Monte Carlo Simulation of the 2D Ising Model

Ronan Hanley, Cassie Metzger, & Jeremy Robin

1 Methods

We began this project by creating a header file `isingModel.h` that encapsulates the various functions needed to simulate the 2D Ising Model.

`isingModel.h` takes in variables `nx`, `ny`, `J`, `kbT`, and `H`. These variables represent the grid size (x and y), the coupling constant which determines the strength of the interaction between neighboring spins, the initial temperature as expressed by the equation $E = k_b T$, and the initial magnetization.

Boundary conditions are then enforced using ghost cells where the value of the spin at $x = 0$ is set equal to the value of the spin at $x = Nx - 2$ and the value of the spin at $x = 1$ is set equal to the value of the spin at $x = Nx - 1$. The same conditions are enforced in the y-direction, creating a box of ghost cells, around the lattice.

A method that calculates the energy depending on the point in the lattice is created. This method utilizes the equation:

$$H = -J \sum_{\langle i,j \rangle} s_i s_j \tag{1}$$

In this equation, H represents the Hamiltonian of the system (not the magnetization as `H` does in the header file). By this equation, if all spins are aligned, the energy is minimized and the system is in its lowest state.

A method to calculate the magnetization is also created. This method implements Eq. 9 of the Project 3 guidelines where $\sigma_{i,j}$ represents the sum of all spins in the lattice.

In addition to these three methods that are all referenced in the `isingModel` method, there are 8 additional methods in the `isingModel` class.

The method `run_model` takes in a number of time steps and a filename. This method utilizes the random number generator, `random.h`, which was written in Homework 9. A random integer that is uniformly distributed is generated for i and j, the indices of the lattice. The spin at random point (i, j) is then found and then flipped. As a result of the random flipped spin, the system has a new energy. This can be calculated by summing the energy of neighboring pairs. This is done using the following bit of code:

```
for(int i = 0; i < timeSteps; i++){
[...]
newEnergy += -2 *
lattice[(random_i - 1) +
Nx * random_j] * spin-2.0*H*spin;
newEnergy += -2 *
lattice[(random_i + 1) +
Nx * random_j] * spin-2.0*H*spin;
newEnergy += -2 *
lattice[random_i +
Nx * (random_j - 1)] * spin-2.0*H*spin;
newEnergy += -2 *
lattice[random_i +
Nx * (random_j + 1)] * spin-2.0*H*spin;
}
```

Here, the random i and j variables choose a random point across the lattice. The spins of the neighboring points are then summed and the magnetization is subtracted. The 2 that is placed in front of the energy accounts for the fact that change in energy when a spin is switched goes from $-1 \rightarrow +1$ or $+1 \rightarrow -1$. We then only accept this energy with the following probability:

$$\begin{cases} 1 & \text{if } E_j > E_j \\ e^{-(E_j - E_i)/k_B T} & \text{if } E_i < E_j \end{cases} \tag{2}$$

If the spin is updated, the magnetization is updated accordingly and the boundary conditions are reinforced.

Next, a method for the average energy is created. This method produces a tuple where the first value is the average energy and the second value is the energy variance. Both of these equations come from the project guidelines.

Similarly, a method for calculating the average magnetization is created.

Then, using the values produced by these methods, the average susceptibility and average heat capacity can be calculated.

The magnetic susceptibility can be defined as the response of the system magnetization as the external magnetic field is raised while the heat capacity is the amount of energy required to raise the temperature of the system by one degree.

Finally, the output method produces a `.csv` file depicting the lattice of spins.

2 The 2D Ising Model

We began by running our simulation at $K_b T = 2$ for 5×10^6 steps with the computation for magnetization (`startAverage`) beginning after 1,000,000 steps. We used the random seed `5befcc0e5d25f74f`. The average magnetization was 0.915641 while the average energy was -17513.9 J. The data for this simulation is produced by the file `isingModel.cpp` while the plots are generated by the file `ising_model_first_plots.py`.

