Ising model report

Tung D. Nguyen - Department of Physics, Wabash College, IN.
Professor James Brown - Department of Physics, Wabash College, IN.

May 2019

Abstract

The topic of ferromagnetism has received great attentions for research and developments in science due to its exotic properties and many applications for generators, transformers, telephones, loudspeakers, and magnetic recording devices. Among the researches for this topic, one of the most fundamental pioneer model comes from the work of the Ising model by Wilhelm Lenz. As the model can be translatable with students that took electrodynamics and thermal physics, this topic has been widely taught and researched by undergraduate faculties and students interested in thermal physics and electrodynamics. One of those undergraduate concepts that we chose for this report is mostly taken from the book "Thermal Physics" by Daniel V. Schroeder. From this book, we aim to explore the ferromagnetism in the Ising Model from the programming and computational aspects. From here, we considered the effectiveness of our coding and matching results with the theoretical model.

1 Introduction

Ferromagnetism has been a widely explored topic in terms of its fundamental behaviors and properties, materials exhibiting ferromagnetism, and applications of those materials in engineering and science. One particular model that was successful in demonstrating the ferromagnetic behavior is the Ising model, named by the physicist Ernst Ising [1]. The model was originally invented by the physicist named Wilhelm Lenz (1920), who gave it as a problem to his student Ernst Ising. The model consists of variables of grids representing a lattice system of atoms, and each variable indicates the magnetic dipole moments of atomic spins that either spinning up or spinning down. These pair of grids act as independent atoms initially, but they will interact with their adjacent atoms and change their own orientations. These behaviors happen under the influences of time evolution and fluctuations in temperatures.

What makes the model interesting is that under a certain temperature, random distribution of atomic spins in a lattice concentrates to form clear clusters of magnetization over a certain amount of time [2]. In another word, the incident of ferromagnetism occurs. For higher temperatures, the lattice distribution shows no evidence of ferromagnetism, as spinning directions continue to be random. Furthermore, under the influence of the magnetic field, the spins in the lattice will align with the direction of the magnetic field to form a complete system of spinning either up or down. Here we can denote spinning up as index 1 and spinning down as index -1. Mathematically, under a certain amount of temperature, a random distribution of variables from -1 to 1 of the grids of lattice will mostly form clusters containing only -1 or 1 over a certain amount of time. Within the addition of magnetic field, the lattice shows mostly variables of either 1 or -1. For larger temperatures, uniform clusters will hardly forms from random distribution. That transitional temperature is called the "Curie Temperature." Under theoretical calculations, the relationship of clustering and the temperature could be understood carefully in terms of probabilistic distributions. Currently, many computational techniques on Ising model, especially the Metropolis algorithm [3], have been extensively used to combine temperature fluctuations from those calculations with the dynamics of atomic spins in very big sizes lattices.

With the rise of modern computational techniques, the model was extensively studied under different spatial dimensions and sizes of lattice. In this report, we aim to discover this model under the Monte Carlo simulation of Metropolis algorithm, with the concept explained under undergraduate concepts of thermal physics and electrodynamics. The concepts are mostly based on [2] and [3]. Specifically for this report, we will focus on discovering the model under the 2-Dimensional constrain. Python 3.7.2 *(Python Software Foundation, https://www.python.org/)* is used due to its ability to handle millions of complex calculations and its familiarity among undergraduate teachings in physics [4]. We discuss our assumptions and procedures on creating the codes for the Ising model. From here, these codes examine and produce the graphs of the dynamic of atomic spins by changing the lattice's size and the temperature. Furthermore, we examine the graph of **correlation** and the distance between two atoms with the graph of correlation length and different temperatures.

2 Theoretical model:

2.1 The idea of ferromagnetism:

Most of these contents are based on [2]. Ferromagnetism is a phenomenon involving the magnetic dipole within the spins of electrons happening in some special materials. If we could magnify these materials and look through a specific region, we could see all the electrons align to each other. But the alignment doesn't necessarily occur in the entire material. In the natural state, the materials will have various regions in which the direction inside each region is randomly oriented.

The question is how does a permanent magnet exist and how could one create it? To create a magnet from random alignment, the common method is to add the external magnetic field. Practically, one would wrap the the coil of wires around the magnetized materials. The equation for this phenomena is:

$$\mathbf{B} = \mu_0 \left(\mathbf{H} + \mathbf{M}\right)$$

where **B** is the external magnetic field created by the induced current, **M** is the magnetization, and **H** is the response of the material from its magnetization and the external magnetic field. In practical usage, **H** is much smaller than **M**, so the induced magnetic field implies magnetization. From this effect, the domains inside the material could align uniformly in only one direction. This phenomena comes from the maintenance on dipole moments, even when after the magnetic field has stopped.

