuration if $\Delta E \leq 0$
   (d) If not, compute $w = e^{-\beta \Delta E}$
   (e) Generate a random number $r \in [0, 1]$. If $r \leq w$, accept new configuration and keep old configuration otherwise.
4. Update expectation values by summing E, E^2, M, M^2 and $|M|$ with their previous values.
5. Repeat for all M Monte Carlo cycles.
6. Finally, the estimated expectation values are computed by dividing with the total number of Monte Carlo cycles m, and the total number of spins L^2 to get the values per spin.

The quantity Δ is here computed as $e^{-\beta(E_1 - E_0)}$ and corresponds to the probability ratio between the two energy states E_1 and E_0. Since $P(E_i) = \frac{1}{Z} e^{\beta E_i}$, we get that

$$\frac{P(E_1)}{P(E_0)} = \frac{\frac{e^{\beta E_1}}{Z}}{\frac{e^{\beta E_0}}{Z}} = e^{-\beta(E_1 - E_0)}.$$

As we want to move towards an energy minimum, the new configuration is always accepted if the energy is smaller in the new configuration than in the previous, i.e., if < 0. However, if the new energy is larger, we only accept the new configuration based on whether the probability ratio is larger than some random value $r \in [0, 1]$, thus we are more like to reject the new configuration the larger the drop in probability. In this way, we gradually move towards a higher density area even if the new configuration is not a step in the right direction, since we minimize the length of the step in the wrong direction with the accept/reject statement.

For a two-dimensional lattice with periodic boundary conditions, it can be shown that ΔE only has five potential values [2]. This means that instead of computing the exponential value for

each lattice sweep, we can precalculate these values in an array and simply look them up when necessary. The five possible values for ΔE are $-8, -4, 0, 4$ and -8, which means that we can precalculate $e^{-\beta \Delta E}$ and look up this value. This should give a significant performance boost, as we would otherwise have to perform a computation of the exponential, an expensive calculation, $M \times L^2$ times.

After the M Monte Carlo cycles have completed, the various estimated expectation values are computed as

$$\langle \hat{E} \rangle = \frac{E}{ML^2}, \quad \langle \hat{E^2} \rangle = \frac{E^2}{ML^2}, \quad \langle \hat{\mathscr{M}} \rangle = \frac{\mathscr{M}}{ML^2}, \quad \langle \hat{\mathscr{M}^2} \rangle = \frac{\mathscr{M}^2}{ML^2}, \quad \langle |\hat{\mathscr{M}}| \rangle = \frac{|\mathscr{M}|}{ML^2},$$

where E, E^2, $\mathscr{M}$, $\mathscr{M}^2$ and $|\mathscr{M}|$ are the summed energy, magnetization and mean magnetization for each Monte Carlo cycle. Using these values, we can then compute the variance for each expectation value, and thus the specific heat capacity C_v and χ per spin as

$$C_v = \frac{\langle \hat{E} \rangle - \langle \hat{E^2} \rangle}{kT^2L^2} \qquad \chi = \frac{\langle \hat{\mathscr{M}} \rangle - \langle \hat{\mathscr{M}^2} \rangle}{kTL^2}.$$

2.3 Approach

Note that in all the experiments we have set $J = 1$, and additionally scaled the

As we are approximating the expectation values defined by the partition function, we need to make sure that the system is in a steady-state when computing the expectation values. To do this, we will choose a lattice with size $20x20$, i.e., where $L = 20$, and plot the energy and absolute magnetization as a function of time (MC cycles). While it is possible to choose a numerical cutoff for when we agree that the system has reached equilibrium, such as when the amplitude of the oscillations do not vary more than 10%, it is also possible to simply observe the resulting plot and choosing a rough point in time where the oscillations are visually stable. We will perform this experiment for $T = 1.0$ and $T = 2.4$, where, for both temperatures, we will start with an ordered initial lattice (where all spins= 1,), and a random configuration to observe how this affects the equilibration time for both temperatures. Additionally, the number of accepted configurations will be counted as a function of time, to further visualize how the system approaches steady-state.

We will then analyse the probability distribution of the energy for the same system by counting the number of times an energy appears in the computation. In this case we want to see how the energy probability is distributed around the expectations value when the system is in a steady state, and we will therefore begin the counting only after the system has reached equilibrium.