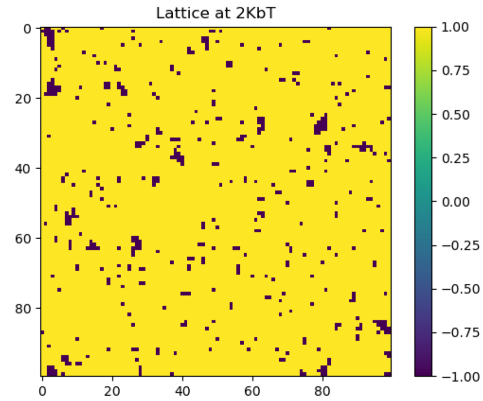

Figure 2.1: Spin lattice when $K_bT = 2$

In this case, we found that equilibrium was reached well before 1×10^6 time steps.

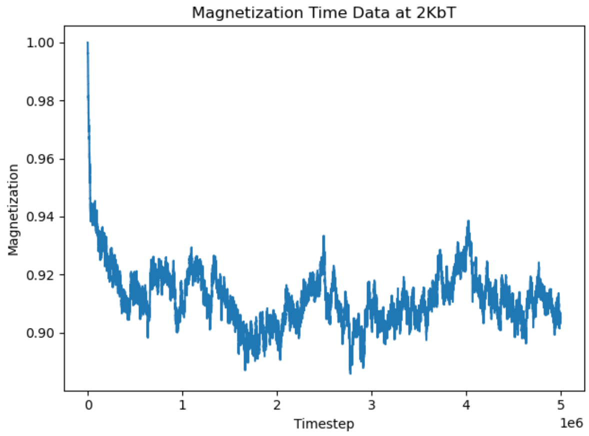

Figure 2.2: The average magnetization value $\langle M \rangle$ over time. This graph depicts the time it took for this situation to reach an equilibrium

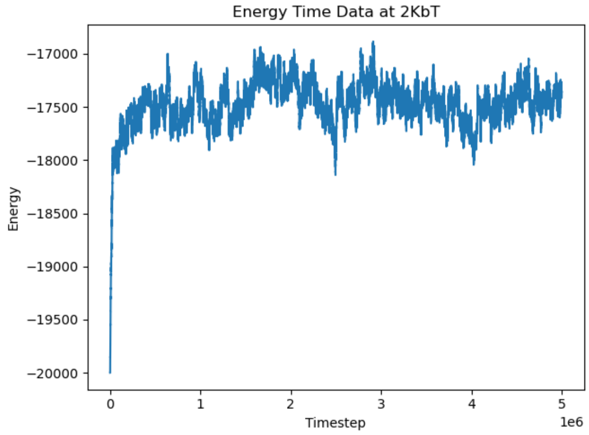

Figure 2.3: The average energy value $\langle E \rangle$ over time. This graph depicts the time it took for this situation to reach an equilibrium

Next, we compared this to an alternate version of this simulation where $k_bT = 3$, again using random seed `7edb6c276105358c`. Here, we found the average magnetization to be -0.00780649 and the average energy to be -8203.94 J.

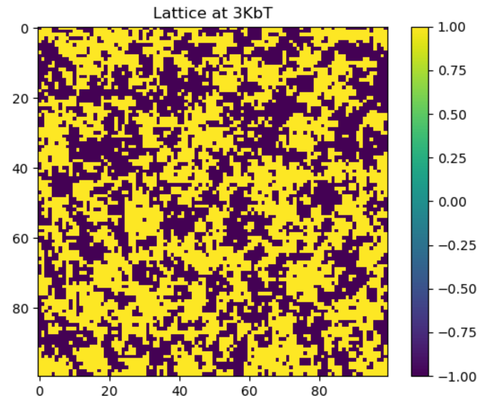

Figure 2.4: Spin lattice when $K_bT = 3$

We found equilibrium to be reached a bit later than in Figures 2.2 and 2.3. The energy and magnetization also experienced fewer fluctuations after equilibrium was reached at a higher temperature.

