

# Project 4 - FYS4150

Nanna Bryne, Johan Mylius Kroken, Vetle A. Vikenes  
(Dated: November 24, 2022)

Abstract. The code is found in the following GitHub repository: <https://github.com/Vikenes/FYS4150/tree/main/project4>.

Please use the following convention for labeling subsections: For a section with label “sec:SectionLabel”, label its subsections with “subsec\_SectionLabel:LabelName”. This will make our lives easier towards the end. Dummy example:

```
\section{Important algorithms} \label{sec:algo}
\subsection{initialization} \label{subsec_algo:initial}
```

## I. INTRODUCTION

[1]

## II. THEORETICAL BACKGROUND

In this section we present a brief introduction to the 2D Ising model, as well some theoretical background of phase transition phenomena and statistical mechanics. Additionally, we briefly describe the overall ideas behind the workings of Markov chain Monte Carlo (MCMC) methods, in particular how this is implemented with the Metropolis algorithm. Details regarding our own implementation of an MCMC algorithm will be presented in Section 69.

### A. 2D Ising model

The Ising model is a mathematical model used to model statistical properties of materials, such as ferromagnetism. It consists of discrete magnetic moments localized in a particular lattice. For simplicity, we will refer to the magnetic moments as “spins” throughout this report. Additionally, we will restrict our analysis to a square lattice in two dimensions without an external magnetic field. Each spin has a fixed position in the lattice, where they are allowed to interact with their immediate neighbors. We will denote a single spin as  $s_i$ , where the subscript  $i$  refers to a certain position in the lattice. An individual spin can be in one of two possible states having a value of  $s_i = +1$  or  $s_i = -1$ . We will refer to the (an arbitrary) spin state of the entire lattice as a *microstate*, which we will denote as  $\nu$  (Don’t want vector or hat/tilde, and other Greek letters may cause confusion). The lattice has dimension ( $L \times L$ ), where we refer to  $L$  as the lattice size. The number of (individual) spins in the lattice is thus  $N = L^2$ . Explain that  $\nu$  is a

matrix? The total energy of the system is given by

$$E(\nu) = -J \sum_{\langle kl \rangle}^N s_k s_l, \quad (1)$$

where  $\langle kl \rangle$  indicates that the sum is taken over the nearest neighboring pairs of spins. Explain which neighbors are summed over. The parameter  $J$  represents the strength of interactions between neighboring spins, and we assume a constant  $J > 0$ , so that energy is minimized when neighboring spins are aligned (Energy minimization sentence should perhaps be removed...).

The total magnetization of the system will be given by the sum over all spins

$$M(\nu) = \sum_i^N s_i. \quad (2)$$

### B. Phase transitions and critical phenomena

This section is yet to be completed. I will write it more properly once I know we shall use it in practice.

One of the most peculiar features of continuous phase transitions is that thermodynamic quantities are found to exhibit similar behavior for a variety of different systems. These phase transitions can be characterized by *critical exponents*, where physical quantities behave according to power laws near the critical point. Mention universality classes? For a 2D Ising model of infinite size, the behavior of the fill in...

$$\langle |M| \rangle \propto |T - T_c(L = \infty)|^\beta, \quad (3)$$

$$C_V \propto |T - T_c(L = \infty)|^{-\alpha}, \quad (4)$$

$$\chi \propto |T - T_c(L = \infty)|^{-\gamma}, \quad (5)$$

$$\xi \propto |T - T_c(L = \infty)|^{-\nu}, \quad (6)$$

where the critical exponents are  $\beta = 1/8$ ,  $\alpha = 0$ ,  $\gamma = 7/4$  and  $\nu = 1$ . For our finite system, the largest possible correlation length is  $\xi = L$ , associated with  $T = T_c(L)$ , which is the critical temperature of our finite system. Using the constraint on  $\xi$  yields the following relation

$$T_c(L) - T_c(L = \infty) = aL^{-1}, \quad (7)$$

where  $a$  is a proportionality constant. I will connect the dots when I know how we will do this.