After analysing the implemented model, we can begin our study of phase transitions. To do this, we will study lattices of size 40, 60, 80, and 100, with a temperature $T \in [2.0, 2.3]$. To be sure we get accurate results, the simulation will be run for as many Monte Carlo cycles as possible within the provided hardware and time frame. As a result, this will more than likely be a lengthy computation, and we will therefore be parallelizing the calculations. While it is very possible to parallelize the Monte Carlo simulation code in itself, in this case we can simply distribute the temperature calculations between the available workers. The general algorithm is then a double loop going over the temperatures and lattice sizes, such that for each temperature, M Monte Carlo cycles are run for each lattice size, and subsequently each relevant quantity is computed. With n workers, this computation can then be performed such that for each iteration, T_n/n temperature values are computed each time, where T_n is the total number of temperature

values. The Monte Carlo simulations will also be timed and compared with a serial run of a selection of the lattice sizes in order to estimate the speedup gained from the parallelization. To compute the total time taken to run Monte Carlo simulations for all temperatures for one lattice size, we need to sum up the maximum time taken by the workers for the n-first temperatures, plus the maximum time taken by the workers for the next $2n$ temperatures, and so on until the final $T_n - n$ temperatures. This can then be compared to the sum of the Monte Carlo simulation times for all temperature for each lattice with the serial run. In the Julia programming language, parallelization involves applying the @*distributed*-macro to the loop we want to parallelize, which distributes the iterations between the workers. We also need to apply the @*sync*-macro, such that none of the workers continue before the others have finished. Finally, we create arrays of type *SharedArrays*, which are shared between all workers, such that we can collect the resulting data in one array which we can save for later analysis.

Due to the fact that the mean magnetization $\langle \mathscr{M} \rangle$ oscillates around zero, we will be utilizing the mean value when computing the magnetic susceptibility. This will give us the wrong values for χ, but as we are only looking for the maximum value, where the phase transitions occurs, it is not problematic to use it in this context. Once we have computed the different quantities for each temperature and for each lattice, we can extract the temperature at which the maximum value of C_v and χ is reached. This should ideally be the same, and indicated the critical temperature for this specific lattice size. Using Lars Onsager's [3] analytical solution to the two-dimensional Ising model, it can be shown that the critical temperature of a L-sized lattice, $T_C(L)$, can be related to the critical temperature of an infinitely sized lattice, $T_C(L = \infty)$ via

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

where ν is defined from the equation $\chi(T) \sim |T_C - T|^{7/4}$ and has an exact value of 1. We observe that this equation is on a linear form $y = ax + b$ where $a = a$, $x = L^{-\nu}$ and $b = T_C(L)$. This means that the critical temperature at $L = \infty$ is equal to $y(x)$, which again means that by using the estimated critical temperatures for each lattice size, we can perform a linear regression to produce a continuous model. The intercept of this model will then be the estimation for the critical temperature at an infinite lattice size.

.

3 Implementation & Results

The analytical values for $\langle E \rangle$, $\langle |\mathscr{M}| \rangle$, C_v and χ was computed for $T = 1.0$. These values were then compared to the estimated values from the Monte Carlo simulations for an increasing number of cycles. The results are shown in table ??

For studying the equilibration time, a range of Monte Carlo cycles from 500 to 5×10^4 with a sample distance of 50 was used. The energy and mean magnetization for each temperature, and with both an ordered and unordered initial state was plotted as a function of MC cycles. The results are shown in figure 1.

For producing the probability distribution of the energy, the simulation was run for 10^6 Monte Carlo cycles, where the new energy was appended to an array for each lattice sweep. This was performed for $T = 1.0$ and $T = 2.4$ with a 20×20 lattice. The resulting histograms were then normalized to produce a probability distribution and plotted for both temperatures. This is displayed in figure 2. The plots also include a red line which indicates the computed expectation

log10 N	$\langle E \rangle$	C_v	$\langle \lvert \mathscr{M} \rvert \rangle$	χ
2	-1.96	0.1536	0.98	0.038
4	-1.997	0.023964	0.99925	0.00149775
6	-1.99721	0.0225368	0.999297	0.00144002
8	-1.99724	0.022356	0.999304	0.00143844
Analytical	**-1.99598**	**0.0320823**	**0.998661**	**0.00401074**

Table 2: Analytical values of the various relevant quantities compared to the estimated values with varying number of Monte Carlo cycles.