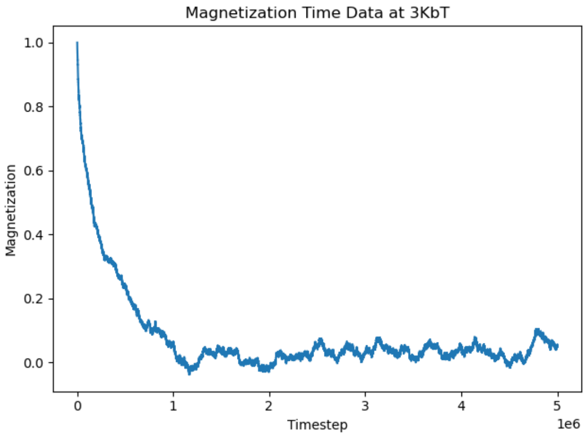

Figure 2.5: The average magnetization value $\langle M \rangle$ over time. This graph depicts the time it took for this situation to reach an equilibrium

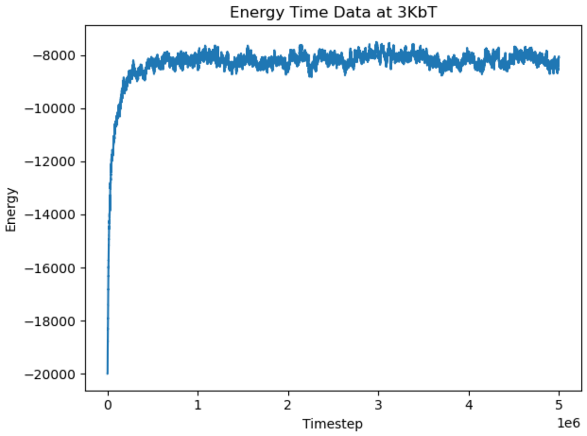

Figure 2.6: The average energy value $\langle E \rangle$ over time. This graph depicts the time it took for this situation to reach an equilibrium

These plots conclude that at higher temperatures, there is the most diversity in the spins of the lattice. The average magnetization and energy across the lattice are lower for higher energies and take a bit longer to reach their equilibrium values. However, once the system is in equilibrium, its temperature and magnetization experience less variance.

3 Phase Transition

To see phase transitions of the simulated magnetic material we varied the temperature from 0 to 4 k_bT with over increments of 0.01. To begin, we ran each model for 5,000,000 time steps and took the last 4,000,000 time steps to find the averages and variance in the data. We looked at the absolute value of the magnetic moment, heat capacity, and magnetic susceptibility. This was done using the file `isingModel_changeTemp.cpp`
The heat capacity is based on the variance in the energy when reaching equilibrium (Eq. 4 in the project overview). At the critical temperature, the states are more likely to change, and therefore the energy is more variable. On the other hand, the magnetic susceptibility is based on the variance of the average magnetic moment. We expected this to change in a similar way to the heat capacity again because the spins will be more likely to change during a phase transition, therefore causing the magnetic moment to be more variable. We were able to approximate the critical temperature for our plots by finding the highest value for C_v and χ and then determining the corresponding temperature. This method was not effective in finding the critical temperature on the average magnetization graph. The following plots were produced by the file `ising_model_over_time.py`

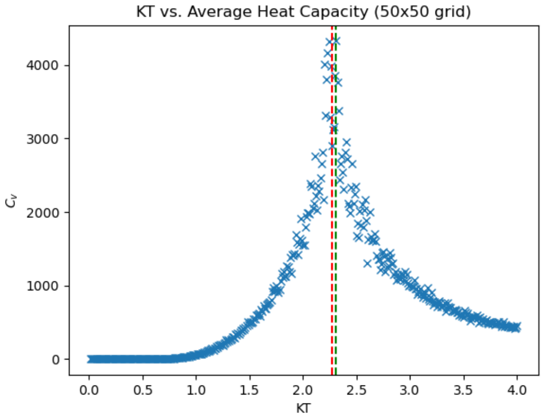

Figure 3.1: The phase transition of the heat capacity (C_v) over a varying temperature. The critical temperature at which the phase transition is expected to occur is plotted in red while the actual phase transition is plotted in green.

