2D Ising Model for Ferromagnetism

Ben McMurtry

18th January 2016

1 Introduction

The Ising model[1] for Ferromagnetism was invented by Wilhelm Lenz, and named after physicist Ernst Ising who solved the 1-dimensional case in his thesis[2] in 1924. It is a statistical mechanics model of discrete atomic spins, that can be in either the up or down state, which are arranged into a lattice, with periodic boundary conditions, and allowing each spin to interact only with it's nearest neighbours, as shown (for the 2-Dimensional case) in Figure 1.

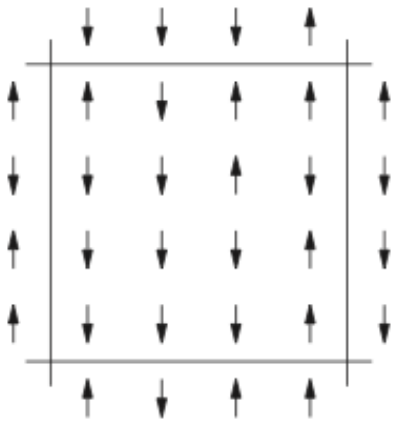

Figure 1: A square lattice of spins, with periodic boundary conditions, with nearest neighbours considered to be in the 4 cardinal directions

The 2-dimensional Ising Model is one of the simplest statistical models to show a phase transtition, consistent with observations, and is also notable for having non-trivial interactions, yet having an analytical solution. The two-dimensional Ising model was solved by Lars Onsager[3] in 1944.

A program was written to simulate a ferromagnet using the 2-D Ising Model, with the ability to evolve the system in time using a Monte Carlo Metropolis algorithm[4].

The Metropolis algorithm is used to evolve the system in time until equilibrium is reached (which is considered to be when macroscopic properties of the system are approximately constant).

Once equilibrium is found, the program calculates the macroscopic properties of interest: total lattice energy E, and magnetisation per particle $\mathbf{M}$, and then continues to record these values for a number of subsequent time evolutions, intended to resemble thermal fluctuations.

Once enough recordings have been made at equilibrium, the program calculates thermodynamic averages for the mean total lattice energy $\langle E \rangle$, magnetisation per particle $\langle \mathbf{M} \rangle$, and uses those to calculate the specific heat capacity C, and susceptibility χ.

The macroscopic properties at equilibrium for that temperature are then printed to a file, and then the simulation moves on the to the next temperature step and starts again trying to find equilibrium with the Metropolis algorithm.

2 Method

2.1 Ising model

The two-dimensional ferromagnet with N sites, was initialised as a $D \times D$ square array of integers. Then each site in sequence was populated randomly with a +1 or -1. Since the aim is to approximate an infinite lattice, periodic boundary conditions were used (in 2D this can be visualised as a torus), in the following form:

```
#define S(x,y)      S[(D+(x))%D][(D+(y))%D]
```

The energy of a finite section of the square lattice in the Ising model is given by:

$$E = -J \sum_{i,j=nn(i)}^{N} \mathbf{s}_i \mathbf{s}_j - \mu_0 \sum_{i=1}^{N} \mathbf{B} \cdot \mathbf{s}_i \tag{1}$$

where the first sum in equation (1) represents the energy of interaction of spins, and is over all nearest neighbours. J denotes the energy exchange constant, which is a measure of the strength of the interaction between nearest neighbour spins. For ferromagnets, $J > 0$, signifying that nearest neighbour spins which point in the same direction are energetically favourable, the opposite is true for antiferromagnets ($J < 0$).

The average magnetization per particle of a $D \times D$ lattice is defined as the average of the spins at each site:

$$\mathbf{M} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{s}_i \tag{2}$$

The Heat Capacity and Susceptibility are defined in the normal way:

$$C = \frac{\delta}{\delta T}\langle E\rangle = \frac{1}{k_B T^2}(\langle E^2\rangle - \langle E\rangle^2) \tag{3}$$

$$\chi = \frac{1}{k_B T}(\langle M^2\rangle - \langle M\rangle^2) \tag{4}$$

2.2 Monte Carlo Metropolis algorithm

According to statistical mechanics, the probability of finding the lattice in a state with energy E is

$$P(E) = \frac{e^{-E/(k_B T)}}{\sum_Z e^{-E_Z/(k_B T)}}, \tag{5}$$

where k_B is Boltzmann's constant.

Because there are N sites, each with two possible states (spin UP or DOWN), there are 2^N possible states of the model magnet. This can lead to exceptionally large numbers of possible configurations for what might seem like a relatively small lattice (for example in a 10×10 lattice, there would be about 1.26×10^{30} possible configurations). It would not be feasible to sample all of these configurations, so random sampling must be used.

