54.5+54+5=113.5

113.5/117

Homework 11 Write-Up

Petra Budavari*
Haverford College Department of Physics
(Dated: May 2, 2024)

1. THE ISING MODEL 10.9

$$E = -J \sum_{<ij>} sisj \tag{1}$$

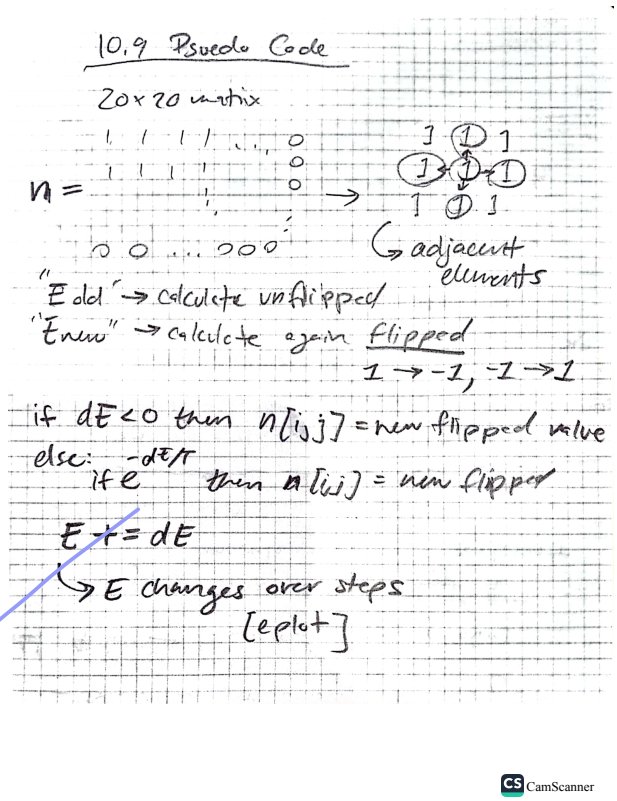

FIG. 1: Pseudo code for Exercise 10.9 Newman.

This exercise helps us explore the magnetization of magnetic material by modeling a matrix of small magnetic dipoles that make up of the material. This theoretical model of magnetization is called the Ising model. If the dipoles are pointed in random directions then the magnetization is approximately zero, but when dipoles point in the same direction the material is magnetized.

In a simple model, one dipole (s_i) can only point in one of two directions indicated by -1 and +1.

The total energy of the system is calculated by Equation 1.

In our model, we used a 20 by 20 matrix with dipoles pointing in direction 1 or -1. Then the energy of the area around the dipole is calculates before and after the dipole is flipped. Because to calculate the energy of the system you use the main element and also the elements directly adjacent to that element, you "pad" the matrix with a column and row of zeros on the sides to make sure that the code does not try to index a non existent element. Using code, we choose whether to accept or undo each dipole flip based on the change in energy. If the difference in energy is less than zero (E new is greater) than the flip is excepted, otherwise, the it is determined by dE and T, somewhat due to random chance.

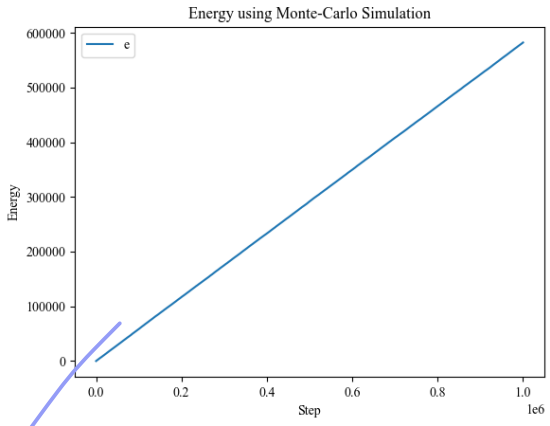

FIG. 2: Energy over steps in Ising model.

In the first method, the initial matrix is made up of ones only and elements are flipped to -1 (and back) from there. The results plot of the energies throughout the steps is Figure 2. This looks different to what I expected - I expected the graph to look more like Figure 3 from the example code for Ising models - however, I am not sure how the code is wrong.

*Electronic address: pbudavari@haverford

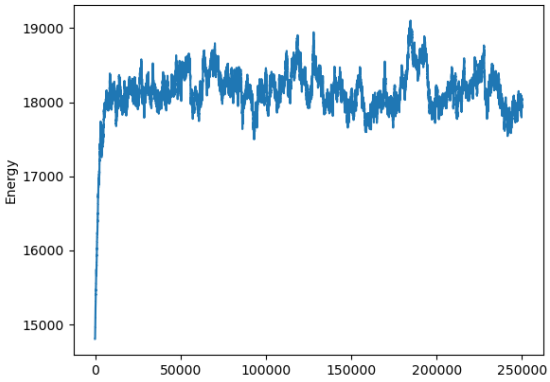

FIG. 3: Expected energy behavior.