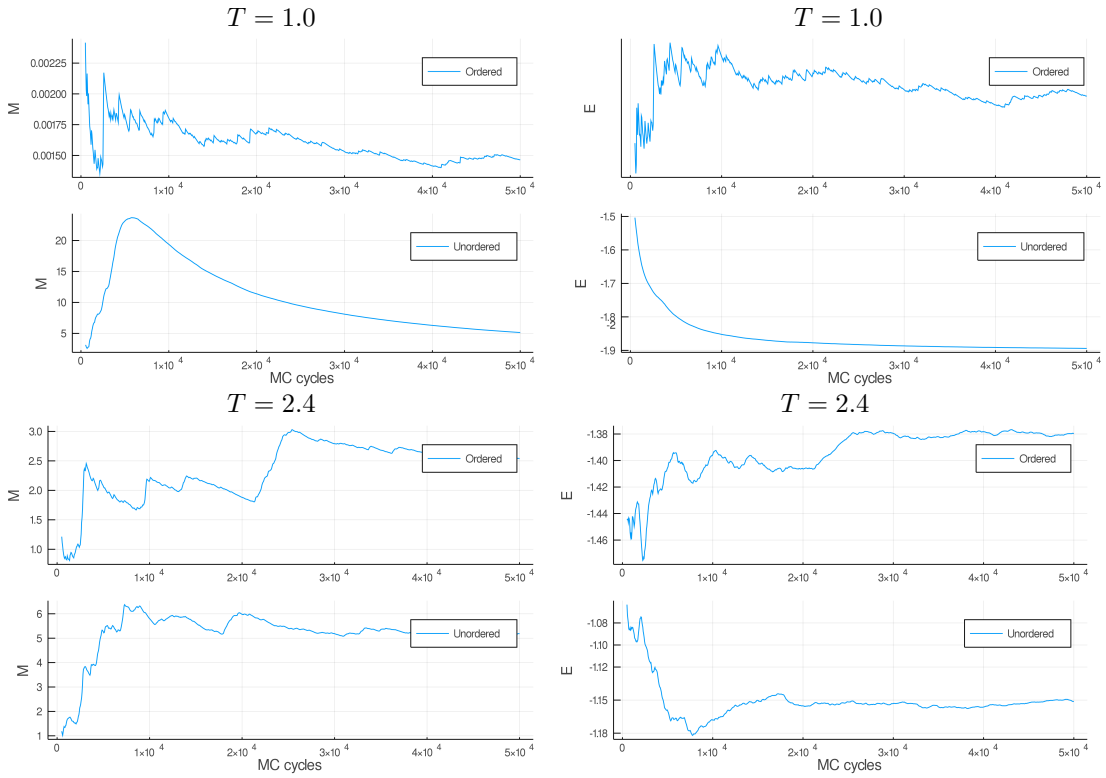

Figure 1: Energy and magnetization as function of time (Monte Carlo cycles) for two different temperatures, and with an initial configuration of ordered spins (all spins = 1) and random configuration.

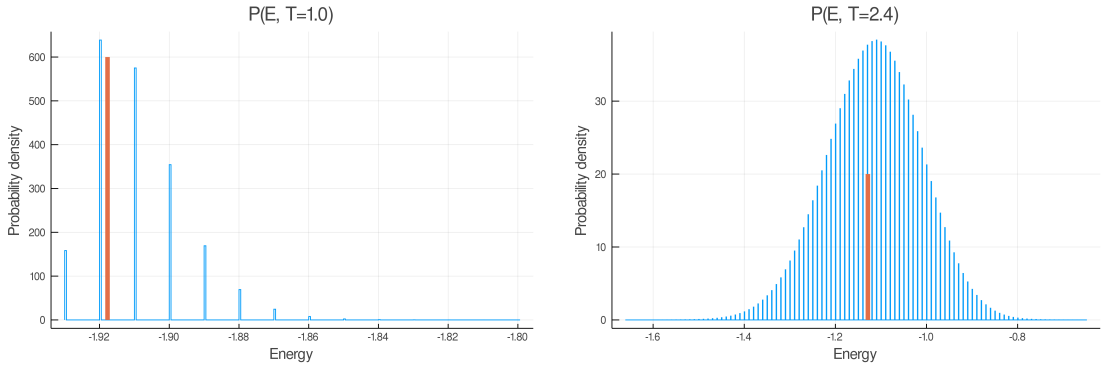

Figure 2: Histogram of energy counts with 10^6 Monte Carlo cycles for $T = 1.0$ and $T = 2.4$. The red line shows the computed expectation value of the energy following all cycles.

