The Ising Model

Halvard Sutterud
(Dated: November 2017)

The Ising model in two dimensions is used to simulate phase transitions in magnetic materials with a parallelized Metropolis algorithm. We first benchmark the method with the 2×2 lattice and reproduce the analytical expectation values. The magnetization is shown to be more unstable with fewer Monte Carlo cycles than the energy, which converges rapidly. For $L \geq 40$, we simulate over a range of temperatures and lattice sizes and extract a critical temperature of $2.2694 k_B/J$, which is close to the analytical value of Onsager [2] at 2.2692. The correspondance with the analytical calculations confirms the power law behaviour of the expectation values and correlation length, as predicted from [1].

I. ACKNOWLEDGEMENTS

I want to thank Ragnar Bruvoll for cooperation on writing the code for this project. Our lecturer Morten Hjort Jensen gets a special thanks for his great lectures at 08:00 two times a week, and for always being ready to help or talk. I also want to thank the group teachers for being accessible throughout long group sessions, pushing us forward in the right direction. Me and Ragnars results would not have been as great if it for example had not been suggested that we use 8 of the lab computers at night for running simulations, providing roughly 25 times the computer power of my laptop. In addition, talks with other students have also been great and helpful, they are a nice bunch.

–Halvard

II. INTRODUCTION

Calculating analytical expressions for the expectation values of physical parameters like the energy and magnetization of thermodynamic systems is often an unfeasible task. In a typical thermodynamic system, considering that the possible states goes as states per particle to the power of number of particles, the required calculations for even simple systems easily get too big to even start thinking about. Even in our case, studyipng the Ising model in two dimension where each element only has two possible values, the number of configurations quickly gets absurdly large.

In this project we will study a method for tackling this problem. The Metropolis algorithm is a widely popular Monte Carlo based algorithm for sampling the probability distribution of complex system. Using this distribution, we can then calculate variance of energy and magnetisation, which again are related to thermodynamic quantities of heat capacity and susceptibility.

An implementation in $c{+}{+}$ is applied to the two dimensional Ising model with periodic boundary conditions. In the first part we are going to benchmark it against analytical values of a 2×2 grid, before moving on to studying when the 20×20 grid reaches its equilibrium state. We will then study the distribution of energy states during a simulation. Finally we want to find a critical temperature for a phase transition, which we will do by assuming that the relevant expectation values follow power laws around the critical temperature.

III. THEORY

The Ising Model is a binary system, where the objects at each lattice site can take on two values, either spin up or spin down. The energy of the Ising Model can generally be expressed as

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

where $s_k = \pm 1$ is the orientation of the spin at a lattice point, and the sum is to be taken only over the neighboring points.

A. Canonical ensemble

We will be working with the canonical ensemble, with energy as an extensive variable following an expectation value. To calculate the energy, we need a probability distribution of states. This is given by the Boltzman distribution

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

where $\beta = 1/k_B T$ is inverse of temperature multiplied with Boltzmans constant, and Z is the partition function for the canonical ensemble given by

$$Z = \sum_i e^{-\beta E_i}. \tag{3}$$

where the sum is over all microstates. The expectation value of the energy can then be calculated from the probability distribution with

$$\langle E \rangle = \frac{1}{Z} \sum_i E_i e^{-\beta E_i}. \tag{4}$$

Similarly one can also find the mean magnetization through

$$\langle M \rangle = \frac{1}{Z} \sum_i M_i e^{-\beta E_i}, \tag{5}$$

As both of these involve the partition function, which even for a lattice size of 100×100 involves a sum over 2^{10^4} elements, we need another way of calculating it. Other interesting quantities involve heat capacity C_V and susceptibility χ, calculated through the variance of energy and magnetization respectively. The former is given by

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

and the latter by

$$\chi = \frac{1}{k_B T} \left(\langle M \rangle^2 - \langle M^2 \rangle \right). \tag{7}$$

B. Units

The temperature is taken to be in units of k_B/J, where k_B is Boltzmans constant, and J is the potential energy from magnetic interaction between two spins. The lengths between the lattice elements is assumed to be baked into the dependence of magnetization for the energy.