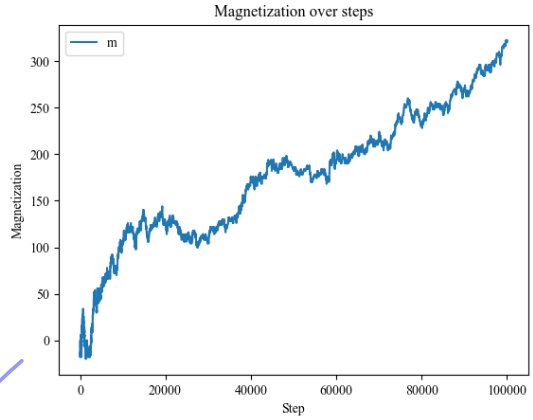

FIG. 5: Magnetization over steps.

In the second method, we used a 20 by 20 matrix with dipoles but with a Metropolis-style simulation, meaning that in the initial matrix the dipoles are already randomized.Elements at random are chosen to be flipped. Then, like in the previous model, flips are either accepted or rejected based on the change in energy of the system.

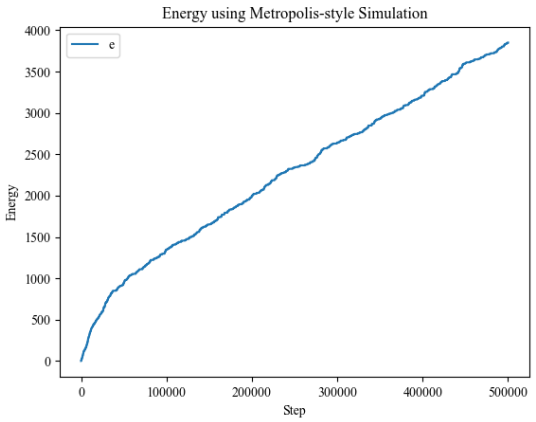

FIG. 4: Energy over steps using Metropolis-style.

We are also asked to plot the magnetization (calculated by Equation 2) as a function of the steps/flips. We expect that the magnetization will become to near a non-zero value. The result of the magnetization calculation is seen in Figure 5 which shows the total magnetization over the flips/steps.

$$M = \sum_i si \tag{2}$$

We are asked to run the plot over 1 million steps.However, for some reason this took a very long time to run on my laptop so, I opted for a shorter step number.

The magnetization starts at zero because when the dipoles are spun randomly we can assume that there are about equal number of dipoles in the 1 and the -1 direction. As dipoles are flipped, the magnetization goes to a non-zero value. This value changes each time I run the code; it can be negative or positive This is because there is no external field. If there were, it would near the same value every time.

We are then asked to show the flips in the matrix of dipoles to visualize it in an animation. **The animation I submitted is uploaded in GitHub.** In the animation you can see that "islands" of dipole-directions are formed because of the nature of the acceptance requirements of the code. I expected that as the temperature increases then so does the amount of flips per set of steps - possibly because heat gives energy to atoms.

One frame of the animation is shown below in Figure 6 to display how the matrix is represented. One color represents -1 dipoles and the other represents 1 dipoles.

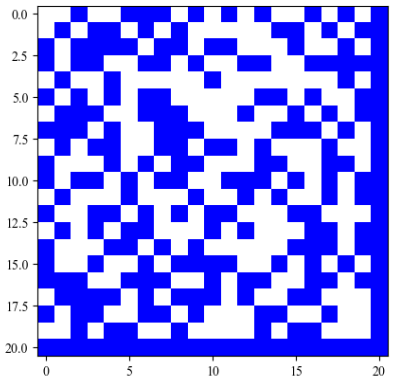

FIG. 6: Matrix plot showing dipole directions.

2. THE XY MODEL

The XY model is very similar to an Ising model however, instead of the dipoles only being able to point in two directions, in the XY model they can point in any direction in the XY field. Thus, dipole directions are indicated by some angle θ between 0 and 2π.

Hence, the new energy equation in Equation 3.

$$E = -Jcos(\theta_i - \theta_j) \tag{3}$$

The code is also very similar to that of the previous Ising model, but again, the range of motion of the dipoles was changes to 0 to $2pi$. To implement this the initial values of matrix elements now randomly range from $cos(\theta)$ where theta is a value from 0 to $2pi$ and each flip is a new theta value.