L	40	60	80	100
Serial time [s]	3996	5703	N/A	N/A
Parallel time [s]	1393	2019	5636	6927
Speedup	3.0	3.0	N/A	N/A

Table 3: Speedup gained from parallelization with 10^8 Monte Carlo cycles

value of the energy.

For the phase transition experiments, a temperature range of $T \in [2.0, 2.3]$ with a step size of $\Delta T = 0.05$ was used. For each temperature value, the mean energy, mean magnetization, heat capacity and magnetic susceptibility was computed for each lattice size. A total of 10^8 Monte Carlo cycles were used for each case. The result is shown in figure 3. A table displaying the CPU time taken for each lattice size, including a comparison with serial CPU time for $L = 40$ and $L = 60$ is shown in table 3.

Upon observing that the temperature step size was not small enough to differentiate the peaks of the heat capacity and magnetic susceptibility, a polynomial was fitted to the estimated values in order to estimate the different critical temperatures for each lattice size. An example of the fitted polynomial is shown in figure 4.

Using the fitted values for the heat capacity and susceptibility for each lattice size, different critical temperatures were successfully produced. The resulting critical temperatures, however, were different depending on whether we fitted to the C_v or χ-data. Using equation 3, a linear fit was made with both critical temperature estimates which produced intercepts $\beta_{C_v} = 2.275$ and $\beta_\chi = 2.305$. This is shown in figure 5.

4 Analysis

The results of the estimated expectations values, heat capacity and susceptibility for the $2x2$ lattice at $T = 1.0$ (table **??**) are not satisfactory. While we would expect the values to get closer to the analytical solutions for larger amounts of Monte Carlo cycles, we actually observe that for all the quantities apart from $\langle|\mathscr{M}|\rangle$, the values actually diverge more from the analytical values for higher N. This is an unfortunate result whose origin was not discovered. Because of this, we

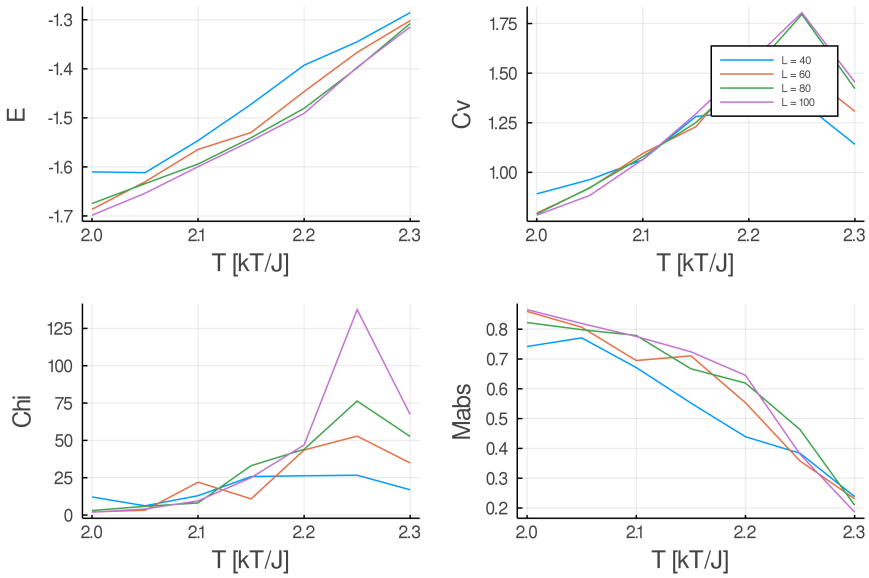

Figure 3: Estimated mean energy, mean absolute magnetization, heat capacity and magnetic susceptibility for varying lattice sizes and increasing temperatures. All values were computed with 10^7 Monte Carlo cycles.

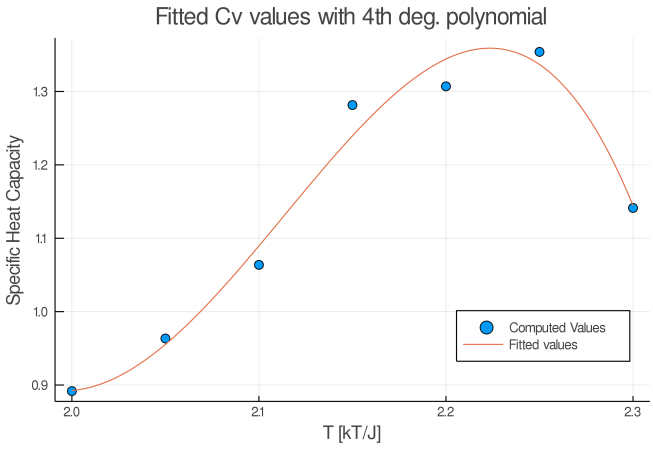

Figure 4: The fitted polynomial to the produced heat capacity data.

