Metropolis Algorithm for Ising Model

Abstract

The following report details an investigation into the properties of the 2D Ising Model, representing a ferromagnetic material, using the Metropolis Algorithm, coded using Python. During the course of the investigation, the evolution of the lattice as it magnetised was visualised. The magnetisation was plotted against time for various different times, in order to investigate the effect of T_c on the magnetisation of the lattice. The system was seen to be unable to magnetise for temperatures in excess of T_c.

Introduction and Methods

Ising Model

The Ising Model is a mathematical model used to replicate the behaviour of ferromagnets, by simulating the interaction of magnetic spins. The model is named after Ernst Ising, who solved the one-dimensional model in 1925. The 2D square-lattice model, which is used in this investigation, was solved for the special case of the external magnetic field, *H*, being equal to zero, by Lars Onsager in 1944. This square lattice model is one of the most basic statistical models which exhibits a phase transition.

The model consists of a square lattice, where each point on the lattice is given a value of either +1 or -1, signifying an “up” or “down” spin respectively. As long-range spin interactions are negligible, each spin interacts only with its four nearest neighbours, i.e. above, below, right, and left. The Hamiltonian for the system, with no external magnetic field is given by:

$$H = -J \sum S_i . S_j$$

Where *J* is the interaction energy of the pair of adjacent spins *i* and *j*, with each of the spins being counted once. For $J > 0$ the interaction is ferromagnetic, so the tendency for adjacent spins to be the same sign will be higher. For $J < 0$ the interaction is antiferromagnetic, so adjacent spins will tend to be of opposite sign, and for $J = 0$ the spins are noninteracting.

The boundary conditions are given such that the surface is continuous, where left of the left edge lattice point will give the rightmost lattice point. This holds vice-versa and also for up and down.

In the lattice, as time goes on, if a spin-flip is more energetically favourable, it will generally occur. Less energetically favourable flips may also occur, with the following probability:

$$P_{flip} = e^{-\beta \Delta E}$$

Where $\beta = 1/K_BT$ and ΔE is the change is energy resulting from the flip. Given the dependency on temperature, at higher temperatures, these spontaneous flips are more likely, leading to a longer magnetisation time, if the system magnetises at all. There is a critical temperature T_c, at which the system exhibits a phase transition, and above which the system will not magnetise, over any length of time. Below this temperature, the spins will eventually align and magnetisation will occur. This critical temperature, for the 2D, square lattice Ising Model, was found to be:

$$T_c = \frac{2}{ln(1 + \sqrt{2})} J/k_B$$

Metropolis Algorithm

The Metropolis Algorithm is a Monte Carlo method for obtaining samples from a given probability distribution. Given the random nature of the flipping spins in a ferromagnetic material, the Ising Model is very well suited to investigation via the Metropolis Algorithm. The random nature also lends itself very well to a computer simulation, where pseudo-random numbers can be generated with ease, and large numbers of iterations can be completed relatively quickly and efficiently.

An iteration of the algorithm would firstly, in this case, consist of randomly choosing a random point on the lattice. The energy cost of a flip is computed, and if it is negative, the spin is immediately flipped. If the change in energy is positive, the spin is flipped with the probability P_{flip}, seen above. This process continues and the system evolves, with a particular dependence on temperature, which is again outlined above

Results

Visualisation of Lattice

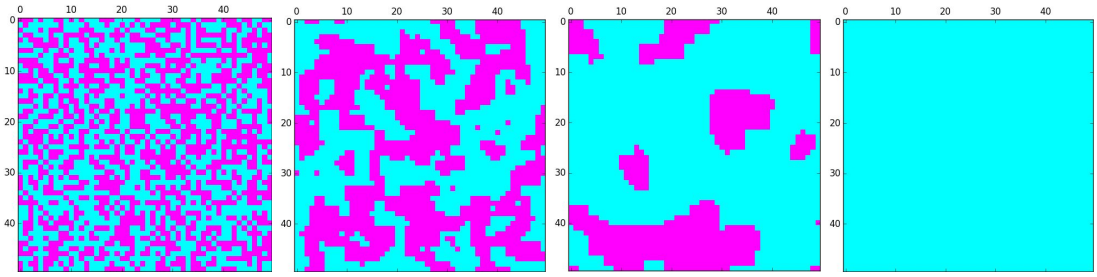

Figure 1: Process of magnetisation in a 50x50 lattice, with T << T_c

In Fig.1 above, a visual representation of the Ising Model can be seen, colour coded for easier visualisation. In this case, blue represents spins of +1, and pink represents spins of -1. This reaching of equilibrium occurs over the course of 1,000,000 iterations, for a 50x50 lattice, with temperature $T << T_c$, the critical temperature. As the simulation runs, domains of

-1 and +1 spins can be seen to form, and as time goes on, shrink or grow as appropriate, until one domain spans the entire lattice. At this point, the system is fully magnetised.

For temperatures above the critical temperature, the system will not magnetise, regardless of the number of iterations. The randomly flipping spins are simply too numerous in comparison to the more energetically favourable flips.

