Simulating the Potts Model

Florencia Nardone, Carter Chapman, and Dev Singh
Department of Physics and Astronomy, The University of North Carolina at Chapel Hill
(Physics 332, Group 15)
(Dated: December 2024)

This paper reviews the development and application of a Potts model simulation created by our group, focusing on its consistency and expected behavior. Our findings validate the simulation's reliability and highlight its relevance in addressing problems such as magnetism, cellular behavior, and the influence of environmental factors on lattice energy. The simulation was implemented in Python, utilizing packages like Pygame, Pillow, and OpenGL for data collection and visualization. Results demonstrate that, regardless of the initial spin configuration of the lattice, the final energy state depends primarily on the coupling constant and the environmental temperature. Additionally, the simulation was employed to investigate the critical temperature of the lattice, identifying the point at which phase transitions occur. This was achieved by analyzing temperature-dependent behavior, including the specific heat and energy fluctuations, which exhibit characteristic peaks near the critical temperature. These results affirm that our simulation functions as intended and can be adapted to explore a variety of physical scenarios. This adaptability underscores its potential for future studies involving lattice-based phenomena and environmental effects on energy structures.

I. INTRODUCTION

The Potts model is a theoretical framework designed to study the behavior of spins on a lattice and their interactions with their environment. A generalization of the Ising model, the Potts model extends the concept by allowing each lattice site to adopt one of N discrete spin states, as opposed to the two-state restriction of the Ising model. This versatility makes the Potts model applicable to a wide range of physical systems, including ferromagnetic materials, cellular structures, and even digital signal processing. Its ability to model phase transitions and cooperative phenomena has made it a cornerstone in statistical mechanics and computational physics. The Potts model was first introduced by Renfrey Potts in 1951 [1], building upon earlier work by Rudolf Peierls and Cyril Domb on phase transitions and lattice systems. While initially conceived to study ferromagnetic interactions, the model has since been applied in diverse fields, including biology, where it is used to simulate cell adhesion and tissue morphogenesis, and computer science, where it plays a role in image segmentation and pattern recognition. This adaptability highlights the model's broad utility across disciplines. At its core, the Potts model describes how the spin state of a lattice point is influenced by the spins of its neighbors and the external conditions of the system, such as temperature. In the simplest ferromagnetic version, the system tends to align spins to minimize its energy, leading to macroscopic phenomena like magnetization. The interplay between thermal fluctuations and spin interactions allows the Potts model to capture critical phenomena, including phase transitions where the system moves between ordered and disordered states.

Experimentally observing these behaviors in real-world systems is often impractical due to the complexity and scale of lattice systems. This is where simulations become invaluable. By leveraging computational methods, researchers can explore the intricate dynamics of lattice models under controlled conditions. In this work, we employed the Metropolis-Hastings algorithm, a widely used Monte Carlo method, to simulate the Potts model. This algorithm efficiently samples spin configurations by probabilistically flipping spins based on the energy difference and system temperature, enabling the study of equilibrium states and dynamic processes within the lattice. The Potts model is not only a theoretical construct but also a tool for gaining insights into practical systems. For example, in solid-state physics, it helps elucidate magnetic behaviors in materials, while in biological systems, it models how cells interact and form complex structures. Its role in computational simulations bridges the gap between theoretical predictions and real-world applications, making it an indispensable framework for understanding lattice-based phenomena. This paper focuses on our implementation of the Potts model to study the relationship between lattice spin configurations, environmental factors, and system energy. Through simulations, we investigate how temperature and coupling constants influence the system's behavior, with particular emphasis on identifying critical points where phase transitions occur. By providing a detailed analysis of the Potts model and its applications, this work contributes to a broader understanding of its potential to address problems across various fields of physics and beyond.

II. THEORETICAL BACKGROUND

The Potts model is a model used in statistical mechanics to describe the energy of a lattice.This model is a generalization of the Ising model, which performs the same task. The model calculates the energy of particles in this lattice system based on the different spin states of it and its nearest neighbor. This calculation is called the Hamiltonian. The energy of the system can be used to

calculate a value called the specific heat, which helps to describe phase transitions. The Potts model can be used in both two- and three-dimensional dimensions, making it incredibly versatile.

A. Relation to the Ising Model

The Ising model is a fundamental framework in statistical mechanics, describing a lattice of interacting particles, each with a binary spin state, $\sigma_i = \pm 1$. The energy of the system is determined by interactions between neighboring spins, favoring aligned configurations to minimize the system's energy. This model is particularly effective for studying binary phase transitions, such as in ferromagnetic systems.