C. Analytical values

The 2×2 lattice gives us $2^4 = 16$ configurations, as shown in table I. With these sizes, it is possible to calculate analytically the expectation values with the partition function, as it only contains 16 elements (most of which are zero). Using the tabulated macroscopical values of the various microstates, the partition function is given by

$$Z = \sum_i e^{-\beta T} \tag{8}$$

$$= 2 \cdot e^{8/T} + 2 \cdot e^{-8/T} + 12 \cdot e^{0/T} \tag{9}$$

$$= 12 + 4\cosh(8/T). \tag{10}$$

We then calculate the energy expectation value using eq. (4):

$$\langle E \rangle = \frac{2 \cdot (-8J) e^{8/T} + 2 \cdot 8J e^{-8/T}}{Z}$$

$$= \frac{8\sinh(8/T)}{1 + 3\cosh(8/T)} \tag{11}$$

Continuing in the same fashion gives the other needed expectation values,

$$\langle M \rangle = 0 \tag{12}$$

$$\langle E^2 \rangle = \frac{64\cosh(8/T)}{1 + 3\cosh(8/T)} \tag{13}$$

$$\langle M^2 \rangle = \frac{8 + 8e^{8/T}}{1 + 3\cosh(8/T)}. \tag{14}$$

D. Metropolis Algorithm

The Metropolis algorithm is based on Markov Chains, where a system evolves from one state

to another by transition probabilities. The probability for transition from one state can be deduced from the principle of detailed balance,

$$W_{i\to j}w_i = W_{j\to i}w_j \tag{15}$$

where w_i is the probability distribution and $W(i \to j) \equiv W_{ij}$ is a transition matrix. We can write the transition matrix as a combination of the probability of trying a transition to another state $T(i \to j)$ and the probability of this transition being accepted $A(i \to j)$. From the lecture notes, we know that the condition of detailed balance states

$$w_j(t)\frac{T_{j\to i}}{T_{j\to i}} = w_i(t)\frac{T_{i\to j}}{T_{i\to j}} \tag{16}$$

This condition and the condition that the system is ergodic (every state is accesible from any other state) are the two conditions of the Metropolis algorithm. Even though the partition function is uncalculatable, by rearranging the terms in eq. (16) and assuming the transition probability T_{ij} is symmetric, we can get the acceptance probability relation

$$\frac{A_{j\to i}}{A_{i\to j}} = \frac{T_{i\to j}w_i}{T_{j\to i}w_j} = e^{-\beta(E_i - E_j)} \tag{17}$$

Where the partition function in w_i and w_j is cancelled.

Consider then that we want a maximum for the acceptance probability at 1. Assuming $E_i < E_j$, we then set $A_{j\to i} = 1$, and get the other acceptance probability at

$$A_{i\to j} = e^{-\beta(E_j - E_i)}. \tag{18}$$

1. *Energy transitions*

When dealing with an Ising model in two dimension, there is only a handfull of possible different changes in energy ΔE for flipping a given spin. Every element has four neighbors, and it is the orientation of their spins that determine the changes in energy. This gives five possible values of ΔE, by how many neighboring spins there are in the same direction:

Spins same dir.	E_{before}	E_{after}	ΔE
4	-4	4	8
3	-2	2	4
2	0	0	0
1	2	-2	-4
0	4	-4	-8

These changes in energy can then be used to precalculate the possible transition rates for a specified temperature using eq. (18).

TABLE I: The microstates with corresponding macrostates sorted by number of spins in the up direction for the 2×2 Ising model, with the corresponding energy E, magnetization M and microstate configurations. We represent spin up by 1 and spin down by 0.

