Simulation of the 2D Ising model with Monte Carlo Markov Chain

(Dated: November 28, 2022)

We study the 2D ferromagnetic Ising Model with the Monte Carlo Markov chain. We use the Metropolis algorithm to sample the probability distribution of the Ising model. As proven in 1944 by Lars Onsager[1], we have a phase transition at the critical temperature $T_c(L = \infty) = 2.269$J/k. We find a matching numerical critical temperature $T_c(L = \infty) = 2.29 \pm 0.04$J/k. To verify that the Metropolis algorithm is working correctly, we used the analytical solution in the case of 2x2 lattice. Additionally, we look at the distribution of our energy normalized by spin ϵ for a 20x20 lattice with different temperatures. Temperature plays an important role in the shape of our distribution. Since we can only compute theoretical values when we reach equilibrium, we used the "Burn-in" method to determine the number of cycles needed to reach equilibrium. For a 20x20 lattice, we determine that around 10000 cycles are enough to reach equilibrium.

Github link: https://github.com/christopheblomsen/FYS4150_pro4

I. INTRODUCTION

The Ising model is a statistical mechanics model that describes ferromagnetism by a system of spins on a lattice. Ferromagnetism is a property of certain materials, allowing them to form a permanent magnet. A phase transition is one important concept in the Ising model for dimensions higher than 1. A phase transition is an extreme change in the physical properties. In our case, the model will go high magnetization to close to 0 with increasing temperature. More detail about Ising model? Other use in science perhaps

There is no analytical solution for the Ising model in dimensions > 2. However, we can use the Monte Carlo Markov Chain (MCMC) algorithm to simulate our model. MCMC was developed in the 40s by Stanislaw Ulam and John Von Neumann in Los Alamos. So the original paper could not be found, but the history is well documented in e.g.[2]. The basic idea is to use randomness to solve a deterministic problem and has shown remarkable results in many situations some example? in computational physics. In this paper, we will apply the Monte Carlo methods to solve the 2D Ising Model, and we will use the Metropolis algorithm to sample the distribution. As explained in more details in section II, we will use the relation between finite lattice and infinite lattice to estimate the critical temperature, $T_c(L = \infty)$ solved analytically by Lars Onsager in 1944.

In section II, we will formally explain the Ising Model with the analytical solution for the 2x2 case used as a benchmark for our algorithm. In the next section III we will discuss the numerical methods and the algorithms used. Section IV and section V respectively presents our results from our simulation and an analysis of them. Finally, section VI summarises this paper's main points.

II. THEORY

A. 2D Ising Model

The 2D Ising model is a 2-dimensional lattice of equal length and row (L) composed of spins s_i. So a lattice of size L will wield LxL = N spins. Each spin can take one of the following values $\{-1, 1\}$ representing their magnetic dipole. Every spin can interact with its next neighbour, meaning that every spin can have up to four interactions. One important detail in physics is what to do with the boundaries. In our case, we will consider a periodic boundary condition. This means that we will always consider four interactions for each spin. This is the geometrical equivalent of having a lattice in the shape of a torus. We will denote the spin configuration by $\mathbf{s} = \{s_1, s_2, ..., s_N\}$. Which allows us to write the Hamiltonian

$$H(\mathbf{s}) = -J \sum_{\langle kl \rangle} s_k s_l + h \sum_{i=1}^{N} s_i \tag{1}$$