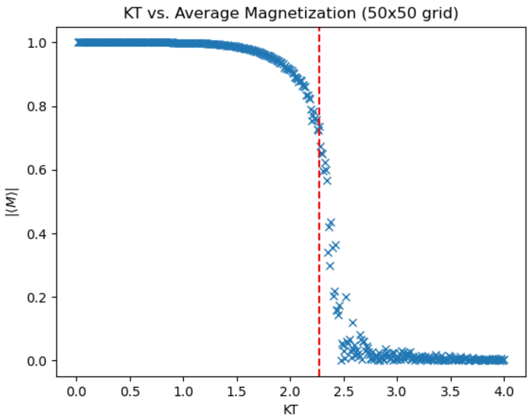

Figure 3.2: The phase transition of the magnetization ($|\langle M \rangle|$) over a varying temperature. We don't calculate the phase change for this graph given that it is more difficult and less likely to be accurate for $|\langle M \rangle|$

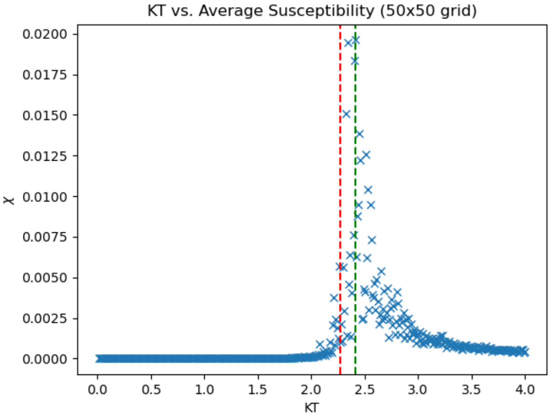

Figure 3.3: The phase transition of the magnetic susceptibility (χ) over a varying temperature

Using the percent error formula, we found the difference between our calculated value of the temperature at which phase transition occurs and the known value, $T_c = 2.269$ J.

Percent error for susceptibility	7.095636844424858 %
Percent error for heat capacity	3.1291317761128132%

3.1 Evaluating different constraints:

We then tested different constraints including the grid size and the time step after which the average value is taken to see how they would affect the phase transitions. We used a percent error calculation as a measurement of accuracy. To find possible theoretical values of our critical temperature we found the highest point on the susceptibility plot vs temperature and on the heat capacity vs temperature and found their corresponding temperatures. We then analyzed the percentage difference between these values and the analytical solution.
We first tried different values for where we started averaging the values of magnetization and energy. We

tried at step 0, 100, 1000, 5,000, 2,000,000, 3,000,000, and 4,000,000. However, due to the random nature of MCMC, the variability between trials is large enough that it is hard to see a concrete trend in either parameter that we varied. Hypothetically, we'd expect that the best place to start would be taking the average as close to right after the steady state was reached as possible. That way, we take the average of as many values as possible without also including values that occur before the steady state is reached. You can see from the graphs in Section 2 that this point is somewhere between 0 and 1 million time steps. But, interestingly, our trials seemed to indicate the percent error for C_v is lowest after 2 million time steps while the percent error for χ is lowest below 0.5 million steps.
Then, we attempted to perform the same procedure to relate the grid size to the error of C_v and χ. We tried grid sizes 5x5-41x41. However, again, we did not find a clear observable relationship.

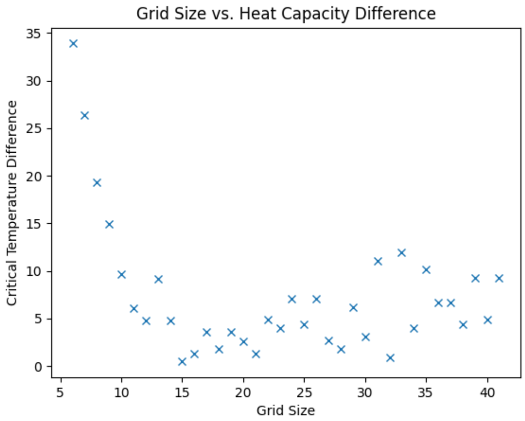

Figure 3.4: We tested various grid sizes ranging from 5x5 to 41x41 and plotted the relationship between the percent difference between our value and the expected value for the heat capacity $(C_v$