The analytical value of  $T_c(L = \infty)$  for the 2D Ising model with no external magnetic field has found to be [1]

$$T_c(L = \infty) = \frac{2}{\ln(1 + \sqrt{2})} J/k_B \approx 2.269 J/k_B. \quad (8)$$

### C. Statistical mechanics

The probability of the system being in a microstate  $\nu$  at a given energy and a constant temperature,  $T$ , is governed by the Boltzmann distribution

$$p_\nu(T) = \frac{1}{Z} e^{-\beta E(\nu)}, \quad (9)$$

where  $\beta = 1/k_B T$  with  $k_B$  being the Boltzmann constant.  $Z$  is the partition function, defined as

$$Z = \sum_{\nu} e^{-\beta E(\nu)}, \quad (10)$$

where the sum goes over all possible microstates  $\nu$ .

For an observable  $Q(\nu)$ , its expectation value is given by

$$\langle Q \rangle = \sum_{\nu} Q(\nu) p_\nu(T) = \frac{1}{Z} \sum_{\nu} Q(\nu) e^{-\beta E(\nu)}. \quad (11)$$

Should perhaps include a paragraph of estimated averages, and how these converge to the true thermal averages (ergodic hypothesis (not the same “ergodic” that is mentioned in the MCMC subsection))

### D. Monte Carlo methods

To study thermodynamic properties of the Ising model, we need to compute the partition function in Eq. (10). However, this would require a sum over all possible microstates. For a lattice of size  $L$  the total number of **unique** microstates are  $2^{L^2} = 2^N$ . This becomes an impossible task for the lattice sizes we are going to consider. To overcome this problem we may draw random samples of the system and use these samples to estimate thermal averages. A common choice for sampling the system is the Markov chain Monte Carlo (MCMC) method. A Markov chain is a stochastic process in which the outcome of an event is independent of the process’s history. Loosely speaking, it can be regarded as a random walk in (our **x-dimensional**) state space. There are several algorithms for constructing these Markov chains, but the one we will consider is the *Metropolis-Hastings* algorithm.

### E. Metropolis algorithm

The Metropolis algorithm uses a Markov process to generate multiple samples of microstates that approximate a Boltzmann weighted ensemble. A Markov process

is defined by a transition probability,  $W(\nu \rightarrow \nu')$ , which is the probability of the system to transition from a state  $\nu$  to any given state  $\nu'$ . For the Markov chain to reach the desired distribution we have to fulfill two conditions. The first condition is *ergodicity* (I will add a footnote here later), which states that the system has a non-zero probability of going from any state to any other state with a finite sequence of transitions. The second condition is *detailed balance*, meaning that we require each transition to be reversible. That is, the probability for the system to be in a state  $\nu$  and transition to a state  $\nu'$  is equal to the probability for the system to be in state  $\nu'$  and transition to state  $\nu$ . Mathematically, this can be formulated as

$$P(\nu)W(\nu \rightarrow \nu') = P(\nu')W(\nu' \rightarrow \nu), \quad (12)$$

where  $P(\nu)$  is the distribution of the Markov process (Improve explanation). It can be shown that when the conditions of ergodicity and detailed balance are fulfilled, the distribution  $P(\nu)$  is both unique and stationary. I’m not sure if this is mathematically rigorous. Citation should definitely be considered (I found this on Wikipedia).