Spins up	Degeneracy	E	M	Microstates
4	1	-8J	4	$\begin{pmatrix}1&1\\1&1\end{pmatrix}$
3	4	0	2	$\begin{pmatrix}0&1\\1&1\end{pmatrix}\begin{pmatrix}1&0\\1&1\end{pmatrix}\begin{pmatrix}1&1\\0&1\end{pmatrix}\begin{pmatrix}1&1\\1&0\end{pmatrix}$
2	2	8J	0	$\begin{pmatrix}1&0\\0&1\end{pmatrix}\begin{pmatrix}0&1\\1&0\end{pmatrix}$
2	4	0	0	$\begin{pmatrix}1&1\\0&0\end{pmatrix}\begin{pmatrix}0&0\\1&1\end{pmatrix}\begin{pmatrix}1&0\\1&0\end{pmatrix}\begin{pmatrix}0&1\\0&1\end{pmatrix}$
1	4	0	-2	$\begin{pmatrix}1&0\\0&0\end{pmatrix}\begin{pmatrix}0&1\\0&0\end{pmatrix}\begin{pmatrix}0&0\\1&0\end{pmatrix}\begin{pmatrix}0&0\\0&1\end{pmatrix}$
0	1	-8J	-4	$\begin{pmatrix}0&0\\0&0\end{pmatrix}$

E. Phase transition

Near the critical temperature of a phase transition, the behaviour of many physical quantities can be characterized by a power law. The mean magnetization then scales as

$$\langle M(T)\rangle \sim (T - T_C)^{\beta} \tag{19}$$

and the heat capacity and susceptibility respectively as

$$C_V(T) \sim |T_C - T|^{\alpha} \tag{20}$$

$$\chi(T) \sim |T_C - T|^{\gamma} \tag{21}$$

with critical components $\beta = 1/8$, $\alpha = 0$ and $\gamma = 7/4$ [1]. Consider the correlation length χ, which also has a power law behaviour given by

$$\chi(T) = |T_C - T|^{\nu}. \tag{22}$$

As a second-order phase transition has a correlation length spanning the entire system, we can use the behaviour of the critical temperature at finite lattice sizes to extrapolate to the thermodynamic limit $L = \infty$, using so-called finite size scaling relations. We rewrite

$$T_C(L) - T_C(L = \infty) = aL^{-1/\nu} \tag{23}$$

which gives us the thermodynamic critical temperature for the Ising model as

$$T_C(L = \infty) = T_C(L) - aL^{-1/\nu} \tag{24}$$

From Lars Onsager [2], we have the analytical critical temperature in the thermodynamic linit

$$kT_C/J = 2/\ln(1 + \sqrt{2}) \approx 2.269, \tag{25}$$

and the parameter $\nu = 1$.

IV. METHODS

A. Implementation

We write a code for the Ising Model using the Metropolis algorithm and periodic boundary conditions. Given a size of the lattice grid L in x and y directions, it calculates the values of the mean energy E, mean magnetization M, the heat capacity C_V and the susceptibility χ per spin as functions of T.

The algorithm for transitioning to a new state is to flip one random spin, check the transition probability due to changes in energy w and comparing this to a random number r between 0 and 1. If the transition probability is higher than the random number $w > r$ then we perform the transition and update the energy, magnetization and spin matrix.

B. Parallelization

We implement two distinct methods for parallelizations, one in python for short calculations, when running the same program multiple times for different Monte Carlo cycles, and one with MPI (Message Passing Interface) in c plus plus. The python parallelization falls under the category of EP, embarissingly parallel, since the simulator is run multiple times separately without any communication between processes.

MPI provides another way of paralellizing, where all involved processors contribute to a common program. The communication between processes happens at specified points in the program. In our case, this is also very similar to EP, as all we are doing is splitting up the temperatures we want to calculate to the processes, which then do their job independently before gathering the data at the root node for printing and saving to file.

V. RESULTS

A. Numerical vs 2x2 analytical

To test the method, we run a 2×2 lattice grid simulation for increasing amounts of cycles to test when the equilibrium is reached. In figs. 1 to 2 we se the results compared to the analytical values from section III C. Observe that while the energy has a good correspondance even after a short amount of time, the magnetization needs more time to get closer to the analytical value.

