Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

PY3C01 Assignment:
A Python Implementation of the 2D Ising Model via Metropolis Algorithm

Dillon O'Rourke-15317753 (orourkdi@tcd.ie)
Junior Sophister Physics
Trinity College Dublin
January 2018

1 Abstract

In this exercise python was used to implement a two dimensional Ising model which allowed for the mathematical modelling of a ferromagnet. This model was implemented via the Metropolis algorithm which allowed for the evolution of the system over time. The evolution of this system is illustrated with images of the 2D square lattice of spins being updated by the algorithm. The updated spins formed domains and eventually relaxed into equilibrium with the average magnetisation becoming ± 1 below the Curie point which was found to be $T_C = 2.30 \pm 5$. The number of steps required for a system of specific dimensions was also investigated. A 50×50 matrix was found to require $\approx 900,000$ iterations of the algorithm.

2 Introduction and Theory

2.1 Statistical Mechanics

Statistical Mechanics is the branch of theoretical physics that combines probability theory with the laws of classical and quantum mechanics in order to describe and predict the properties and behaviours of materials on the macroscopic level based on the microscopic constituents of the materials. Take thermal energy as an example, the statistical mechanic approach is to interpret thermal energy as the sum of the energy of each individual particle in disordered states and temperatures.

Consider a system described as free particles in box which is in a heat bath. This system is described by its microstates s and their energy values E_s the system will take on the temperature of the heat bath after a suitable amount of time. The probability p_s of a microstate with an energy value (E_s) is represented by the following;

$$p_s = \frac{1}{Z} e^{-\beta E_s}$$

Where $\beta = \frac{1}{k_B T}$ and Z is a normalisation called the partition sum and is given by;

$$Z = \Sigma^{\Omega}_{s=1} e^{k_B T}$$

To compute the state of a system, one would need to evaluate the partition sum for every possible state of the system of which there are too many. Take for example a spin system with N spins, it contains 2^N states and takes $2N \times 2^N$ steps to calculate the possible energy values.

2.2 The Ising Model

The Ising model is essentially a simple mathematical model of a ferromagnet. It consists of a finite square lattice with discrete $[-1, +1]$ values assigned to each lattice point. These values represent the magnetic dipole moments of atomic spins at the site (spin up or spin down). The 2D square-lattice Ising model is one of the best methods that shows a phase transition. For each lattice site, there is a variable σ_k that takes only one of two possible values $\{-1, +1\}$. There are interactions between every adjacent site such that the interaction between site i and j is equal to J_{ij} which is also called the coupling constant and an external magnetic field h. The Hamiltonian which gives the energy of a configuration is expressed by;

$$H(\sigma) = -\Sigma_{<i,j>} J_{ij} \sigma_i \sigma_j - h \Sigma_j \sigma_j$$

2.3 The Metropolis Algorithm

The idea of the metropolis is to have a procedure that can generate a sequence of states such that they relax into thermal equilibrium over time. Essentially the algorithm picks a random lattice point $[i, j]$ and evaluates the Hamiltonian as is, E_1, and then calculates the Hamiltonian after the spin is flipped, E_2. It accepts this spin flip if either the energy of the new state is lower than

the initial one or it meets a probability condition that comes from Maxwell Boltzmann statistics given by;

$$p = e^{\frac{\Delta E}{k_B T}} \quad \text{where } \Delta E = E_1 - E_2$$

A number X between 0 and 1 is generated randomly and if $X \leq p$ then the condition is met and the spin is flipped even though ΔE is > 0. This process is repeated for a specified number of iterations and the new updated lattice will be returned after the process is complete. A flowchart is provided explaining more intuitively how this algorithm works.

2.4 Magnetisation

The average magnetisation determined by summing up the spin of every lattice point and calculating the average. A plot of the average magnetisation vs the number of iterations of the metropolis algorithm should be seen to approach a constant value at some point. This constant value represents the number of steps of the algorithm required for the system to reach equilibrium. The average magnetisation per unit spin can be expressed by;

$$< M > \; = \; \frac{1}{N^2} \Sigma_{(i,j)} \sigma_{ij}$$

3 Procedure

3.1 Implementation of the Metropolis algorithm

The first objective was to create a two dimensional square lattice of spins. To do this, a matrix was created with variable dimensions $(N \times N)$. This matrix was to represent the two dimensional square lattice with a particle assigned to every lattice point. Each lattice point (matrix entry) was randomly assigned a value of either plus or minus one which represented the spin of the particle at that point.