A naive approach for sampling configurations would be to sample random microstates uniformly. This has the disadvantage of including microstates which the system is unlikely to be in, and many computations would be required to get a satisfying distribution. Instead, the Metropolis algorithm works by choosing the Boltzmann distribution to sample spins, i.e.  $P(\nu) = p_\nu(T)$ . If we insert this into Eq. (12) and rewrite it, we get

**I should not introduce delta E here. I will change later**

$$\frac{W(\nu \rightarrow \nu')}{W(\nu' \rightarrow \nu)} = \frac{p_{\nu'}(T)}{p_\nu(T)} = e^{-\beta \Delta E}, \quad (13)$$

where we defined  $\Delta E \equiv E(\nu') - E(\nu)$ . Importantly, in Eq. (13) we now see that the partition function has vanished. In general,  $W(\nu \rightarrow \nu')$  is unknown, and the Metropolis algorithm works by assuming that it can be written as the product of the two probabilities  $A(\nu \rightarrow \nu')$  and  $T(\nu \rightarrow \nu')$ , the probability of accepting the transition from  $\nu \rightarrow \nu'$  and the probability of making the transition to  $\nu'$  being in state  $\nu$ . One common choice is to assume a symmetric transition probability  $T(\nu \rightarrow \nu') = T(\nu' \rightarrow \nu)$ . Eq. (13) then reads

$$\frac{A(\nu \rightarrow \nu')}{A(\nu' \rightarrow \nu)} = e^{-\beta \Delta E}. \quad (14)$$

Since energy tends to be minimized, a natural choice is to always accept new states if it has a lower energy than the initial state since  $\Delta E < 0$  corresponds to transitioning to a state with a higher probability. Assuming an acceptance probability of 1 for a transition resulting in a lower energy, we can express the transition probability  $A(\nu \rightarrow \nu')$  as

$$A(\nu \rightarrow \nu') = \min(1, e^{-\beta \Delta E}). \quad (15)$$

With Eq. (15), we will always transition to states with lower energies. Since ergodicity must be fulfilled, we may not reject any transition that doesn't lower the energy. In Section 420 we discuss how we implement this in practice.

### III. IMPLEMENTATIONS

#### Have we mentioned PBC yet?

##### A. Energy change due to single spin flip

When flipping a single spin there is a limited number of possible values  $\Delta E$  can take. Consider an arbitrary spin,  $s_k$ , in a lattice with  $L > 2$ . The energy contribution from this spin's interaction with its neighbors is

$$E(s_k) = -J s_k \sum_{\langle l \rangle} s_l. \quad (16)$$

Flipping this spin corresponds to  $s_k \rightarrow -s_k$ . The change in energy from this is  $\Delta E = E(-s_k) - E(s_k)$ , resulting in the following expression:

$$\begin{aligned} \Delta E &= -J(-s_k) \sum_{\langle l \rangle} s_l - (-J s_k) \sum_{\langle l \rangle} s_l \\ &= 2J s_k \sum_{\langle l \rangle} s_l. \end{aligned} \quad (17)$$

Since each of the spins  $s_l$  takes a value of  $-1$  or  $+1$ , taking the sum over four spins can only yield a value of  $\sum_{\langle l \rangle} s_l = \{-4, -2, 0, 2, 4\}$ . With  $s_k = \pm 1$ , it's evident that flipping a single spin results in five possible values of  $\Delta E$ , which are

$$\Delta E = \{-8J, -4J, 0, 4J, 8J\}. \quad (18)$$

Relate this to metropolis.

Flipping a spin also yields a change in magnetization,  $\Delta M \equiv M(\nu') - M(\nu)$ , from the transition  $s_k \rightarrow s'_k = -s_k$ . The new magnetization is easily seen to be

$$M(\nu') = M(\nu) + 2s'_k, \quad (19)$$

with  $\Delta M = -2, +2$  as the only possibilities, corresponding to initial spin values of  $s_k = +1, -1$ , respectively.

##### B. Metropolis algorithm

To implement the Metropolis algorithm we determine whether to accept a spin flip by comparing  $A(\nu \rightarrow \nu')$  from Eq. (15) with a random number  $r \sim \mathcal{U}(0, 1)$  (should it be  $r$  in  $\mathcal{U}(0, 1)$ ?). This method guarantees that the proposed spin flip is accepted if  $\Delta E < 0$ . Additionally, transitions yielding  $\Delta E > 0$  have a higher probability of being accepted if the associated reduction of the Boltzmann distribution is small.