Eq. 1 is composed of two terms. The first one is the contribution of every neighbour. $\langle kl \rangle$ represents the neighbouring pairs of k and l. J is the coupling constant and represents the strength of spin interaction. We produced all the results in this paper with $J = 1$. The second term is here to account for a potential external magnetic field with h acting as the strength of this magnetic field. This paper will not consider an external magnetic field, so $h = 0$ and eq. 1 becomes

$$H(\mathbf{s}) = -J \sum_{\langle kl \rangle} s_k s_l \tag{2}$$

In statistical mechanics, we use probability and observables. In our problem, we have two observables the magnetisation and the energy, respectively

$$M(\mathbf{s}) = \sum_{i=1} s_i \tag{3}$$

$$E(\mathbf{s}) = H(\mathbf{s}) = -J \sum_{\langle kl \rangle} s_k s_l \tag{4}$$

M is the magnetisation summing all spins in the lattice. Since we will study these observable for different lattice sizes, we introduce the normalised by spin magnetisation and energy

$$m(\mathbf{s}) = \frac{M(\mathbf{s})}{N} \tag{5}$$

$$\epsilon(\mathbf{s}) = \frac{E(\mathbf{s})}{N} \tag{6}$$

Given a temperature T to determine the probability of being in the configuration $\mathbf{s}$ we will use the Boltzmann distribution with the partition function Z

$$p(\mathbf{s}, T) = \frac{1}{Z} e^{-\beta E(\mathbf{s})} \tag{7}$$

$$Z = \sum_{\mathbf{s}} e^{-\beta E(\mathbf{s})} \tag{8}$$

where $\beta = \frac{1}{kT}$, k the Boltzmann constant. the summation for the partition function means that we iterate over every possible $\mathbf{ss}$. These numbers scale as 2^N showing with the analytical calculation of the partition for a big lattice becomes quickly impossible.

In statistical mechanics, we can only experimentally calculate the averaged values of our observable. Meaning that for a discrete observable $A(X)$ with probability $p(X)$ we have

$$\langle A \rangle = \sum_X A(X) p(X) \tag{9}$$

We can then apply eq. 9 to our observables, and we can define two values of interest, C_v and χ. The first is the heat capacity, and χ is the magnetic susceptibility.

$$C_v = \frac{\langle E^2 \rangle - \langle E \rangle^2}{NkT^2} \tag{10}$$

$$\chi = \frac{\langle M^2 \rangle - \langle |M| \rangle^2}{NKT} \tag{11}$$

B. 2x2 analytical solution

Since it gets quickly impossible to calculate the probability distribution due to the partition function. We will first solve the 2x2 case where we only have $2^4 = 16$ possible $\mathbf{s}$.

# $s_i = 1$	E [J/k_B]	M[1/J]	degeneracy
0	-4	-4	1
1	0	-2	4
2	0	0	4
2	4	0	2
3	0	2	4
4	-4	4	1

TABLE I. Summary of all possible energy E, magnetisation M and the degeneracy is the number of spin configurations with the same characteristic

Table I shows all possibilities in the 2x2 case, and we see that summing all of the degeneracy gives us 16, assuring us that we covered every possibility. Now that we have found these values. We can proceed with the calculation of the partition function.

$$Z = 12 + 4\cosh(8\beta) \tag{12}$$

With Table I, eq. 12 and 9 we obtain the following relations

$$\langle E \rangle = \frac{-32\sinh(8\beta)}{Z}$$

$$\langle \epsilon \rangle = \frac{\langle E \rangle}{4} = \frac{-8\sinh(8\beta)}{Z}$$

$$\langle E^2 \rangle = \frac{256\cosh(8\beta)}{Z}$$

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

$$\langle M^2 \rangle = \frac{32}{Z}(1 + e^{8\beta})$$

$$\langle C_v \rangle = \frac{\langle E^2 \rangle - \langle E \rangle^2}{4}$$

$$\langle \chi \rangle = \frac{\langle M^2 \rangle - \langle |M| \rangle^2}{4}$$

C. Phase transition

One of the most interesting physical properties of the 2D Ising model is the phase transition. It means that our lattice will suddenly magnetise for a specific critical temperature T_c. For an infinite-size model, Lars Onsager found in 1944 that the theoretical value of

$$T_c(L = \infty) = \frac{2}{\log\left(1 + \sqrt{2}\right)} \text{ J}/k_B \approx 2.269 \text{ J}/k_B \tag{13}$$

Another important aspect of phase transition is that different system exhibit similar behaviour around the critical temperature. This is known as the critical exponents. We have the finite size from chapter 13 in the lecture notes[3].

$$\xi \propto |T - T_c(L = \infty)|^{-\nu} \tag{14}$$

with critical exponent $\nu = 1$, ξ is the correlation length and the largest possible value is $\xi = L$, which we associate with the critical temperature of our system.