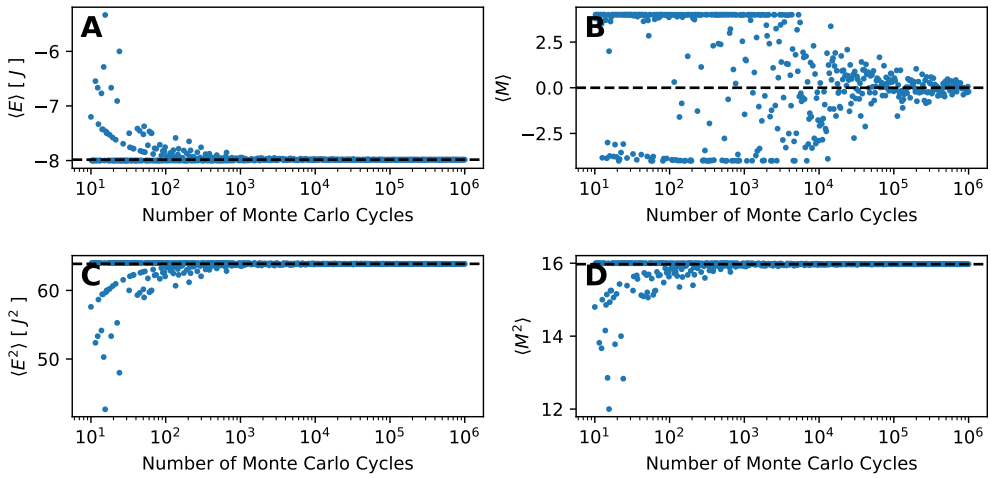

FIG. 1: Comparison of analytical and numerical expectation values for a 2×2 lattice at temperature $T = 1.0J/k_B$, as a function of Monte Carlo cycles. **A:** Energy, analytical value $\langle E \rangle = 7.9839$. **B:** Magnetization, $\langle M \rangle = 0$. **C:** Energy squared, $\langle E^2 \rangle = 63.871$. **D:** Magnetization squared, $\langle M^2 \rangle = 15.9732$.

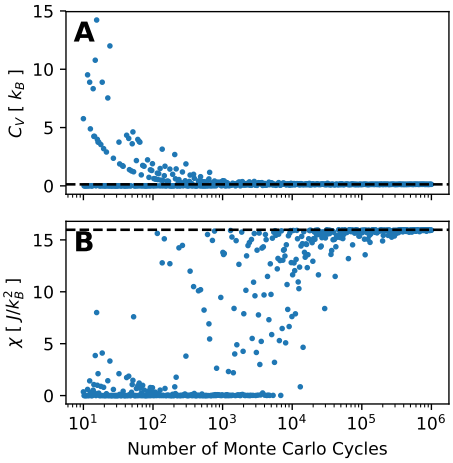

FIG. 2: Comparison of analytical and numerical quantities for a 2×2 lattice at temperature $T = 1.0J/k_B$, as a funciton of Monte Carlo cycles. **A:** Total heat capacity, analytical value $C_V = 0.12833$. **B:** Total susceptibility, analytical value $\chi = 15.9732$.

B. Equilibrium time

As the Metropolis algorithm is a stochastic method, we want to study how many cycles are needed for approaching an equilibrium as a function of temperature. We do this by running the code with $L = 20$, and printing the resulting expectation values while increasing the number of Monte Carlo cycles. This is done for two temperatures, $T = 1$ and $T = 2.4$. The results can be seen in fig. 3. In fig. 4, the number of accepted states during the simulation as a function of cycles is plotted, for the two temperatures. In terms of equilibration time, there is little difference between the runs for the two temperatures, and the same holds for an ordered spin configuration. We observe that the number of accepted states increases linearly with number of cycles, but that it initially is a bit higher.