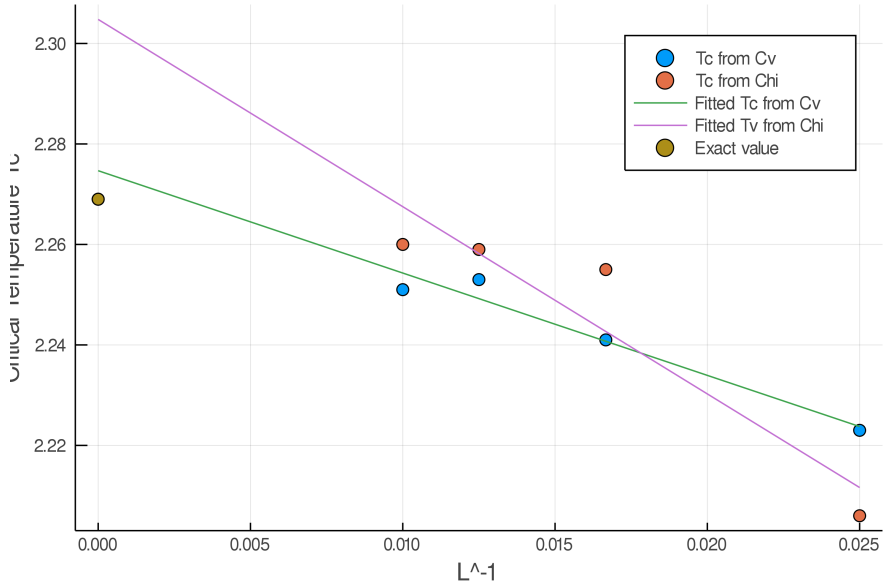

Figure 5: Linear fit of the critical temperatures estimated from the maximum value of the heat capacity C_v and magnetic susceptibility χ with the goal of producing a model with an intercept equal to the exact value.

can not accurately trust the results produced for all the following experiments. Nevertheless, we will still analyze the results that were produced, as they still provide the possibility of studying the Ising model and the phase transitions.

For the plots of the equilibration times, we observe that for $T = 1$, the mean energy and magnetization of the system with an ordered initial configuration reaches equilibrium after approximately 2×10^4 cycles, while for the unordered initial configuration some strange behaviour is observed. The magnetization increases to a large value to around 5×10^3 cycles, before slowly decreasing again. While the slope does seem to stabilize around 5×10^4 cycles, it does not stabilize to the same value as the magnetization with the ordered initial configuration. The reason for this is unknown. For the unordered energy at $T = 1.0$, we see that the value moves to an energy minimum and stabilizes around 2×10^4. We observe that the fluctuations for the ordered configurations at $T = 1.0$ fluctuate around very small values. This is because at $T = 1.0$, the expected values are very close to the initial mean energy and magnetization when all spins are equal to 1.0. For $T = 2.4$, both quantities with an ordered initial state equilibrates slightly before 3×10^4 cycles, while the unordered configurations reaches a steady state already at around 1×10^4. This is the behaviour to be expected, as for a high temperature, the energy and magnetization computed with an ordered initial configuration is further from the true mean value, which means that more cycles are required to "correct" this large error. With a random configuration, the energy and magnetization is more likely to be closer to the true expectation values. However, we observe that both the quantities at $T = 2.4$ equilibrates around different values depending on their initial configuration. As the expectation value is the same whether or not the system was initialized randomly or ordered, this is the indication of an error in the model.

The produced probability distributions for the energy, as displayed in figure 2, shows that for the lower energy, there are fewer possible energy states that the system can inhabit. This

results in a probability distribution which is similar to an exponential probability distribution, with the higher probability density close to the expectation value. For $T = 2.4$, we get a normal distribution of probability density centered on the expectation value of the energy at $T = 2.4$. These plots show the Monte Carlo simulation with Metropolis sampling produces energies distributed in such a way that the mean of the values is equal to the analytical value we want to approximate.