Again, the resulting plot (Fig. 7 for energy over steps at T=0.2 does not seem quite right... But, it is increasing which is a good sign. And it looks similar to the Ising method.

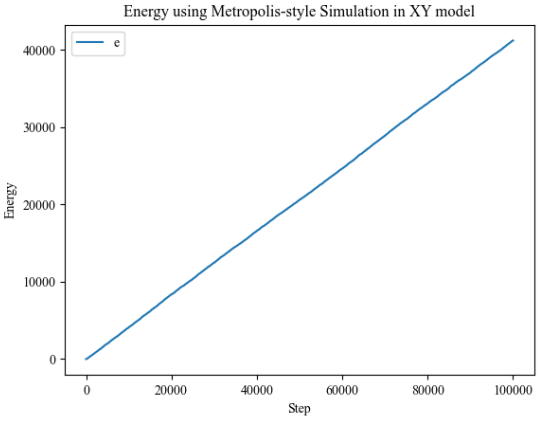

FIG. 7: Energy over steps in XY model.

When we plot the energy over time, we get seemingly random behavior. I think this is because there is no external field. If there were, I would expect the energy to decreases as temperature increases because if temperature is higher than the molecules move around faster and dipoles are more likely to flip randomly or to reach the max energy faster (more flips).

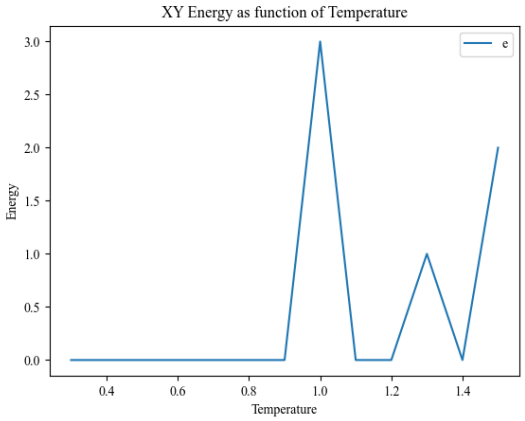

FIG. 8: Calculated energy over temperature.

3. SURVEY QUESTIONS

The homework this week took approximately 14 hours. While, I understood the Ising model, the coding was difficult and I had trouble getting the animation to work. I think this would have been more fun and less time-consuming assignment if the example code was more developed or had better comments. I do feel like it helped me learn about the Ising model.

10.9 54.5/66

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

3 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

save fig error -1

2 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points)

3 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points)

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

3.5 6. Is the output answer correct? (+4 points).

↳ small error updating n -0.5

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

---

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

3 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)

✓ . Is a clear, legible LaTeX type-set write up handed in?

3 . Are key figures and numbers from the problem given? (+ 3 points)

4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)

3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)

2 . Is a brief explanation of physical context given? (+2 points)

1 . If relevant, are helpful analytic scalings or known solutions given? (+1 point)

3 . Is the algorithm used explicitly stated and justified? (+3 points)

2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

2 . Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)

1 . Are collaborators clearly acknowledged? (+1 point)

2 . Are any outside references appropriately cited? (+2 point)

54/56

Computational Physics/Astrophysics, Winter 2024:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 56 points will be available per problem. Partial credit available on all non-1 items.

4 1. Does the program complete without crashing in a reasonable time frame? (+4 points)

2 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? (+2 points)

3 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) (+3 points)

5 4. Is the algorithm appropriate for the problem? If a specific algorithm was requested in the prompt, was it used? (+5 points)

4 5. If relevant, were proper parameters/choices made for a numerically converged answer? (+4 points)

→ errors updating n -2

2 6. Is the output answer correct? (+4 points).

→ energies need to be divided by 20·20

3 7. Is the code readable? (+3 points)

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

3 8. Is the code well documented? (+3 points)

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

9. Write-up (up to 28 points)

5 . Is the problem-solving approach clearly indicated through a flow-chart, pseudo-code, or other appropriate schematic? (+5 points)

✓ . Is a clear, legible LaTeX type-set write up handed in?

3 . Are key figures and numbers from the problem given? (+ 3 points)

4 . Do figures and or tables have captions/legends/units clearly indicated. (+ 4 points)

3 . Do figures have a sufficient number of points to infer the claimed/desired trends? (+ 3 points)

2 . Is a brief explanation of physical context given? (+2 points)

1 . If relevant, are helpful analytic scalings or known solutions given? (+1 point)

3 . Is the algorithm used explicitly stated and justified? (+3 points)

2 . When relevant, are numerical errors/convergence justified/shown/explained? (+2 points)

2. Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (+2 points)
1. Are collaborators clearly acknowledged? (+1 point)
1. Are any outside references appropriately cited? (+2 point)