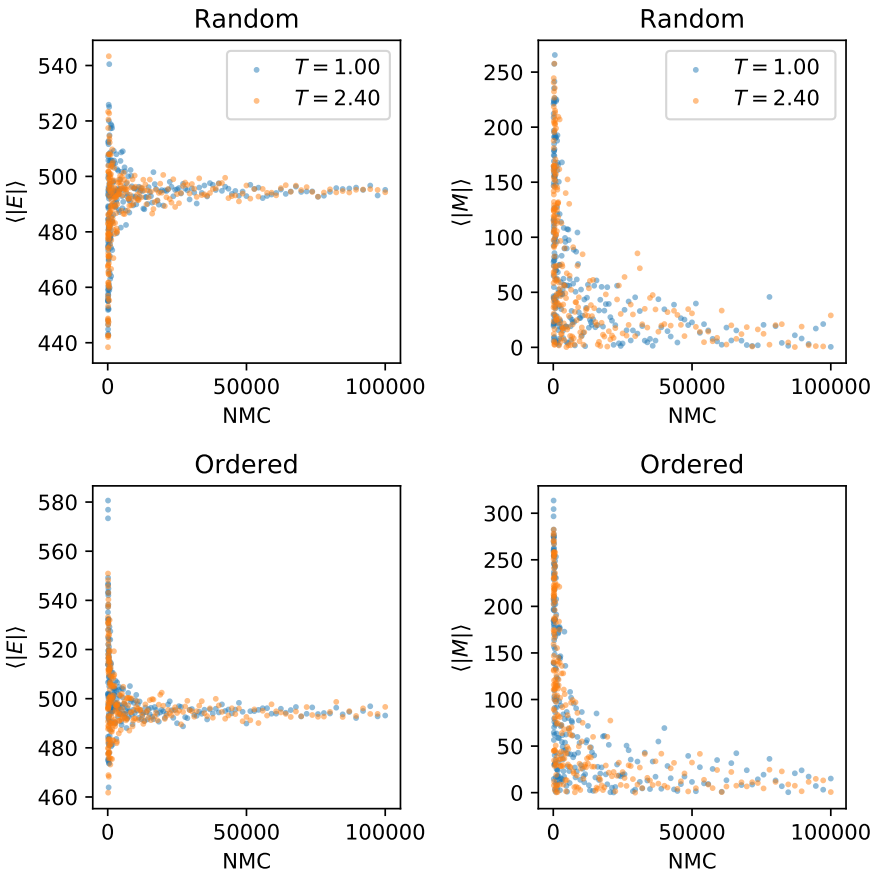

FIG. 3: Name

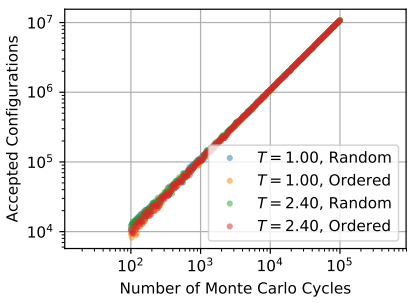

FIG. 4: Accepted

C. Probability distribution

After steady state situation is reached, we record which energies occur for a 20×20 lattice for both $T = 1$ and $T = 2.4$ for a ordered and unordered initial configurations, and get the distribution of energies shown in fig. 5. Figure 6 shows the evolution of the energy and the calculated expectation value (running average). As one can see, there is not much difference between random and ordered configuration.

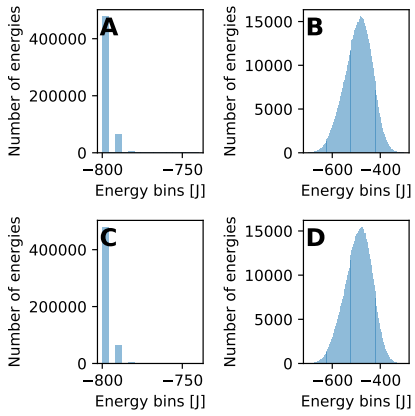

FIG. 5: Probability distribution of energies for $L = 20$. **A:** Random configuration, $T = 1$ **B:** Random configuration, $T = 2.4$ **C:** Ordered configuration, $T = 1$ **D:** Ordered configuration, $T = 2.4$