2.2 Temperature influence and the Ising model:

The contents in this section are based on [3]. The alignment of dipole moments across the material could also attribute to the temperature influence. Low temperature enhances the process of magnetization and high temperatures exhibits destruction of magnetization. The transitional temperature from high to low of this phenomenon is called a **critical temperature**.

The model will be based on both [3] and [5]. This phenomenon could be putted into equations by using the Ising model. Consider a Lattice model of $N \times N$ in 2D with N^2 spins, either up or down. Let s_i be the spin variable of the i dipole which double only takes the values $+1$ (↑) and -1 (↓). The Hamiltonian could be written as:

$$\mathbf{H}\left(\{s_i\}\right) = -J \sum_{i,j} s_i s_j - \mathbf{B} \sum_{i} s_i$$

where J is described as the strength of interaction, **B** is the external magnetic field, and the sum $\mathbf{H}\left(\{s_i\}\right)$ is over the adjacent neighbor pairs. Each spin has 4 nearest pairs. Under the work of While Onsager (1968 Nobel Prize in Chemistry), the reciprocal temperature under non-equilibrium thermodynamics is:

$$k_B T_C = \frac{2J}{ln(1+\sqrt{2})} = 2.269 \cdots J$$

And the energy represented is ([3]

$$U = -\epsilon \sum_{\text{neighboringpairs i,j}} s_i s_j$$

If a spin at position $k = i, j$ points up, the interaction between this dipole and its neighbor is:

$$E_{\uparrow} = -\sum_{\text{neighbor}} s_{\text{neighbor}} = \epsilon n \bar{s}$$

where n is 4 in 2-dimension. If the spin goes down, then:

$$E_{\downarrow} = \epsilon n \bar{s}$$

From here, we could calculate the temperature difference when changing a spin $k = (i, j)$ into the opposite direction is:

$$\Delta E(i, j) = \epsilon s_k \sum_{\text{neighbor}} s_{\text{neighbor}}$$

These formulas become useful to explore many algorithm on Monte Carlo simulations to do computations on and dynamics of the Ising model.

3 Procedures

3.1 Ising model on 2D

3.1.1 Metropolis algorithm:

The Metropolis Algorithm could be summarize as the following from [3]:

- Set up a lattice size N and the desired temperature T under consideration
- Set the spins inside the lattice to be one-half up and one-half down under random number 0.5.
- Start with a random state (i, j) in the lattice.
- Define the function $DeltaU(i, j)$ to calculate the temperature difference when changing the direction of the spin at position (i, j).
- Make an iteration loop with the duration $100{*}\text{size}^2$ and choose an arbitrary i and j to calculate the temperature difference. If $DeltaU < 0$, we change the direction of the spin (sign of $s_(i, j)$). If $DeltaU >= 0$, we only change the direction of the spin under the probability $p = e^{-DeltaU/T}$
- Assign the up-spin into one color and down-spin into another color to print out the resulting lattice.

  From the demonstration in [3] about the algorithm, the probability when the energy difference is non-negative effectively maintains the ratio of Boltzmann factor of the two changing states. This means that for any two states 1 and 2, being different by only the spinning direction, the ratio of changing from 1 to 2 and back from 2 to 1 is balanced together and maintains the Boltzmann probabilities. The code for this algorithm is demonstrated in the file *"problem 8.29 - metropolis algorithm.py"*

3.1.2 The correlation function

The correlation function over a distance r is calculated by the formula

$$c(r) = \bar{s_i s_j} - \bar{s_j}^2$$

where s_i and s_j is chosen random over the lattice, and r is the shortest over 9 distances between s_i and s_j by the periodic boundary conditions. To set up the algorithm for the graph, we use the random distribution library *Random* to set up two arbitrary pairs of grids. After we set up the iteration among $100 * \text{size}^2$ then we calculate the shortest distances, with the formulas are demonstrated in the file *"problem 8.29 - the colleration distance.py"*. Then, the part $\bar{s_i s_j}$ of the function and the part $\bar{s_j}^2$ are added over each iteration to then averaging all of those parts together. Finally, we represent the indexes by setting up the command *colors* in the libraries *matplotlib.pyplot* and *matplotlib.*