The Potts model generalizes the Ising model by allowing each lattice site to take on q discrete spin states, $\sigma_i \in \{1, 2, \ldots, q\}$. For $q = 2$, the Potts model reduces to the Ising model. For $q > 2$, the model enables the study of more complex systems with multiple coexisting states. Systems with $q = 3$ can represent interactions among three phases, such as liquid, gas, and solid, while larger q values are used to model systems like multi-component alloys or biological structures.

B. The Hamiltonian of the Potts Model

The Hamiltonian of the Potts model quantifies the energy of a spin configuration on a lattice. It is given by

$$H(\omega) = -J \sum_{\langle i,j \rangle} \delta(\sigma_i, \sigma_j), \tag{1}$$

where J is the interaction strength, $\langle i, j \rangle$ denotes pairs of nearest neighbors, and $\delta(\sigma_i, \sigma_j)$ is the Kronecker delta function, defined as

$$\delta(\sigma_i, \sigma_j) = \begin{cases} 1 & \text{if } \sigma_i = \sigma_j, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

This Hamiltonian assigns lower energy to configurations where neighboring spins are aligned. The magnitude of J determines the preference for alignment, with higher J values leading to more ordered configurations. The lattice structure also influences the interactions, with the number of nearest neighbors varying between two-dimensional and three-dimensional systems, as described in Section II D.

C. Specific Heat and Energy Fluctuations

The specific heat, C_v, of the system measures its ability to absorb thermal energy. It is directly related to fluctuations in the energy of the system and is expressed as

$$C_v = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2 V}, \tag{3}$$

where $\langle E \rangle$ is the average energy, $\langle E^2 \rangle$ is the mean square of the energy, k_B is the Boltzmann constant, T is the temperature, and V is the system's volume.

For two-dimensional systems, the volume V in Equation 3 is replaced by the system's area A, yielding

$$C_v^{\text{2D}} = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2 A}. \tag{4}$$

In three-dimensional systems, the volume V remains as the spatial extent of the lattice:

$$C_v^{\text{3D}} = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B T^2 V}. \tag{5}$$

Specific heat peaks at critical temperatures where phase transitions occur. In two-dimensional systems, these transitions involve reorganization of spins across a surface, while in three-dimensional systems, they involve bulk changes in the material's properties.

D. Dimensionality of the Potts Model

The dimensionality of the system significantly affects the behavior of the Potts model. In two-dimensional systems, interactions are confined to a finite area, with the number of nearest neighbors depending on the lattice geometry. For instance, a square lattice has four nearest neighbors per site, while a triangular lattice has six. These constraints result in a relatively simpler energy landscape compared to three-dimensional systems.

In three-dimensional systems, the lattice extends into an additional spatial dimension, increasing the number of nearest neighbors and interaction pathways. For example, a cubic lattice has six nearest neighbors per site, while body-centered and face-centered cubic lattices have eight and twelve, respectively. The additional connectivity in three dimensions leads to sharper phase transitions and more pronounced energy fluctuations.

This dimensional dependence underscores the versatility of the Potts model in capturing the interplay between local interactions and macroscopic properties, offering insights into both surface-dominated and bulk-dominated phenomena.

III. METHODS

A. Simulation Overview

The Potts model was implemented in Python to numerically simulate the behavior of a three-dimensional

lattice under varying conditions of temperature and coupling constants. The primary objective was to explore phase transitions and equilibrium energy states of the lattice using the Metropolis Algorithm. The 3D model serves to both visually simulate the behavior of the lattice so as to give a physical interpretation to trends, as well as generate a basis for quantitative analysis. In this way, the model was not merely a visualization tool but also the primary source of data for all subsequent analysis. Key trends, such as those shown in the Energy vs. Steps and Specific Heat vs. Temperature graphs (in Results section), were derived directly from data from the 3D lattice simulation, ensuring consistency between the visualization and numerical results.

B. Visualization Models

1. *Cross-Section Model*

This model displays a two-dimensional cross-section of the three-dimensional cube lattice. It output a video that shows the evolution of the cross-section over time/steps, progressing from its initial lattice configuration (either random or uniform) to its final state, determined by the number of steps taken.

For this model, Matplotlib was used to render the slice and create animations, and IPython.display (HTML) to embed the animations in Jupyter Notebooks for interactive use.