With our finite critical temperature $T_c(L)$, we can estimate numerically the infinite critical temperature with the scaling relation

$$T_c(L) = aL^{-1} + T_c(L = \infty) \tag{15}$$

a is a constant. This allows us to perform a linear regression to find an approximate value of the slope (a) and the intersect point, $T_c(L = \infty)$.

III. METHODS

A. MCMC and Metropolis algorithm

We will use the Monte Carlo Markov Chain (MCMC) to compute our model. This method use statistic to study deterministic problem like the Ising model. We define a probability distribution function and sample states from it. After some iteration or cycle, we obtain an ensemble of average, which allows us to understand a system when an analytical solution does not exist.

A Markov Chain is a stochastic process where each step possesses a probability independent of the history. This means that the only relevant information needed is the latest step to calculate the probability of transitioning, but all previous states are unnecessary. This removes many constraints where keeping track of history would raise big memory issues. Fortunately, this is not the case here.

For the Ising model, we have defined the distribution probability.

$$p_{\mathbf{s}} = \frac{e^{-\beta E(\mathbf{s})}}{Z} \tag{16}$$

Now we will need two assumptions. The first one is ergodicity. This means that every state can be reached within a finite amount of time. This allows us to consider every state's possibilities as we assume that no state can prevent us from reaching equilibrium (the states with the biggest probability). We also use the detailed balance. This means that if we have two states P_1 and P_2, we have as much transition from state 1 to 2 then 2 to 1. Here this means that:

$$P_1 T(1:2) = P_2 T(2:1) \tag{17}$$

$$\frac{T(1:2)}{T(2:1)} = \frac{P_2}{P_1} \tag{18}$$

$$= e^{-\beta \Delta E} \tag{19}$$

Where $T(i : j)$ denotes the transition probability of going from state i to j. The last line is obtained when we replace the P_i with eq 16.

However, we do not know the actual transition probability, but we do not need to know it with the Metropolis algorithm. This algorithm creates an acceptance function that respects our system's constraint and is used to represent our transition probability. We assume that our transition probability should depend on 19. This gives us

$$T(1:2) = \min(1, e^{-\beta \Delta E}) \tag{20}$$

This equation satisfies all of our constraints. We can see that the transition can only take values between 0 and 1.

The algorithm

We create a lattice of size $L \times L$ full of spin of values {-1, 1}. Since we consider boundary conditions, we use the modulo operator to access the spin at the beginning or end of our lattice once we reach the opposite side while calculating the Hamiltonian. We initialise our lattice either randomly or ordered with all spin in the up position (value of 1).

Once we have initialised our system with the correct temperature and a random initial state. We start our Monte Carlo algorithm. One cycle corresponds of N spin flip attempt. N being the total number of spins. We take one random spin, flip it, calculate the energy difference, and have three options. If $\Delta E \leq 0$, we accept the state, change our lattice and compute the magnetisation and the energy. If $\Delta E > 0$, we have two possibilities. We will draw a random number u between 0 and 1 from a uniform distribution , and if $u < \Delta E$, we accept the new state. If that is not the case, we refuse the state to keep our current state and add their energy and magnetisation to our data once more.

We will compute K number of cycles, but we will keep only $K - N_{\text{burn}}$ values because in statistical mechanics, we need to reach equilibrium to match the experimental. This method is called the burn-in method and assumes that N_{burn} cycle is enough to reach equilibrium in our system. The determination of this value will be discussed in more detail in Section IV B.

Finally, we will use parallelisation with OpenMP to accelerate our computational time. We will use it to simulate many temperatures while studying the phase transition. This will allow us to parallelise the temperature loop as each iteration is independent. This operation reduced our computing time up to 50%.

