

Spring 2021
PHYS 377 Advanced Computational Physics
HW # 9a

Problem 1: The Ising Model

The **Ising model** is a theoretical model of a magnet. The magnetization of a magnetic material is made up of the combination of many small magnetic dipoles spread throughout the material. If these dipoles point in random directions then the overall magnetization of the system will be close to zero, but if they line up so that all or most of them point in the same direction then the system can acquire a macroscopic magnetic moment—it becomes magnetized. The Ising model is a model of this process in which the individual moments are represented by dipoles or “spins” arranged on a grid or lattice:

↑	↓	↑	↑	↓	↑	↓	↓
↑	↓	↑	↓	↓	↓	↑	↓
↓	↑	↓	↑	↓	↓	↑	↑
↑	↑	↑	↓	↑	↓	↓	↑
↓	↑	↑	↓	↑	↑	↓	↓
↓	↓	↑	↑	↑	↑	↓	↓
↓	↓	↑	↓	↑	↓	↑	↓
↑	↑	↓	↑	↓	↑	↑	↓

In this case, we are using a *square lattice in two dimensions*, although the model can be defined in principle for any lattice in any number of dimensions.

The spins themselves, in this simple model, are restricted to point in only two directions, up and down. Mathematically the spins are represented by variables $S_i = \pm 1$ on the points of the lattice, +1 for up-pointing spins and −1 for down-pointing ones. Dipoles in real magnets can typically point in any spatial direction, not just up or down, but the Ising model, with its restriction to just the two directions, captures a lot of the important physics while being significantly simpler to understand.

Another important feature of many magnetic materials is that the individual dipoles in the material may interact magnetically in such a way that it is energetically favorable for them to line up in the same direction. The magnetic potential energy due to the interaction of two dipoles is proportional

to their dot product, but in the Ising model this simplifies to just the product $s_i s_j$ for spins on sites i and j of the lattice, since the spins are one-dimensional scalars, not vectors. Then the actual energy of interaction is $-J s_i s_j$, where J is a positive interaction constant. The minus sign ensures that the interactions are *ferromagnetic*, meaning the energy is lower when dipoles are lined up. A ferromagnetic interaction implies that the material will magnetize if given the chance. (In some materials the interaction has the opposite sign so that the dipoles prefer to be antialigned. Such a material is said to be *antiferromagnetic*, but we will not look at the antiferromagnetic case here.)

Normally it is assumed that spins interact only with those that are immediately adjacent to them on the lattice, which gives a total energy for the entire system equal to

$$E = -J \sum_{\langle ij \rangle} s_i s_j$$

where the notation $\langle ij \rangle$ indicates a sum over pairs i, j that are adjacent on the lattice. On the square lattice we use in this exercise each spin has four adjacent neighbors with which it interacts.

Write a program to perform a *Markov chain Monte Carlo* (MCMC) simulation of the Ising model on the square lattice for a system of 20×20 spins. You will need to set up variables to hold the value ± 1 of the spin on each lattice site, probably using a two-dimensional integer array, and then take the following steps.

(a) First **write a function** to calculate the **total energy** of the system, as given by the equation above. That is, for a given array of values of the spins, go through every pair of adjacent spins and add up the contributions $s_i s_j$ from all of them, then multiply by $-J$.

Hint 1: Each unique pair of adjacent spins crops up only once in the sum. Thus, there is a term $-J s_1 s_2$ if spins 1 and 2 are adjacent to one another, but you do not also need a term $-J s_2 s_1$.

Hint 2: To make your final program to run in a reasonable amount of time, you will find it helpful if you can work out a way to calculate the energy using Python's ability to do arithmetic with entire arrays at once. If you do the calculation step by step, your program will be significantly slower.

(b) Now use your function as the basis for a **Metropolis-style** simulation of the Ising model with $J = 1$ and temperature $T = 1$ in units where the Boltzmann constant k_B is also 1. Initially set the spin variables randomly to ± 1 , so that on average about a half of them are up and a half down, giving a total magnetization of roughly zero. Then choose a spin at random, flip it, and calculate the new energy after it is flipped, and hence also the change in energy as a result of the flip. Then decide whether to *accept the flip* using the **Metropolis acceptance formula**. If the move is *rejected* you will have to flip the spin back to where it was. Otherwise you keep the flipped spin. Now repeat this process for many moves.

(c) **Make a plot** of the total magnetization $M = \sum_i s_i$ of the system as a function of time for a million Monte Carlo steps. You should see that the system develops a “spontaneous magnetization,” a nonzero value of the overall magnetization.

Hint: While you are working on your program, do shorter runs, of maybe ten thousand steps at a time. Once you have it working properly, do a longer run of a million steps to get the final results.

(d) Run your program several times and observe the sign of the magnetization that develops, positive or negative. Describe what you find and give a brief explanation of what is happening.