The code uses the standard Metopolis Monte Carlo algorithm (attempting to minimise the energy), in which a series of trial flips are attempted. A spin will flip if the resulting ΔE is 0 or negative or if a randomly generated number between 0 and 1 is less than the Boltzmann distribution $e^{-\frac{\Delta E}{kT}}$. If $\Delta E \leq 0$ then $e^{-\frac{\Delta E}{kT}} \leq 1$ and hence will always result in flipped spin, so only this second condition needs to be tested.

For the sake of computation time, so that the system equilibrates faster, the metropolis algorithm has been implemented to go through the lattice in sequence, as oppposed to randomly, which seems to be a more realistic time evolution method for the lattice[5].

2.3 Code Structure

For the purposes of the program, all properties were calculated in units where both J and k_B are equal to 1.

Two structures are defined near the top of the program:

1. The struct eqbm_sum contains a sum of the E, E^2, M and M^2, along with a counter, to which values are added every timestep after equilibrium has been found. All of which are used in calculating the thermal macroscopic properties. according to the equations in section 2.1.

2. The struct thermal_properties holds all the calculated macroscopic properties for a given temperature for the purpose of printing.

For the case of deciding when equilibrium had been reached, I decided to look at the change in energy, since it is easy to calculate, and from early investigations it was clear that the energy of the system followed an exponential decay.

The program takes samples of the energy every N timesteps and compares the proportional change from the last timestep to an input threshold value. To avoid accidental triggers due to random fluctuations, this check must be completed twice in a row for equilibrium to be recognised.

Once equilibrium has been achieved macroscopic values are calculated. The heat capacity and susceptibility calculations require calculating the time variance of the energy and magnetisation respectively at equilibrium.

A key difficulty in the design of good algorithms for this problem is that formulas for the variance may involve sums of squares, which can lead to numerical instability as well as to arithmetic overflow (including catastrophic cancellation) when dealing with large values[6]. To avoid this, a shifting value for the energy and magnetisation was used to keep the variance unchanged, but maintain precision, and stability. When equilibrium is reached the value for the energy and magnetisation are stored as energy_shift and mag_shift. These values are subtracted from all energy values used in the variance calculation to reduce the magnitude of the energy and magnetisation, resulting in greater retained precision. These values are summed and stored in the eqbm_sums_s struct for every temperature step.

3 Results

3.1 Theory

The analytic solution derived by Onsager in 1944[3] predicts a critical Temperature T_C, at which a phase transition occurs . At this temperature the system undergoes an order to disorder transition as spontaneous magnetisation falls to zero. For an infinite lattice, this critical temperature should follow the relation[1]:

$$T_C = \frac{2J}{k \ln\left(1+\sqrt{2}\right)} \approx 2.269 \; for \; J = k = 1 \tag{6}$$

And we expect the magnetisation, which takes the place of the order parameter, to follow:[7]:

$$M(T) = (1 - 2\sinh\left(\frac{2}{T}\right)^{-4})^{\frac{1}{8}} \tag{7}$$

As such, when plotting thermal properties against temperature (without the presence of a magnetic field), we expect to see:

1. The magnetisation of the sample go immediately to zero at the critical temperature.

2. Energy to have a sharp step at the critical temperature, as the spins go from a spin aligned, minimum ground state energy, to a disordered maximum energy above the critical temperature.

3. The heat capacity is the temperature derivative of the mean energy; hence we expect the heat capacity to have a sharp spike and diverge to infinity at the critical temperature and be zero elsewhere.

4. The magnetic susceptibility measures how much the magnetisation changes with respect to temperature, hence we expect a similar shape to the heat capacity, with a divergence to infinity at the critical temperature and a value of zero elsewhere.

Onsager also used critical exponents to characterise the behaviour of second order phase transitions very close to the critical temperature. Taking a power series about the critical temperature and neglecting high order terms we find that the macroscopic properties should behave as follows:

$$C \propto (T - T_C)^{\alpha} \qquad M \propto (T - T_C)^{\beta} \qquad \chi \propto (T - T_C)^{\gamma} \tag{8}$$

With $\alpha = 0$, $\beta = \frac{1}{8}$, $\gamma = \frac{-7}{4}$.
These values are tested in the following section.

3.2 Simulation Results

The output of a normal run of the program is shown in Figure 2, with the parameters used in the caption.

Overall there are two clear differences from the theoretical predictions given in the previous section.

The first difference is that the magnetisation does not sharply rise/fall to a value of zero, and stay there, but instead shows fluctuations of decreasing amplitude around zero after T_C. This could be explained by local spin alignment (domains) still arising above T_C, causing residual magnetisation.