IV. RESULTS

A. 2x2 numerical simulation

We used $T = 1\ \mathrm{J}/k_B$ and simulated {500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000, 20000, 30000, 40000, 50000, 60000, 70000, 80000, 90000, 100000} cycles and $L = 2$. We computed the analytical solution described in Section II B. Figures 1 and 2 shows the expected value of χ and C_v for the different number of cycle done. The theoretical values are plotted in a dashed line. We did not represented $\langle m \rangle$ or $\langle E \rangle$ as they are a necessity to compute χ and C_v so finding good results for the last two shows that we have evaluated correctly the other quantities.

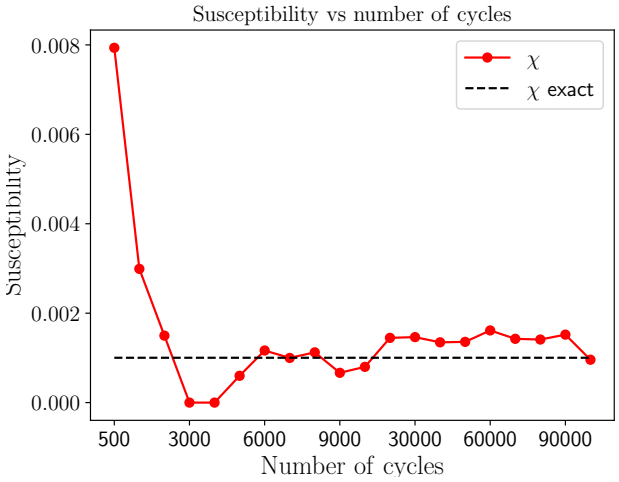

FIG. 1. The calculated susceptibility over number of cycles in red which converges to the exact solution in black.

B. Burn-in time estimation

For the Burn-in testing, we took the same number of cycles as Section IV A, with $L = 20$ instead of $L = 4$. We used two different temperatures $T_1 = 1\ \mathrm{J}/k_B$ and $T_2 = 2.4\ \mathrm{J}/k_B$ starting from ordered (all spin being 1) and random. We plotted the expected value of m and ϵ according to the number of cycles. The results are presented in Figures 3 and 4.

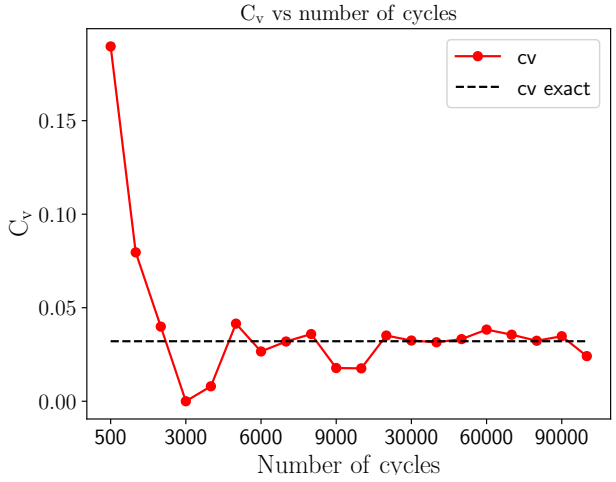

FIG. 2. Heat capacity plotted against the number of cycles in red which converges to the exact solution in black.