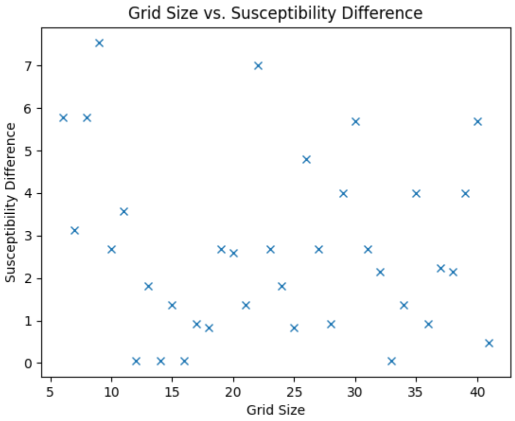

Figure 3.5: We tested various grid sizes ranging from 5x5 to 41x41 and plotted the relationship between the percent difference between our value and the expected value for the susceptibility (χ)

Our results from figure 3.4 suggest that anything beyond a grid size of 15x15 does not improve our results, while figure 3.5 shows no relationship whatsoever.
Using the results of our trials, we find that a grid size of 15 x 15 is optimal. We also find that beginning to calculate the error after 2 million steps for C_v is optimal while calculating it almost immediately for χ is ideal.

4 Magnetic Hysteresis

We implemented magnetic hysteresis in the file `hysteresis.cpp` by adding a permanent magnetic field component to the Hamiltonian.

$$H = -J \sum_{\langle i,j \rangle} s_i s_j - H \sum_{\langle i \rangle} s_i \qquad (3)$$

The hysteresis effect occurs when an external magnetic field that was originally very strong is gradually decreased to 0 and then increased in the other direction. In this case, the magnetic field of the system is hesitant to switch and only flips its magnetization when the external field reaches a certain magnetization in its new direction. To see this, as opposed to waiting for our system to reach an equilibrium as we did in Section 2, we need to slowly change the magnetic field as we compute more and more time steps. We chose to run the process for 5000 steps and then increment the temperature by 0.01 when sweeping the external magnetic field from 3 to -3 and -0.01 when again sweeping from -3 to 3. This means we are going over 6,000,000 total time steps (5000 × 12 × 100). We computed the average values after the first 3000 terms. Then, we observed the effect of magnetic hysteresis over 3 separate temperatures: 1.5, 2, 2.5, and 3 k_bT. We noted that as the temperature decreased, the effect of magnetic hysteresis increased. This could be seen as the delay that results from switching the direction of the magnetic field increased as temperature decreased. The following figures were produced by the file `hysteresis.py`.