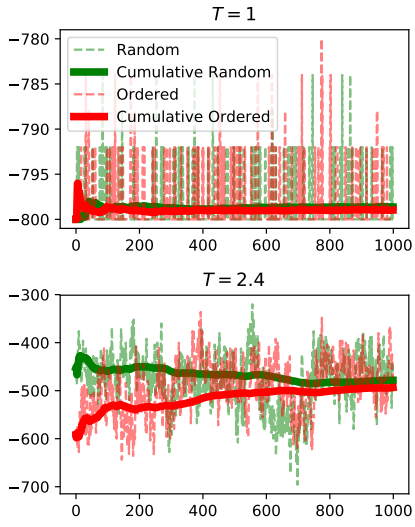

FIG. 6: Graphs over energies for $T = 1$ and $T = 2.4$, and random and ordered spin configurations, with running average (expecation values) as thick lines. Green is random and red is ordered.

TABLE II: Critical Temperature as function of lattice size L

L (Grid length)	40	60	80	100
T (Units of kT/J)	2.289	2.286	2.280	2.276

D. Phase transition and critical temperature

For lattice of $L \in \{40, 60, 80, 100\}$, we first run the model with 200 temperatures in the range $T \in [2, 2.6]$, with 10^6 Monte Carlo cycles. Figure 7 shows the expectation values of E and M, and the calculated values of heat capacity c_V and susceptibility χ, everything per spin. A piecewise Savitzky Golay filter is applied to C_V get a good estimate of the critical temperature, which we calculate from where the energy capacity has its peak. The extracted critical temperature for the different values of L is shown in table II. We solve eq. (23) for the values we know to get a function $T_C(L, T_C(L = \infty), a)$, to which we apply a curve fit with python and get the optimal parameters. The extracted critical temperature for the thermodynamic limit is

$$T_C(L = \infty) = 2.2694. \tag{26}$$

E. Method analysis

We then run several trial runs on 4 processors. To test the amount of overhead compared to the time usage of the actual Metropolis algorithm, we run two batches, one with 100 cycles per run, and one with 1000. The timing results can be seen in table III, with raw data shown in fig. 8.

TABLE III: Time gain from parallelization.

	1 prosessor	4 prosessors	Ratio
1000 cycles	2.224s	0.851s	2.612
10000 cycles	21.657s	8.273s	2.618

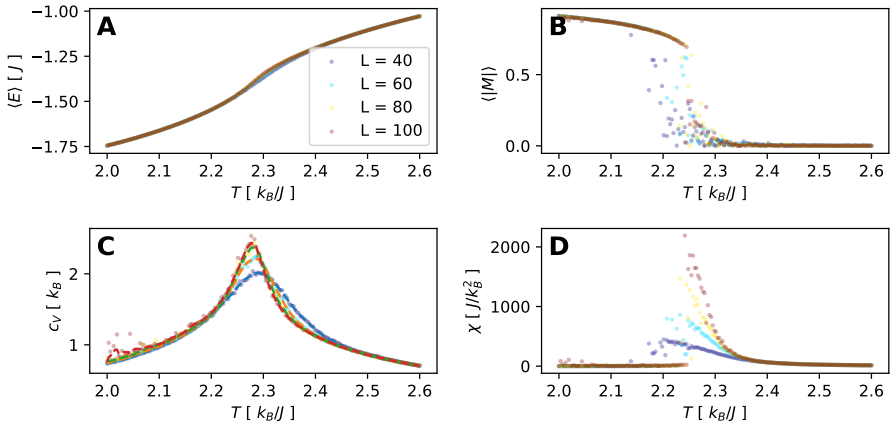

FIG. 7: Expectation values and other values for the two dimensional Ising model for temperatures near the critical temperature. **A:** Expectation value of the total energy, units of J. **B:** Expectation value of the magnitude of total magnetization. **C:** Specific heat capacity c_V, with a smoothed line. **D:** Specific susceptibility χ

.