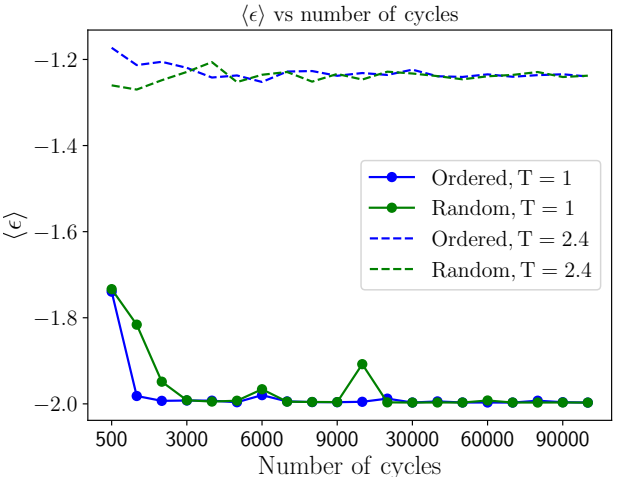

FIG. 3. The Burn-in for the $\langle \varepsilon \rangle$ for both the ordered (in blue) and random (in green) states. And the $T = 1\ \mathrm{J}/k_B$ and $T = 2.4\ \mathrm{J}/k_B$.

C. Distribution probability of ϵ

For the probability distribution of ϵ, we took the data set used in the section IV B, so 100000 cycles computed for both temperatures without any burn-in time and starting from a random position. Figures 5 and 6 shows the histogram of our distribution for their temperature. We also computed the variance and found $\sigma \approx 0.00011$ for $T = 1\ \mathrm{J}/k_B$ and $\sigma \approx 0.02023$ for $T = 2.4\ \mathrm{J}/k_B$.

V. DISCUSSION

Figures 1 and 2 shows the expected value of the susceptibility and the heat capacity for different MCMC cycle.

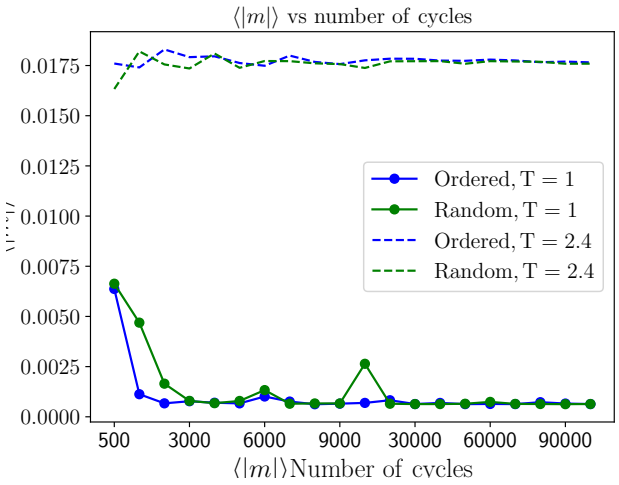

FIG. 4. The Burn-in for the $\langle|m|\rangle$ for both the ordered (in blue) and random (in green) states. And the $T = 1$ J/k_B and $T = 2.4$ J/k_B.

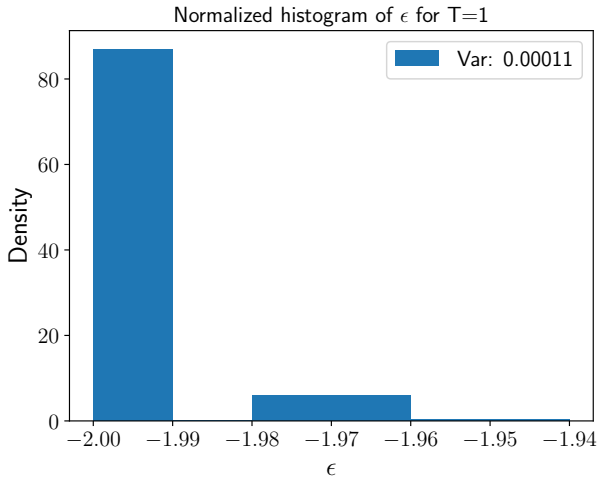

FIG. 5. Normalized histogram over the energy when the temperature was $T = 1$ J/k_B.

We see that we quickly reach the theoretical value in 2000 cycles. Increasing the number of cycles will provide better and better approximation as expected. However, we notice that we never are perfectly constant and adding more cycles does not assure us that we will reach the theoretical value. This can be explained due to the randomness of our algorithms. Indeed, if we start in a low probability state, we can spend some time equilibrating which would weigh heavily on the expected value. These two figures show that the burn-in method would significantly improve the numerical value.