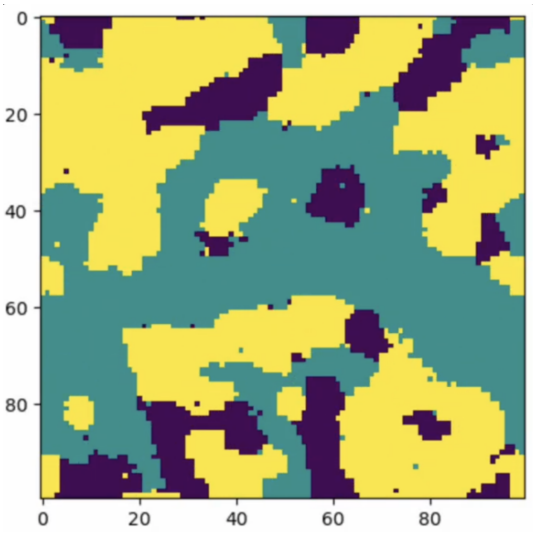

FIG. 1: Pictured above is a 2D simulation for a 3D cube-shaped lattice. This was our first step in running the simulations to ensure that we were seeing the expected behaviors. This specific example was run on a 100x100x100 lattice with 3 possible spin states. This is a cross section of the cube cut down the middle in the Kth dimmension.

2. *3D Cube Model*

This model is similar to the first in that it iterates over time with a predetermined initial lattice (chosen by the user). It differs in that it displays the full three-dimensional visualization of the lattice in an interactive way (not a video). This model allows its user to rotate the cube lattice using keyboard controls (A, W, D, S keys) as the lattice evolves over time, iterating and progressing over the steps to its final state.

For this model, the Pygame library was used to create the application window, manage user input and handle real-time updates to the visualization. PyOpenGl and GLU were employed in order to render the 3D cube, applying textures and managing interactive rotation. Moreover, each spin in the cube mapped to a unique RGB color in order to clearly represent the different states.

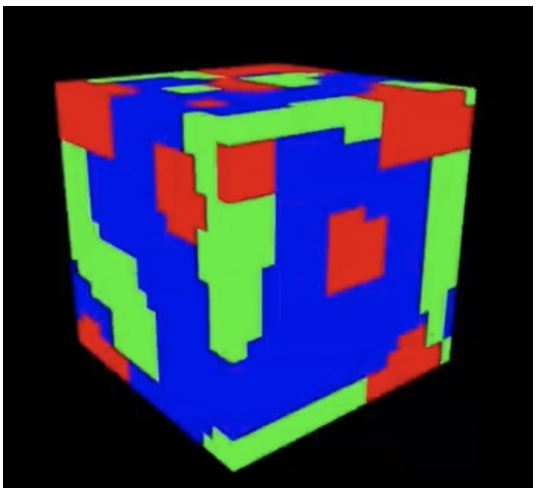

FIG. 2: Pictured above is a 3D simulation for a 3D cube-shaped lattice. Due to computing power we were constrained to a 20x20x20 lattice. As with the 2D simulation this example shown has 3 spin states.

C. Initialization

The lattice was defined as a three-dimensional grid of dimensions N×M×K, with each site assigned one of n discrete spin states. The initial configuration of the lattice was set to either a randomized or uniform state. For the randomized configuration, each site in the grid was randomly assigned one of n possible spin states. For the uniform configuration, all sites in the grid were assigned the same spin state.

For the 3D Cube model, the dimension of the lattice were set to 20x20x20. The size was chosen to balance the visualization quality with performance as large lattice dimension result in significant lag due to the computation demand of rendering. In contrast, the cross-section model is better suited for handling and creating visualizations of larger lattices because it processes only a

two-dimensional slice at a time.

The current implementation of the 3D Cube model supports n = 3 spin states (as well as n = 2 spin states for the Isings model). However, the model can be extended to handle more spin states with minor modifications to the code.

D. Algorithm: Metropolis Hastings

The Metropolis-Hastings algorithm was used to model the evolution of the lattice. The procedure for this algorithm is as follows:

1. Site Selection: A random lattice site is selected.

2. Spin Proposal: A new random spin is proposed for the selected site.

3. Energy Difference Calculation: The change in energy (Δ E) resulting from the proposed spin flip, compared to the current spin state, is computed using the energy formula described in the theoretical background. This formula accounts for the short-range interactions of the spin with its nearest neighbors.