**w in the enumeration below must be defined/introduced.**

To sample spins, we will draw spins in the lattice according to a uniform random distribution. The basic outline of our MCMC algorithm is as follows:

1. Pick a random spin in the lattice,  $s_k$ .
2. Compute  $w = \exp(-\beta \Delta E)$  resulting from spin flip
3. If  $w \geq r$ : Accept flip, update  $E$  and  $M$ .
4. If  $w < r$ : Reject proposed flip.
5. Repeat.

Repeating this process  $N = L^2$  number of times constitutes one MC *cycle*. After each cycle, we use the final values  $E$  and  $M$  to store as our samples.

and we accept the flip, and proceed by picking a new spin at random.

The algorithm will be changed later.

---

#### Algorithm 1 Flip

---

Pick  $s_k$  on a random position in the lattice.

Compute  $\Delta E$

**if**  $\Delta E \leq 0$  **then**

$s_k = s_k * (-1)$

$E = E + \Delta E$

$M = M + 2 * s_k$

**else**

$w = \exp(-\beta \Delta E)$

$r \in \mathcal{U}(0, 1)$

**if**  $r \leq w$  **then**

$s_k = s_k * (-1)$

$E = E + \Delta E$

$M = M + 2 * s_k$

---

### IV. METHODS

#### A. Analytical comparison

To test our implementation, we will first consider a lattice of size  $L = 2$  for which we can compute the analytical solution. The quantities we will consider are  $\langle \epsilon \rangle$ ,  $\langle |m| \rangle$ ,  $C_V$  and  $\chi$ . The analytical derivation of these quantities for  $L = 2$  is given in Appendix A. These analytical quantities will be compared to the numerical estimates after different number of cycles have been run. Bleh, drittforklaring.

#### B. Equilibration time

When initializing the system is unlikely to be in a state near equilibrium. This means that a majority of the cycles we run in the beginning will consist of transitions such that the system approaches an equilibrium. After a certain number of iterations the samples we draw are

likely to oscillate around the equilibrium values. The number of cycles needed to reach this equilibrium is called the *equilibration time*,  $t_{\text{eq}}$ . To reduce the number of cycles needed to obtain accurate estimates of thermal averages, we only include samples drawn after we have performed  $t_{\text{eq}}$  equilibration steps.

To estimate  $t_{\text{eq}}$ , we will consider a lattice with  $L = 20$ , and plot  $\langle \epsilon \rangle$  and  $\langle |m| \rangle$  as a function of cycles. From this we will choose a rough estimate of  $t_{\text{eq}}$ . We will do this for  $T = 1$  unit and  $T = 2.4$ , and in both cases study the result from a lattice that is initially ordered (all spins up) and a lattice that is unordered (random). **Mention ordered/disordered.**