A suitable value of N would require an extremely costly amount of computational power and time. So to keep N at a reasonable value that could be worked with while increasing accuracy, periodic boundary conditions were implemented.

To simulate a non-finite lattice periodic boundary conditions were implemented essentially the lattice wraps back around on itself. Pythons modulo operator was used to add periodic boundary conditions by making the neighbouring sites of the random site chosen equal to “((i-1)%N, j)” as opposed to ((i-1), j) for example.

Now the algorithm decides whether a spin flip is energetically favourable. First the method picks a lattice point at random and the energy change after the flip, (ΔE), must be calculated according to the Hamiltonian. If the energy change is less than zero then the spin is flipped. If it is greater than zero it is flipped if it meets the following condition;

- A random number X in the interval [0, 1] is generated

- Is $X < e^{-\frac{\Delta E}{k_B T}}$?

If the condition is not met then the spin is not flipped. The new matrix is returned after a number of n iterations of this algorithm. See below for a flowchart explaining how this algorithm works;

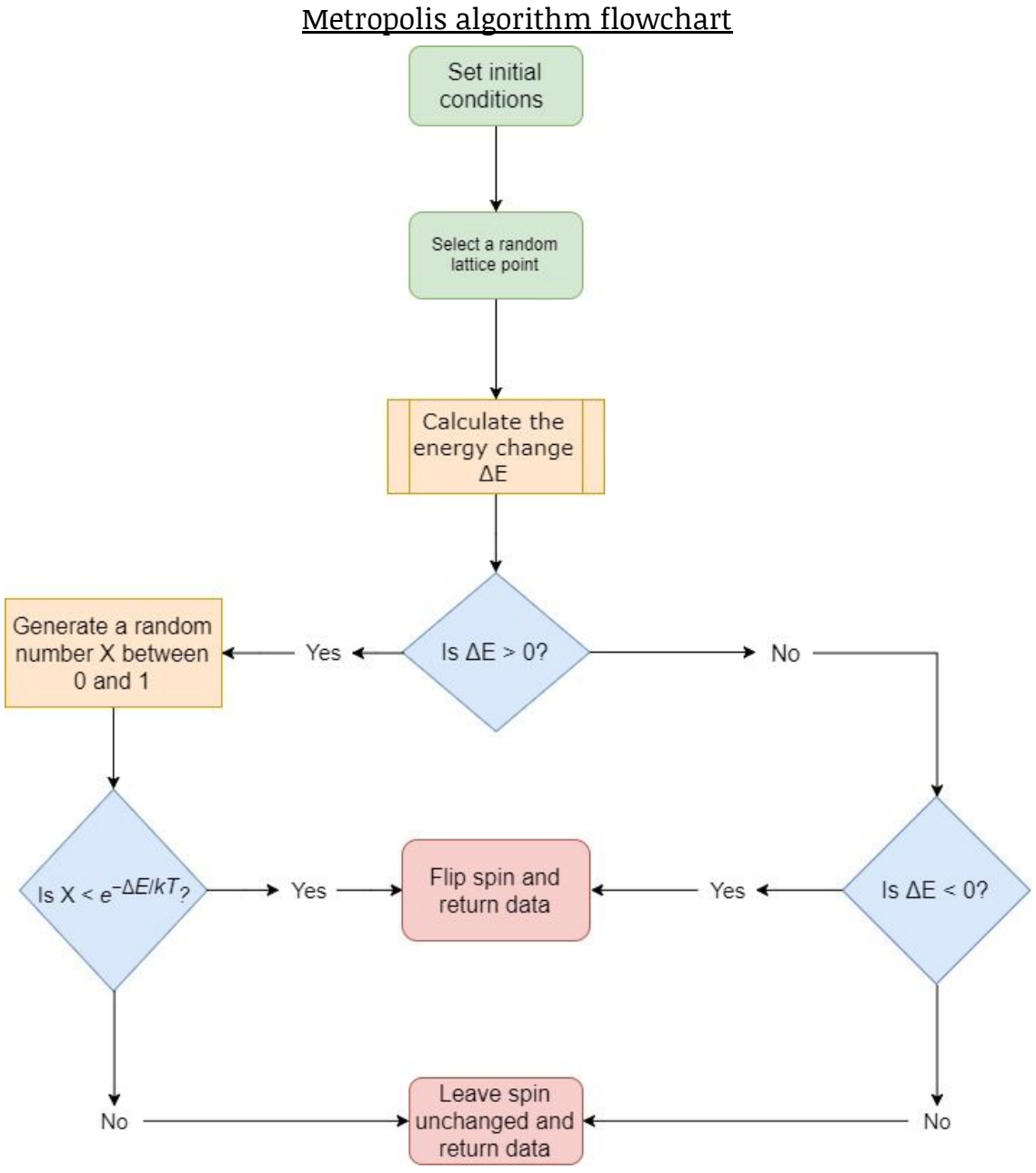
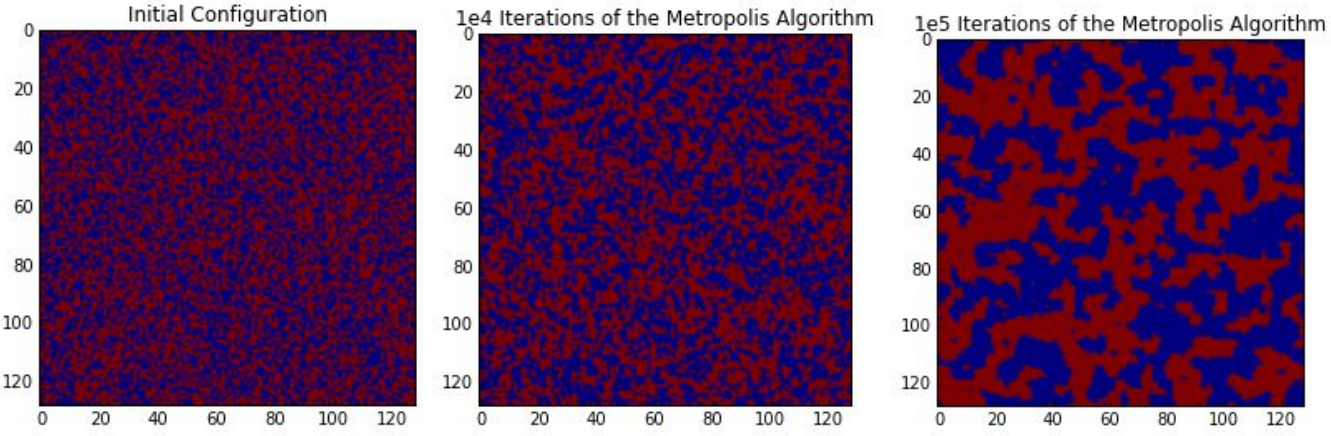