4. The proposed spin is accepted or rejected based on the following conditions:

   - If Δ E ≤ 0, the proposed spin is automatically accepted, as it reduces the system's energy
   - If Δ E > 0, the proposed spin is accepted probabilistically, with a probability P $= e^{-\beta \Delta E}$, where β is proportional to the inverse of temperature

5. If the spin is accepted, the flip is applied to the lattice while the rejected proposals leave the site unchanged.

This iterative process is repeated for multiple steps until the system reaches equilibrium.

E. Energy and Specific Heat Calculations

Energy and specific heat data were recorded at regular intervals during the simulation and plotted using Matplotlib and Excel. The total energy was calculated using the formula Eq.1 . The specific heat was calculated by averaging energy fluctuations over time. These calculations allow for the identification of phase transitions and other trends in the graphs further explained in the results sections.

F. Parameter Explorations

The lattice was explored varying conditions, allowing the user to control key parameters to study their effects on the system's behavior:

- Coupling Constant (J): The coupling constant, which governs the interaction strength between neighboring spins, was varied to investigate how spin states align. This is directly tied to ferromagnetism and antiferromagnetic.

- Inverse Temperature: The inverse temperature parameter, β, was adjusted to test the effects of thermal energy (or the lack thereof) on the lattice.

Both random and uniform initial configurations were used in conjunction with these parameters to examine their combined effects. This approach provided a better understanding of the relationships between initial conditions, coupling constants, and temperature, particularly in studying phase transitions and equilibrium states.

G. Computational Tools

The simulation was implemented using Python libraries including:

- NumPy: Used extensively for numerical computations, including generating random lattice configurations, calculating energy differences, and iterating over lattice sites.

- Matplotlib: For the 2D cross-section simulation. Allows for creation of static plots such as the Energy vs. Time/Steps graph, and dynamic animation showing the evolution of the lattice.

- IPython.Display(HTML): Facilitated the embedding of animations directly into Jupyter Notebook environments. This functionality allowed for an interactive and seamless presentation of simulation results within the notebook.

- Pygame: Utilized in the 3D cube model to create the application window, manage user inputs, and handle real-time rendering updates.

- OpenGL: Played a key role in the 3D cube visualization, rendering the lattice and enabling interactive rotation. These libraries allowed textures to be applied to the cube faces, creating a visually intuitive representation of the lattice.

- Pillow: Used for generating and manipulating textures in the 3D cube model. It handled tasks such as converting spin field data into RGB textures and resizing them to fixed dimensions for compatibility with OpenGL.

IV. DATA COLLECTION AND ANALYSIS

The two main goals of the analysis was too try and find the critical temperature as well as the final energy of each lattice. In order to do this we used our code to take "snapshots" of the total energy of the lattice as time progressed throughout the simulation. All of the data shown in this section was collected using the 3D simulation and not the 2D simulation. This was not done for any reason other than to have one file with all the code present.

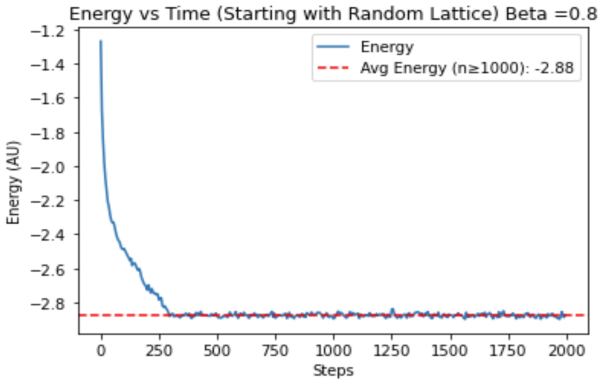

FIG. 3: Pictured above is the energy vs time graph for a random lattice where the β value is equal to 0.8. Since the lattice begins with extremely high energy we expect the energy to drop rapidly until it "oscillates" around a specific value. This is what we observe as the value converges to -2.88.

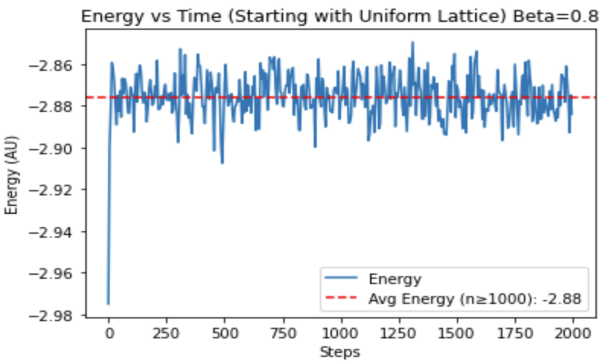

FIG. 4: Pictured above is the energy vs time graph for a uniform lattice where the β value is equal to 0.8. Contrary to the FIG 3, we expect this lattice to quickly increase in energy until it converges to the same final value of -2.88. This is what we observe.