As the temperature gets higher these local magnetisations become smaller as thermal excitation randomly distributes the spins. This is likely an issue since we are using a finite, periodic simulation of the lattice, resulting in local clumping not completely cancelling out as it should on an infinite plane.

The second difference is that the phase transition appears to be stretched in all plots. This may be due to the size of the temperature steps, and the residual magnetisation acting in the opposite direction to the phase transition (retaining a memory of its previous form), stretching the transition.

These two differences likely are the reason for the susceptibility not being a perfectly divergent spike at the critical temperature. The smoothening to the right of the susceptibility curve is most likely due to the residual magnetisation.

The energy plot is smoothed and stretched, compared to the expected step shape, most likely due to the finite nature of the simulation lattice. An infinite lattice would be completely disordered, but a finite section of lattice may hold

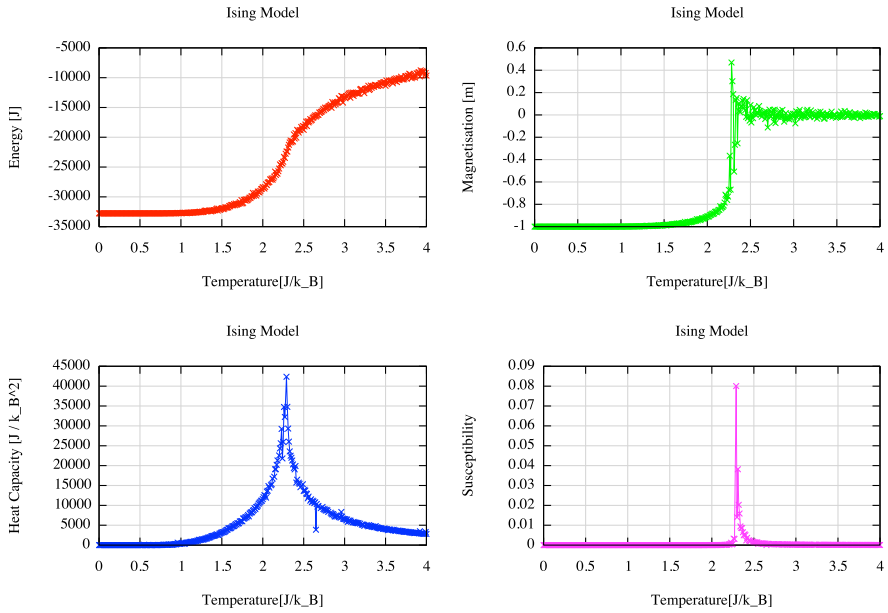

Figure 2: Results of running the program with a 128x128 lattice and 0.01 Temperature steps (takes approx 1 hour on a 2012 macbook.)

local domains of aligned spins resulting in a slower transition to the disordered energy state.

The stretched nature of the energy plot results in the heat capacity not being a perfect spike but rather a curve. This result is expected and well documented[8], with larger lattice sizes theoretically having sharper peaks.

3.3 Critical Exponents

Using Matlab's curve fitting tool[9], to fit the data points before T_C ($T = 1.78 - 2.27$) for magnetisation to $M = K \times (T - T_C)^{\frac{1}{8}}$, gave the graph shown in Figure 3, showing good agreement with theory, close to T_C, with K = 1.071.

Fitting the data in the same range for Heat Capacity did not work with the critical exponent α set to 0, which implies heat capacity should be a constant. The best fit that was found was using $C = K \times (T - T_C)^{\frac{-1}{2}}$, which gave the graph shown in Figure 4, with K = 1224.

Fitting the data in a larger range for Susceptibility using $\chi = K \times (T - T_C)^{\frac{-7}{4}}$, gave the graph shown in Figure 5, showing fair agreement with theory, with K $= 7 \times 10^{-7}$.

Overall, the critical exponents from the simulation, show fair agreement with the theory, but would be more easily analysed by more advanced methods, such as the finite-size scaling which is based on finding values for macroscopic