Fig.1) Flowchart for the Metropolis algorithm which explains the steps taken to update the state of the system

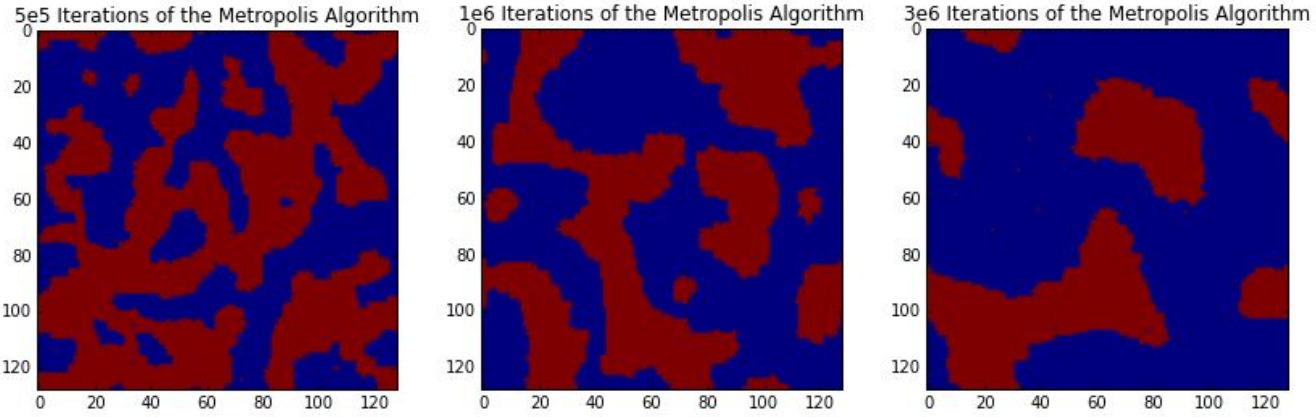
4 Results & Discussion

A 2D Ising model using the metropolis algorithm was successfully implemented. It ran through each point in the lattice, updating the spins based on the energy change in the flip (ΔE), and the probability it would flip, p, even if energetically unfavourable.