3.1.3 The correlation and temperature graph

From the Metropolis algorithm and the correlation function, we use those two packages to find out the correlation length of the corresponding temperature. We incorporate those definitions into the file *"isingmodelgraphics.py"* and extract 3 functions inside the new file *"correlation length - temperature.py."* The resulting codes from 3 different temperature ranges are combined in a text file name *"list 0.5 - 3.5.txt."* This is the most extensive piece of computation, as the iteration requires a lot of steps that could take a couple of days to produce the results from a Macbook with Processor 2.9 GHz - Intel Core i7 and Memory of 12 GB - 1600 MHz - DDR3. We process three temperature ranges, one from 0.5 to 1.5, 1.5 and 2.5, and 2.5 and 3.5, with the ranging step is 0.05. In each temperature, we calculate the of the correlation distance function over 10 times the size of the lattice. For each time, we track down the data in the correlation function to obtain the closest lower bond of $1/e$. The final closest number was averaged over 10 times the size.

4 Results

The results of these computational codes are demonstrated as follows:

4.1 Metropolis Algorithm

There are matching transition of the Ising model by temperatures and how they are collated with the theoretical predictions about the magnetization of the entire materials. Here we represent the two following cases:

4.1.1 Without magnetic field:

For size = 400, the lattice dynamics with the corresponding temperature among 2.27, 1.8, 1.5, and 1.00 are shown below from Figure 1 to Figure 4. With

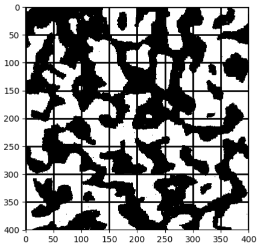

Figure 1: Lattice dynamics for T = 1

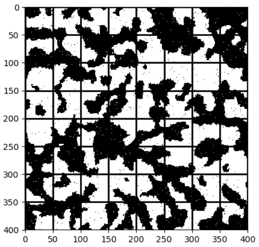

Figure 2: Lattice dynamics for T = 1.5

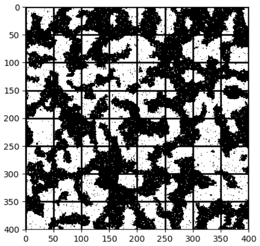

Figure 3: Lattice dynamics for T = 1.8

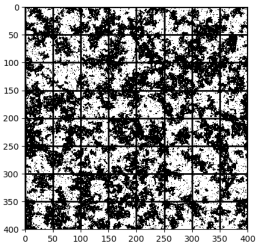

Figure 4: Lattice dynamics for T = 2.27

low temperatures, the clusters inside the lattice are more concrete and mark with specific regions. As the temperature rises, the clusters are more setting apart and begin to transition into random states. These graphics demonstrates the coherency with the model predictions, as lower temperatures show more influences on clustering.

4.2 With the magnetic field

With the magnetic field, we consider the lattice size 400 with the magnetic field B chosen as 1, .2, and $-.2$, and the temperature are chosen from $T = 2$ to $T = 4$. These results are shown from Figure 5 to Figure 8. The result shows that for $T = 4$ and $B = 1$, the lattice still exhibits random distribution of spins. For $T = 2$, the lattice shows strong behaviors of magnetization. For $B = 0.2$, the magnetic field show strong signs of spinning up. For $B = -0.2$, the lattice magnetization aligns with spinning down. These graphs show that the magnetic field does affect in the process of magnetization over all domains inside the materials. Also, low temperatures make the process of magnetization more transparent.

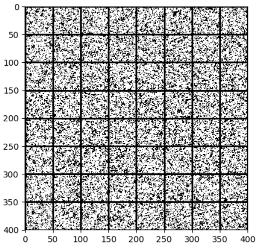

Figure 5: B = 1 and T = 4

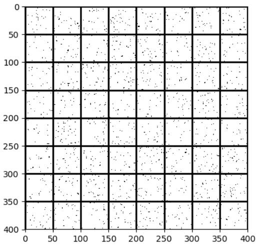

Figure 6: B = 1 and T = 2

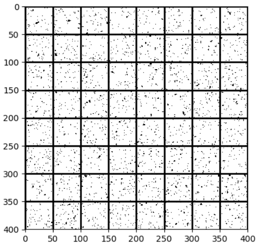

Figure 7: B = 0.2 and T = 2

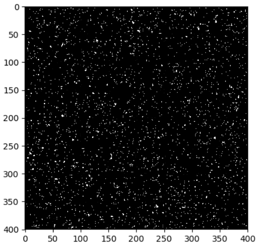

Figure 8: B = - 0.2 and T = 0.2

4.3 Correlation distance graph

The graph is represented in Figure 9, and we see that the graph starts at a very high peak and starts to be exponential decreases to a specific value of distance. What the graph shows makes sense as the smallest length of two grids could not distributed over all possible distances. In the beginning, the graph starts to escalate down quickly as a exponential graph. This behavior seems reasonable with the probability distribution of Metropolis Algorithm. We check if this graphs can give us a consistent information with the correlation length respecting to temperature.