Data and Analysis

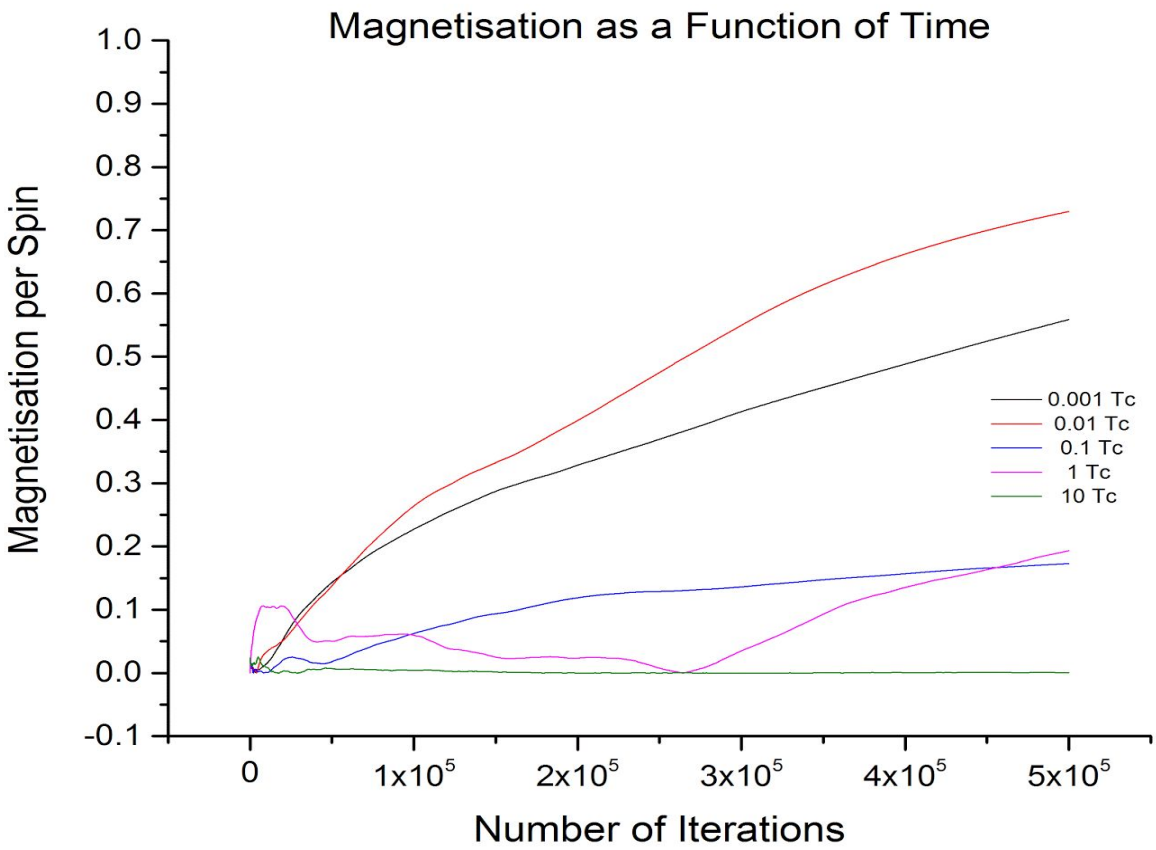

Figure 2: Graph of Magnetisation over time for a 40x40 lattice, at varying temperatures

The above graph, Fig. 2, illustrates how the magnetisation of the system evolves over time. The y-axis is Magnetisation per Spin, which gives an asymptote at 1, meaning full magnetisation. The x-axis is the number of iterations run through by the simulation, which is very much analogous to time.

For those temperatures well below T_c, the magnetisation per spin can be seen to increase towards 1 quite quickly. However, as temperature reaches the critical temperature, the speed of magnetisation slows down, and once $T = T_c$ the system will not magnetise. For temperatures well in excess of T_c, it can be seen that the magnetisation per spin rarely strays very far from zero. This is due to the continuous, random flipping of spins throughout the lattice, preventing any domains from proliferating.

Conclusions

The investigation was overall a success, as both the Ising Model and the Metropolis worked as intended, and I myself feel much more confident using Python, and would be much more likely to use it in the future, where previously I would have sought other methods. I also felt the assignment was a good opportunity for self-directed learning, and actively seeking new ways of doing things.

I was disappointed in the final code, however, for reasons both in and outside my control, I did not spend as much time as I would have liked on the assignment, as as such some very important, and relatively straightforward results were not obtained. These results include energy per spin, specific heat, and magnetic susceptibility, and their dependence on temperature. These graphs would have shown the phase change that was the real aim of this investigation.

References

- Dr. Thomas Archer, PY3C01, Computational Methods.
- Hammel, Ben. "Monte Carlo Simulation of the Ising Model using Python". TheBrokendesk.com.
- Prof. Stefan Hutzler, PY3C01, Computer Simulations I - Numerical and Statistical Methods.