Images illustrating the evolution of the system over time



Figures 2-4) Initial configuration, after $1e4$ *iterations where* $1e4 = 10^4$ *or and after* $1e5$ *iterations*

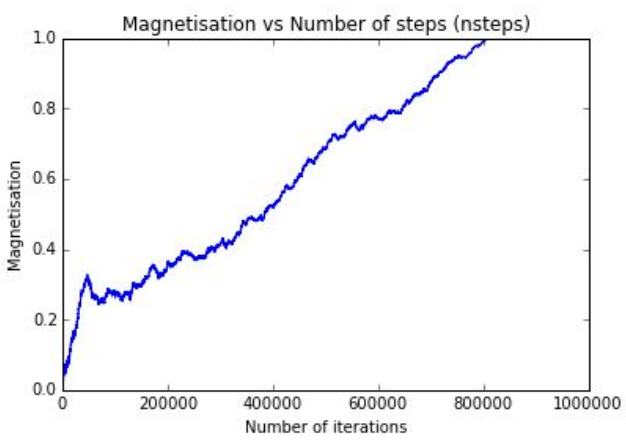


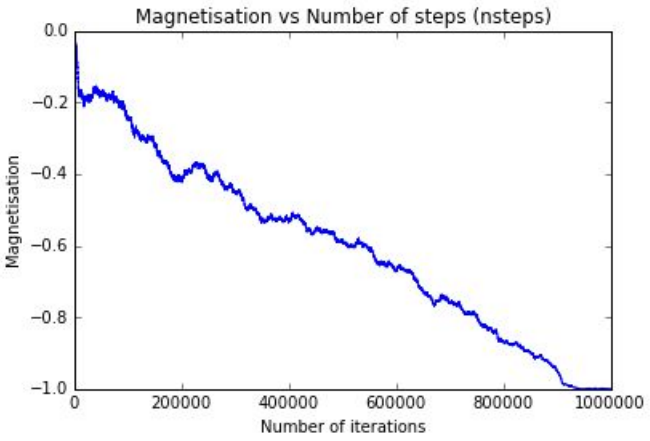
Figures 4-6) Show the increasing amount of spin flips as the number of iterations increase.

As the system updates, one can see that the spins begin to form domains which increase in size over time. The effects of the periodic boundary conditions can also be seen in the borders of the images. The system is approaching equilibrium but is not at equilibrium even after 3 million iterations. Implementing the average magnetisation per spin and plotting it against the number of iterations can be used to determine the number of iteration required for a system of a given size to reach equilibrium.

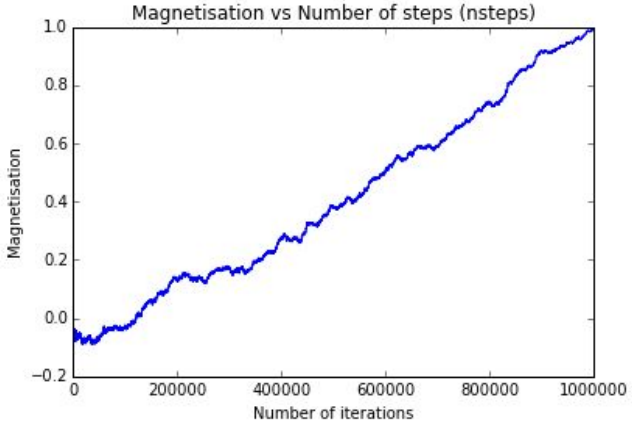
One issue that had arisen was that iterations $\geq$ 10,000,000 would cause a memory error in python so the size of the matrix was decreased to 50×50. The number of iterations required for this specific system to reach equilibrium was $\approx 900,000$ which was the average of a the following graphs.