### C. Estimating the probability distribution

### D. Phase transitions

## V. RESULTS

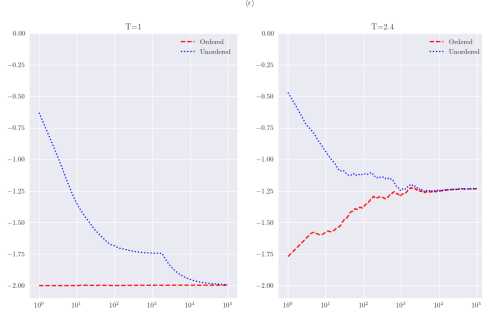


FIG. 1. Hei

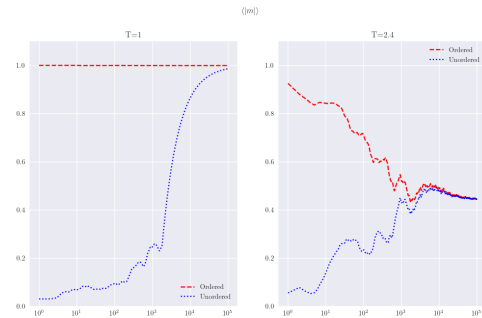


FIG. 2. Sug meg

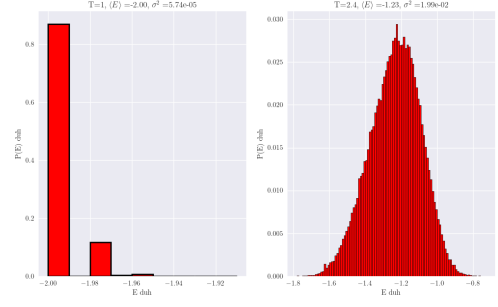


FIG. 3. Ser man det

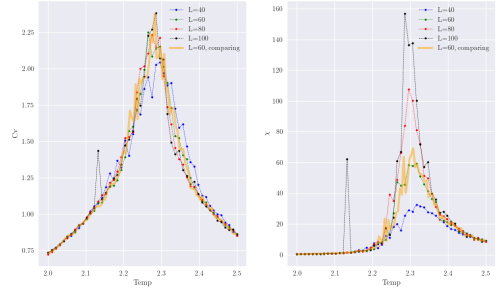


FIG. 4. Hvem vinner VM?

$N$	$\langle \epsilon \rangle$	$\langle  m  \rangle$	$\langle C_V \rangle$	$\langle \chi \rangle$
$10^2$	-1.940	0.980	0.466	0.058
$10^3$	-1.984	0.994	0.127	0.023
$10^4$	-1.996	0.999	0.029	0.004
$10^5$	-1.996	0.999	0.029	0.003
Analytical	-1.996	0.999	0.032	0.004

TABLE I. Comparison of analytical results.

## VI. DISCUSSION

## VII. CONCLUSION

### Appendix A: Analytical results

$N_{\uparrow}$	$E(s)$	$M(s)$	Degeneracy
4	-8 J	4	1
3	0	2	4
2	8 J	0	2
2	0	0	4
1	0	-2	4
0	-8 J	-4	1

TABLE II.

$$\langle \epsilon \rangle = -\frac{\partial \ln Z}{\partial \beta} = -\frac{8J}{Z} \sinh(8\beta J), \quad (\text{A2})$$

$$\langle \epsilon^2 \rangle = \sum_s \epsilon(s)^2 \frac{1}{Z} e^{-\beta E(s)} = \frac{16J^2}{Z} \cosh(8\beta J), \quad (\text{A3})$$

$$\langle |m| \rangle = \sum_s |m(s)| \frac{1}{Z} e^{-\beta E(s)} = \frac{2}{Z} (2 + e^{8\beta J}), \quad (\text{A4})$$

$$\langle m^2 \rangle = \sum_s m(s)^2 \frac{1}{Z} e^{-\beta E(s)} = \frac{2}{Z} (1 + e^{8\beta J}). \quad (\text{A5})$$

$$\begin{aligned} C_V &= \frac{\beta}{NT} \left( \langle E^2 \rangle - \langle E \rangle^2 \right) \\ &= \frac{64\beta J^2}{ZT} \left[ \cosh(8\beta J) - \frac{4}{Z} \sinh^2(8\beta J) \right], \end{aligned} \quad (\text{A6})$$

$$\begin{aligned} \chi &= \frac{\beta}{N} \left( \langle M^2 \rangle - \langle M \rangle^2 \right) \\ &= \frac{8\beta}{Z} \left[ (1 + e^{8\beta J}) - \frac{2}{Z} (2 + e^{8\beta J})^2 \right]. \end{aligned} \quad (\text{A7})$$

$$Z = 12 + 2e^{-8\beta J} + 2e^{8\beta J} = 12 + 4 \cosh(8\beta J) \quad (\text{A1})$$

- 
- [1] Onsager, L. (1944). Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149.