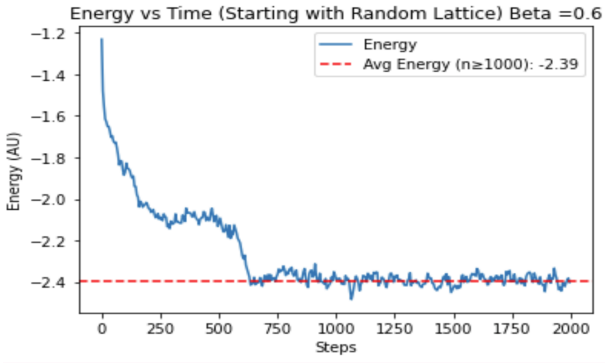

FIG. 5: Pictured above is the energy vs time graph for a random lattice where the β value is equal to 0.6. The same logic applies to the previous graphs.

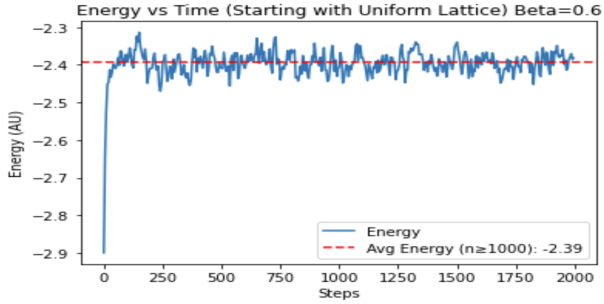

FIG. 6: Pictured above is the energy vs time graph for a uniform lattice where the β value is equal to 0.6. The same logic applies to the previous graphs.

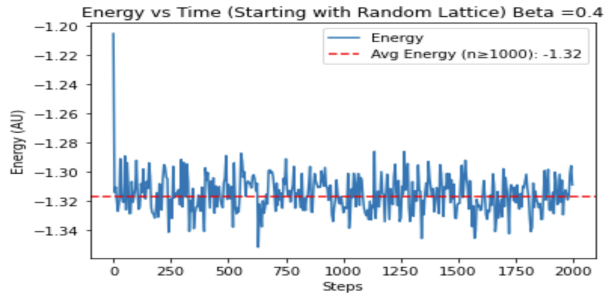

FIG. 7: Pictured above is the energy vs time graph for a random lattice where the β value is equal to 0.4. The same logic applies to the previous graphs.

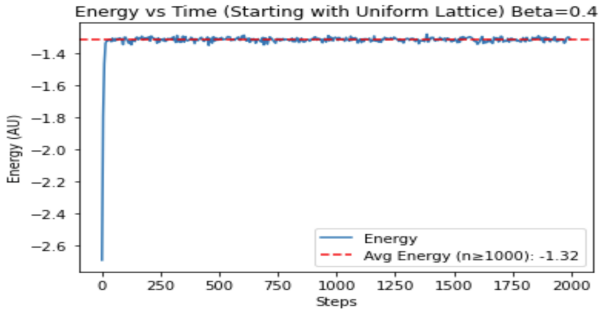

FIG. 8: Pictured above is the energy vs time graph for a uniform lattice where the β value is equal to 0.4. The same logic applies to the previous graphs.

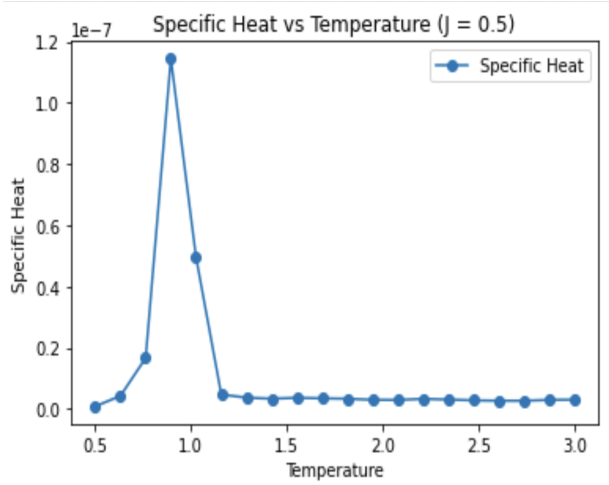

FIG. 9: Pictured above is the graph showing how the specific heat of the lattice changes with the temperature of the system. This data had a coupling constant value, J = 0.5. As stated previously we expect a clear peak to indicate a phase transition and the corresponding temperature value to be T_c.