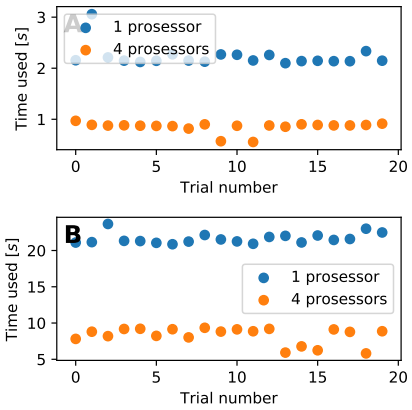

FIG. 8: Timing of the parallelisation. Our code is run on 1 and 4 prosessors to test whether we get an optimal gain. **A:** 1000 Monte Carlo cycles per trial. **B:** 10000 Monte Carlo cycles per trial.

VI. DISCUSSION AND CONCLUSIONS

Our Metropolis algorithm produces a value of 2.2695, very close to Onsagers result of 2.2692. This shows that the algorithm can be applied to other systems where the shear number of possible states makes direct evaluation of the probability distribution impossible.

We have several sources of errors, most of which we unfortunately have not quantified. The most important is probably the systematic statistical error. Near the critical temperature, the metropolis algorithm has a long equilibration time [1], as it methodically has to flip the right spins in order to reach a final "good" configuration near the equilibrium. This causes the values we pick for the system in the beginning to be highly unlikely values, and we should then run for a long time to let these be insignificant compared to the other more relevant values. The high equilibrium time can be seen in fig. 3, where the average magnetization needs roughly 10^6 monte carlo steps to reach a value of $1/e$ of the initial.

Figure 7 B shows the clearest example of

a second order phase transition in our Ising model, especially for higher grid sizes; for temperatures above to the critical temperature, the magnetization is close to zero, while below we get a spontaneous magnetization.

For the $L = 20$ lattice, our results are not as expected. We believe that our method for finding expectation values midrun is not an optimal one for the task, because we start the simulation over again each time to record a new expectation value for the magnetization. In hindsight, we realized that we should have used magnetization for this task instead, as it used longer time to equilibrate.

A. Method and parallelization

In section V E we gained a 2.6 times speedup for four prosessors with MPI. When parallelizing with MPI, it is not theoretically possible to get an infinite speed up, as there is a certain amount of overhead when allocating memory and distributing and gathering variables between processes. We have tried to minimize overhead, and for the longest runs, this is negligible in comparison to the time usage of the Metropolis method. The speed up for the EP-parallelization with python was not quantified, but the feeling of speed was significant.

APPENDIX

See `https://github.com/halvarsu/FYS3150/projects/Project4` for code. table IV shows the different programs developed, with brief explanations.

TABLE IV: Description of programs developed in this project

c++	(in folder `cpp`)
`main.cpp`	Main framework around metropolis solver, handles input, parallelization, initialization and output
`metropolis.cpp`	Holds the random number generators and main algorithm for Monte Carlo cycles
`tools.cpp`	Tools for reading data and getting periodic i's
python	(in folder `analyse`)
`analyse.py`	Runs cpp-program with parameters for analysis of 2x2 and 20x20 grid and produces plots. Part is defined by `-p` argument, use `-h` for help.
`energy_analysis.py`	Plots and analyses generated data for energy probability distribution
`phase_transitions.py`	Analyses phase transitions and critical temperatures. `-h` for help
`timing.py`	Sets up, runs and times the MPI-parallelization. Produces plots. `-h` for help
`read_data.py`	Deprecated
`tools.py`	Tools used by analyse.py
`setup_pt.py`	sets up bash-files
bash	(in folder `shell`)
`run_pt_subX.sh`	Simple scripts for running a subset of temperatures, to be distributed to different computers.

[1] M.H.Jensen. Fys3150 lecture notes. *CP. UiO*, fall 2015.

[2] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944.