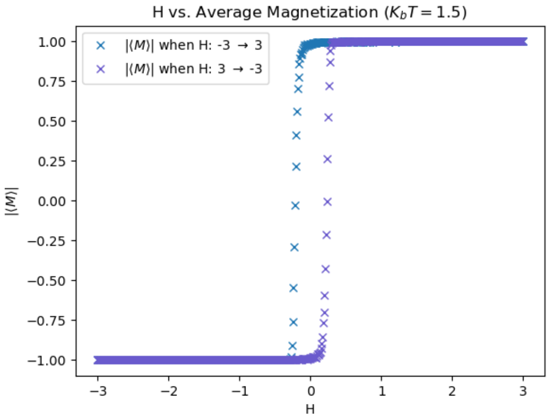

Figure 4.1: Hysteresis Effect at $k_bT = 1.5$

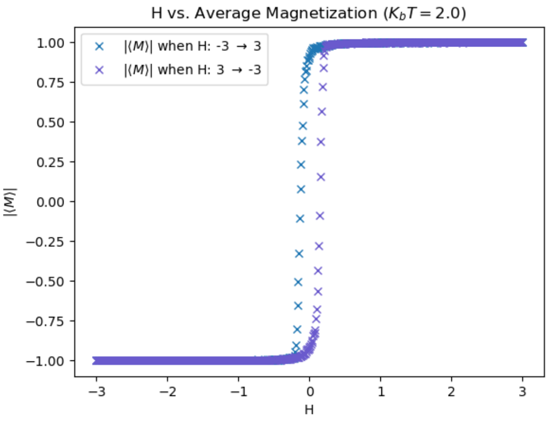

Figure 4.2: Hysteresis Effect at $k_bT = 2.0$

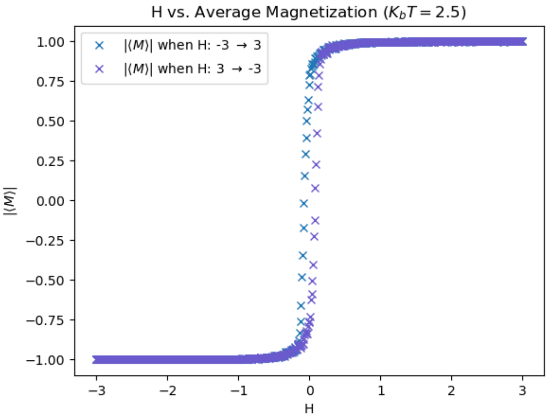

Figure 4.3: Hysteresis Effect at $k_bT = 2.5$

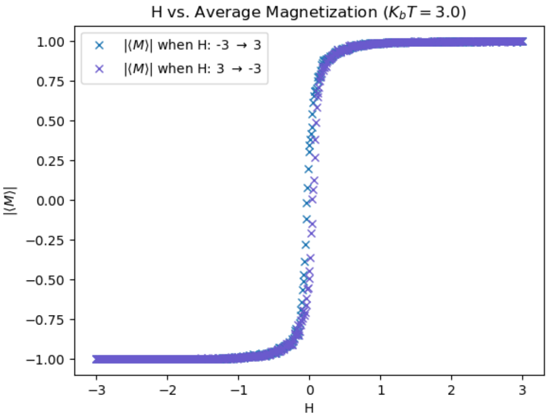

Figure 4.4: Hysteresis Effect at $k_bT = 3.0$

As stated, these plots show that the difference between varying H from -3 to 3 and 3 to -3 increases as the temperature decreases. It appears that at higher temperatures, it is either easier to reach the external magnetization required to flip the internal magnetization or the internal system is more responsive to the external magnetization.

4.1 Changing the H range:

In order to test this, we held all other variables the same, leaving $K_bT = 1.5$, the number of steps $=$ $5,000$, the step number at which the average begins to be calculated $= 3,000$. We began with an H range of $[-10, 10]$.

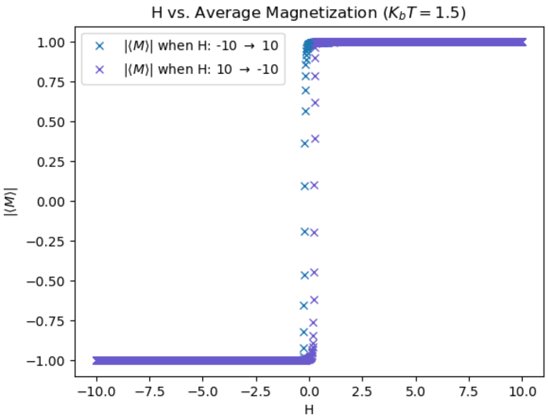

Figure 4.5: Hysteresis Effect over a longer range with $k_bT = 1.5$,

A similar test done with a range of $[-100, 100]$ yielded the similar resulting range.

Next, we tested how reducing the H-range would affect the hysteresis effect. We ran the simulation using $K_bT = 1.0$ and decreased the range of H to -0.13 to 0.13, which in previous trials was the range where the jump in average magnetization happened. What we can see is that there is no splitting and only the top part of the hysteresis curve is shown.

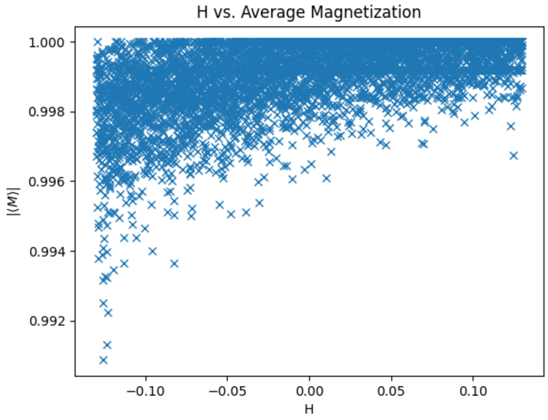

Figure 4.6: Hysteresis Effect over small range

Therefore, we conclude that the hysteresis effect is more prominent over lower temperatures.