For the phase transition experiment, we produced results that successfully displayed a phase transition in that both the magnetic susceptibility and heat capacity reached a peak at the critical temperature before decreasing again. However, too few temperature grid points were used, such that all the critical temperatures landed on the same value for all lattice sizes. As a result, a polynomial fit had to be implemented (fig. 4). This did give us different values for the critical temperature for each of the lattice sizes, but it does of course add an extra source of error in the final result. This also produced different critical temperatures for the same lattice sizes for the heat capacity and susceptibility. As a result, both critical temperature estimates were fitted with a linear polynomial, as seen in figure 5. We see that the line fitted with the heat capacity values is significantly more accurate the line fitted on the χ-data, with an intercept closer to the true value of $T(L = \infty)$. This produced an estimated $T(\infty)$ as 2.275 with an error of 0.006 or $\approx 0.27\%$. Considering the fact that the implemented model did not even pass the initial benchmark test of the analytical values for the 2×2 lattice at $T = 1.0$, this relatively accurate result does possibly imply a stroke of luck. One way to verify this is to include more temperature values when looking for the critical temperature for the different lattice sizes. By choosing more temperature values we would have been able to produce a more accurate estimate of the critical temperatures as we would not have to fit a polynomial and thereby removing a source of error. However, it is also possible that we did not simply Get Lucky©[1], but that the errors we produced with the first experiments were not relevant for the simulation of the phase transition. Since a majority of the errors revolved around producing the wrong values for the expectation values, this is entirely possible as the study of the phase transitions only looking for how the expectation values changed when the temperature exceeded the critical temperature.

For the speedup of the parallel implementation (table 3), we see that we achieved a speedup of only 3 while the program was run on 4 cores. The reason for this is that the maximum speedup with the parallel implementation used for this problem is limited by the number of grid points in the temperature values. If we had had the same number of temperature grid points as number of workers (cores), we would have seen an ideal speedup equal to this number as we would have been able to compute all the temperature values at the same time. If we had parallelized the Monte Carlo simulation itself, the maximum theoretical speedup would have been the number of cores, as each worker would have performed N/n iterations, where N is the total number of Monte Carlo cycles and n is the number of workers. However, this would also have included certain drawbacks, as it would have required a higher complexity code, which would have taken more time to implement, and it would also have involved more overhead and data transfer, including reductions, which would have slowed down the run time. With the implementation used in this article, the overhead is minimum, as each workers initializes its own environment of the Monte Carlo integration and the required variables that follows in its own memory. The only data transfer required is that of the collecting of the results and CPU times, which is carried out by the master processor and only performed at the end of each Monte Carlo cycle. As a result, a speedup of 3 is not necessarily a bad result even when using 4 cores, as a speedup equal to the number of cores is often hard to achieve even with such parallelizable algorithms as the Monte Carlo.

[1] Copyright Daft Punk 2012

5 Conclusion

The results produced in this article were generally unfavourable, with errors in the expectation values starting at already the benchmark test. However, we were still able to observe interesting characteristics of both the Monte Carlo algorithm with Metropolis sampling, and the Ising model itself. More specifically, we observed how the Metropolis sampling produces a sample distribution which is normally distributed with a mean close to the value we want to approximate. Additionally, the relaxation time for the Ising model was observed for different temperatures and initial configurations, visualizing how a thermodynamic system always equilibrates to the state with the highest multiplicity, which is also the most likely state. Additionally, we successfully observed a phase transition by computing the estimated expectation value for the heat capacity and magnetic susceptibility for increasingly higher temperatures. However, to few temperature values forced us to perform a polynomial fit in order to extract different critical temperature estimates for each lattice size. The resulting critical temperatures were then used to estimate the critical temperature at the thermodynamic limit where $L = \infty$, which produced a surprisingly accurate result with an error of only 0.3%. Finally, a speedup of 3 was gained from parallelization of the temperature loop in the study of the phase transitions. This was smaller than the potential maximum speedup of 4 when parallelizing the Monte Carlo simulation itself, but this method did show to carry some strengths in that came with significantly smaller overhead and data transfer than the alternative implementation.

References

[1] Giovanni Gallavotti and Giovanni Gallavotti. "Classical Statistical Mechanics". In: *Statistical Mechanics*. 1999. DOI: 10.1007/978-3-662-03952-6_1.

[2] Morten Hjorth-Jensen. *Computational Physics Lectures: Statistical physics and the Ising Model*. Aug. 2017. URL: http://compphysics.github.io/ComputationalPhysics/doc/pub/statphys/html/statphys.html.

[3] Lars Onsager. "Crystal statistics. I. A two-dimensional model with an order-disorder transition". In: *Physical Review* (1944). ISSN: 0031899X. DOI: 10.1103/PhysRev.65.117.