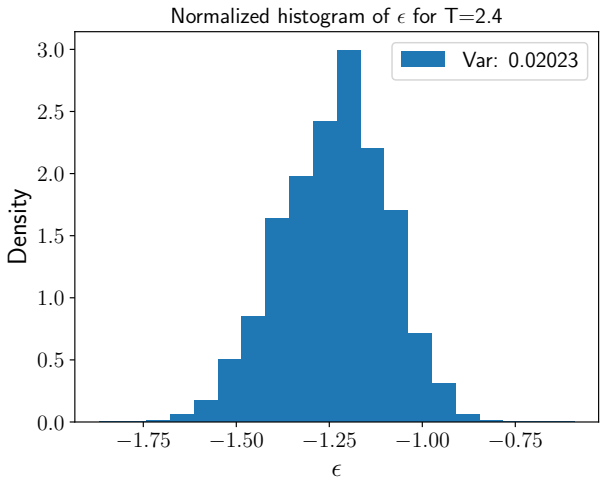

FIG. 6. Normalized histogram over the energy where the temperature was $T = 2.4$ J/k_B.

A. Burn-in

We now take a look at the equilibration time. There are many ways of calculating the required time for our algorithm to reach equilibrium time, as discussed in chapter 13.9 in the lecture notes [3]. We chose to implement the burn-in method. This method lets our code run for a certain number of cycles N_{burn}, and once we reach the $N_{\text{burn}} + 1$ cycle, we consider that we have reached equilibrium, and we start saving the data of interest.

B. Distribution of ϵ

We observe from the Figure 5 and Figure 6 that the temperature directly impacts the probability distribution of the energy states that our Ising model can take. We can imagine a Boltzmann distribution from our normalised histogram Figure 5. While Figure 6 looks more like a gaussian distribution, but could also see as a Boltzmann distribution. We can conclude from this that the distribution of energy is lower for lower temperatures. This is to be expected since one needs temperature to perform the transitions.

C. Parallelization

The parallelisation done with OpenMP significantly boosted the run time. On our local machine, it went from having a run time in serial: 41 s. When we parallelised over the temperatures, we got the time down to 12 s. That is an increase on ~ 3.4, which we could do because the temperature simulations were independent.

D. Phase transition and critical temperature

We observe from the Figure 4 that the magnetisation converges towards 0. The 2D Ising model means that we have surpassed a critical temperature. Which is an example of what is usually referred to as the critical phenomena. And in our case, that is $\langle |m| \rangle$. This does not happen in the 1D Ising model but in the 2D and 3D models. When we reach the critical temperature and turn it into a paramagnetic.

VI. CONCLUSION

We have studied the 2D Ising model with the Monte Carlo Markov Chain algorithm using the Metropolis algorithm to sample the spin configuration. We found different distributions of ϵ for different temperatures and that it plays an important role in the distribution's shape. The main focus of this paper is the phase transition. For different lattice sizes, we iterated over four different lattice sizes $L = \{40,\ 60,\ 80,\ 100\ \}$, and used the MCMC algorithm to compute the expected values of χ and C_v. This allowed us to find a critical temperature of $T_c(L = \infty)$ to be 2.29 ± 0.04 J$/k_B$ where the error is calculated via the fitting function *linregress* from the **scipy** library. The numerical value is in the range of the theoretical value $T_c^{\text{th}} \approx 2.269$ J$/k_B$.

REFRENCES

[1] L. Onsager, Crystal statistics. i. a two-dimensional model with an order-disorder transition, Phys. Rev. **65**, 117 (1944).

[2] C. Robert and G. Casella, A short history of markov chain monte carlo: Subjective recollections from incomplete data, Statistical Science **26**, 10.1214/10-sts351 (2011).

[3] M. Hjorth-Jensen, Computational physics (2015).