4.4 The correlation length and temperature graph

Producing this graph has been a great challenge to process run, due to its computationally extensive. We wrote the definitions for all the codes in the file named *”isingmodel graphics.py”* At first, we tried to combine the temperature from 0.5 to 3.5. But the running process takes three days without being finished. Thus, we change the approach on the temperature range. We separate the running ranges into 3 categories: 0.5 to 1.5, 1.5 to 2.5, and from 2.5 to 3.5. For each range, the running time only takes from 1.5 hours, thus we were able to obtain the data in only a day. These data is graphed in the file names *”Graph of correlation and distance.py.”* The result is shown in Figure 10. The graph

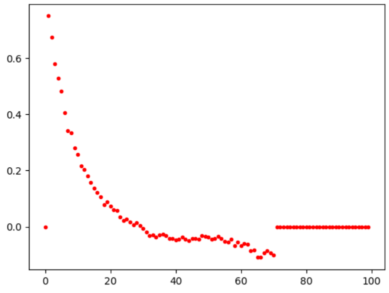

Figure 9: Correlation distance function at T = 2

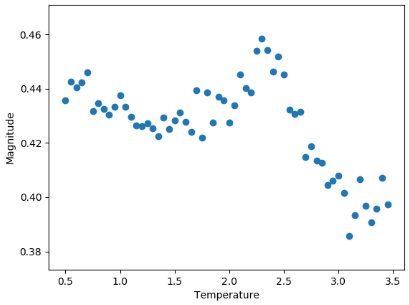

Figure 10: Correlation length and temperature graph for size = 20.

shows the consistent peak with the critical temperature 2.27, with the form of the graph is similar with the peak-graph from [6] and [7].

5 Analysis

We have obtain some consistencies with the original theoretical model. Asides from that, there are rooms for improvement. Based on our Metropolis algorithm's graphs, we could not be certain about the critical temperature. The reason could be during the running process or an unidentified reason.

Furthermore, the graph of correlation – temperature shows the peak very close to 2.27. Comparing with [6] and [7], the curves on both side of the graph shows asymmetrical to each other. This asymmetry could be due to our method of using the lower bond of $\frac{1}{e}$. One improvement could be finding the average of the values between $\frac{1}{e}$ to find a closer approximation.

6 Future work:

For future reference, we could improve the model from considering different factors from coding and from probability distribution of the Monte Carlo simulation based on Metropolis algorithm. Secondly, the improvement of the correlation-temperature graph could make the two curves on both side of the peak more symmetrical. With proper bounds and methods, the graph could be match with the Lorentz peak-uniform distribution, and we could check more carefully about the temperature value of the peak.

This report has been supported by Dr. James Brown, Physics Department at Wabash College, and by the approval of the independent study class named Magnetism in Solid States.

References

[1] Thomas Ising, Reinhard Folk, Ralph Kenna, Bertrand Berche, Yurij Holovatch. The Fate of Ernst Ising and the Fate of his Model. *arXiv preprint, arXiv:1706.01764 [physics.hist-ph], 2017.*

[2] David J. Griffith. Introduction to electrodynamics. *Cambridge University Press, 2017.*

[3] Daniel V. Schroeder. An Introduction to Thermal Physics. *Pearson, 1st edition, 1999.*

[4] Mark Newman, Computational Physics in Python.

[5] Wei Cai. ME346A - Introduction to Statistical Mechanics - Handout 12. Ising Model. *Stanford University, winter 2011 CreateSpace Independent Publishing Platform. http://www-personal.umich.edu/ mejn/computational-physics/, 2012.*

[6] Tom Kennedy. University of Arizona, 2008. The Ising model. *https://www.math.arizona.edu/ tgk/541/chap1.pdf*

[7] Being Critical of Critically in the Brain - Scientific Figure on ResearchGate. Available from:*https://www.researchgate.net/figure/Correlation-length-as-a-function-of-temperature-for-a-simulation-of-the-Ising-Model-Near_fig4_225375861* - May 2019