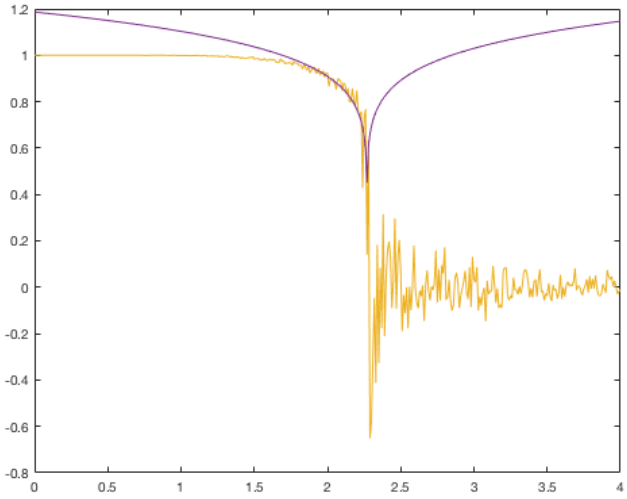

Figure 3: Simulation results for Magnetisation vs temperature, fitted with purple curve with critical exponent of 1/8

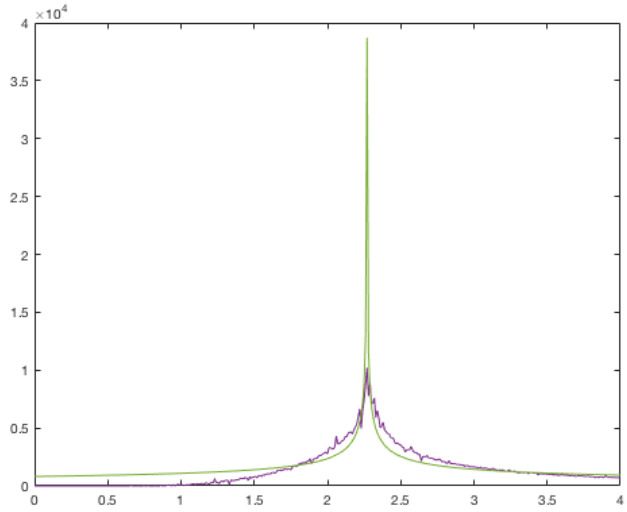

Figure 4: Simulation results for Heat Capacity vs temperature, fitted with purple curve with critical exponent of -1/2

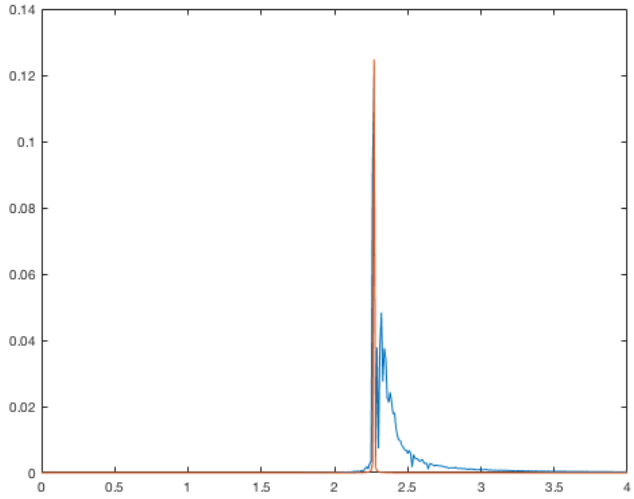

Figure 5: Simulation results for Heat Capacity vs temperature, fitted with purple curve with critical exponent of -7/4

properties at different lattice sizes and linearly fitting these results to find values for the critical exponents. Given more time on this project, that is one of the most important improvements to be made.

Other interesting projects, that could be done with the use of this program, could be to investigate the Ising model for non-zero magnetic fields, or even for anti-ferromagnetic materials.

4 Conclusions

This program can accurately, and efficiently simulate a 2-D ferromagnet using the Ising model, and produce numerical and graphical results for the magnet's macroscopic properties (E, $\mathbf{M}$, C and χ) over a range of temperatures, including the critical temperature.

The simulation results show good agreement with the theory provided by Onsager, and others. However, the fitting of the critical exponents requires more sophisticated methods, and investigation.

References

[1] `https://en.wikipedia.org/wiki/Ising_model`

[2] *Ernst Ising - Contribution to the Theory of Ferromagnetism (1924)*

[3] *L. Onsager, Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition Phys. Rev. 65 117 (1944)*

[4] `https://en.wikipedia.org/wiki/MetropolisHastings_algorithm`

[5] *H. Gould, J. Tobochnik, and W. Christian - An Introduction to Computer Simulation Methods: Applications to Physical Systems - Chapter 17 Monte Carlo Simulation of the Canonical Ensemble Addison-Wesley (2006)*

[6] `https://en.wikipedia.org/wiki/Algorithms_for_calculating_variance`

[7] `http://www.hiskp.uni-bonn.de/uploads/media/ising_II.pdf`

[8] `http://www.physik.uni-leipzig.de/~janke/Paper/spinmc.pdf`

[9] `https://uk.mathworks.com/help/curvefit/curvefitting-app.html`