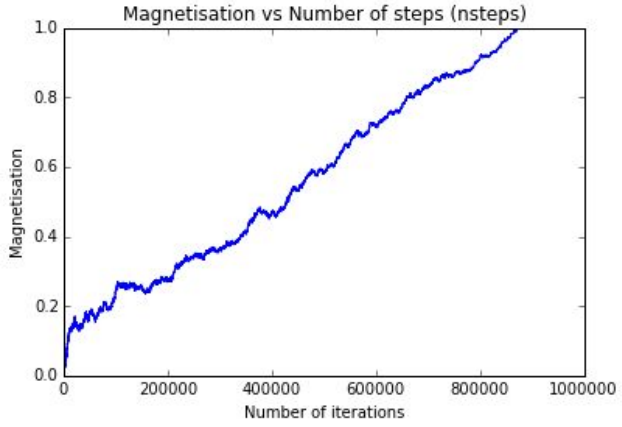
Graphs of Magnetisation vs number of steps





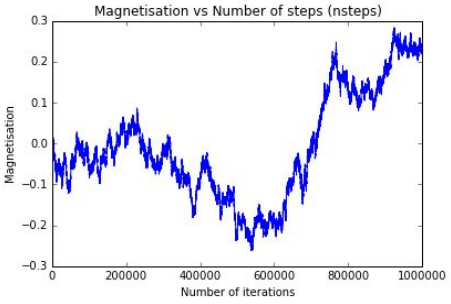
Figures 7&8) These are two graphs with the same parameters that show how to system reaches equilibrium after a number of steps. The dimensions of the matrix are 50×50

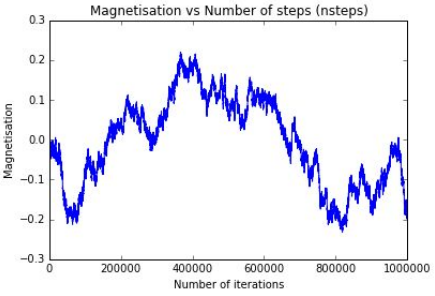




Figures 9&10) Show the same as 7&8

This graph was plotted for a range of temperatures between 0 and 4 which showed that the system no longer came to equilibrium with the average magnetisation equal to neither one or minus one as T approaches a value of 2.3 even for iterations > 900000. This indicated a phase transition occurs at that point.





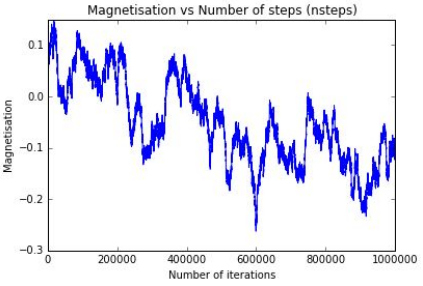


Fig.11) At T=2.25 *Fig.12) At T=2.30* *Fig.13) At T=2.35*

As T is increased the plot of magnetisation vs number of iterations seems to require more steps to reach equilibrium until it begins to look like it will no longer do so. This occurs somewhere around $T_C \approx 2.3$

5 Conclusion

A 2D Ising model was successfully implemented via the metropolis algorithm. The metropolis algorithm was shown to be able to update the system correctly. This can be seen in figures two to six which illustrate how the system evolves over time as the metropolis algorithm updates the spins of lattice points based on the steps outlined in the flowchart provided. It was evident that even for this method, large scale systems would be impractical due to the large number of iterations of the algorithm required. To reduce computation times, smaller size systems were used. To retain some accuracy using smaller systems, periodic boundary conditions were implemented.

The system comes to equilibrium with the average magnetisation $=\pm 1$ which indicates that is a ferromagnetic system. However a phase transition was identified at $T_C = 2.3 \pm 0.5$ which is when the system is no longer seen to reach equilibrium with the average magnetisation $=\pm 1$.

Increasing the dimensions of the system or using an average of multiple simulations could be ways of increasing the accuracy of the method used in this exercise. However both would come at the cost of much longer computation times.

6 Bibliography

[1] S. Hutlzer, PY3C01 – Computer Simulations I, Numerical and Statistical Methods, School of Physics, Trinity College Dublin

[2] Franz Mandl, *Statistical Physics*, 2nd Edition, Wiley, May 12, 1988

[3] Kittel and Kroemer, *Thermal Physics*, 2nd Edition, W. H. Freeman, (25 Mar. 1980)