TABLE I: Convergence Values for β

β	Initial State	Final Energy
0.8	Random (High Energy)	-2.88
0.8	Uniform (Low Energy)	-2.88
0.6	Random (High Energy)	-2.39
0.6	Uniform (Low Energy)	-2.39
0.4	Random (High Energy)	-1.32
0.4	Uniform (Low Energy)	-1.32

Based on this data we come to the conclusion we expect where the final energy of the lattice has no dependence on the initial energy, and is only dependent on the coupling constant, lattice size, and temperature of environment the lattice dwells.

The next step was too use Eq.5 too compare the specific heat and temperature of the system. We do this too look for the temperature where a phase transition occurs and the lattice changes from uniformity to non-uniformity. We are specifically interested in the relationship between the coupling constant, J, and the Critical temperature, T_c

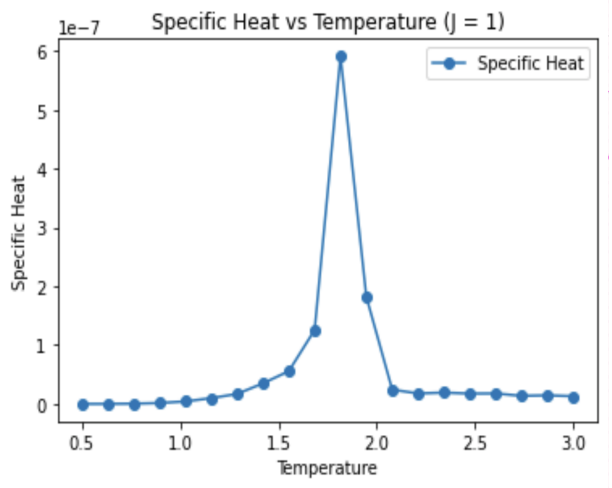

FIG. 10: Pictured above is the graph showing how the specific heat of the lattice changes with the temperature of the system. This data had a coupling constant value, J = 1.

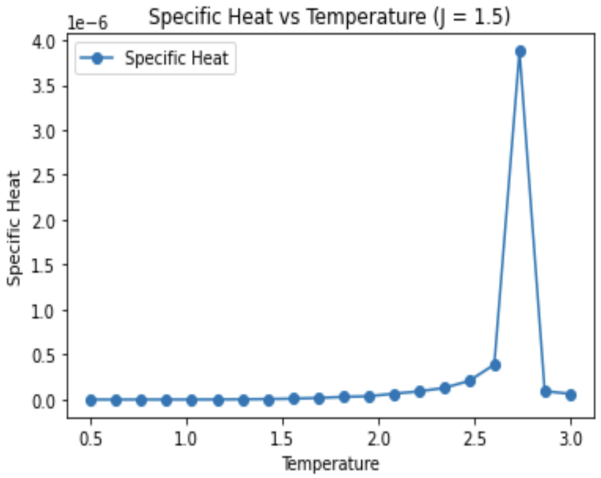

FIG. 11: Pictured above is the graph showing how the specific heat of the lattice changes with the temperature of the system. This data had a coupling constant value, J = 1.5.

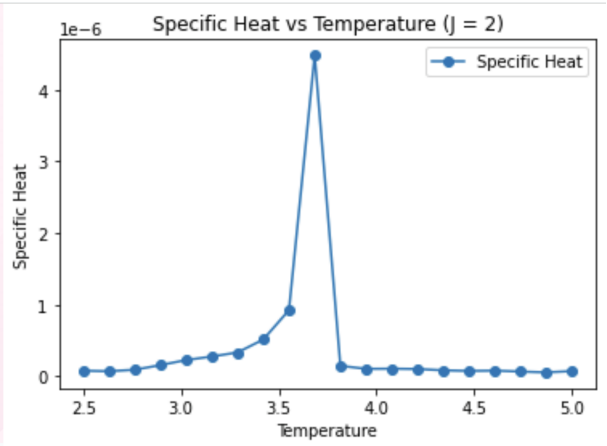

FIG. 12: Pictured above is the graph showing how the specific heat of the lattice changes with the temperature of the system. This data had a coupling constant value, J = 2.

TABLE II: Coupling Constant vs Critical Temp.

Coupling Constant (J)	Critical Temperature (T_c
0.5	0.9
1	1.8
1.5	2.7
2	3.6

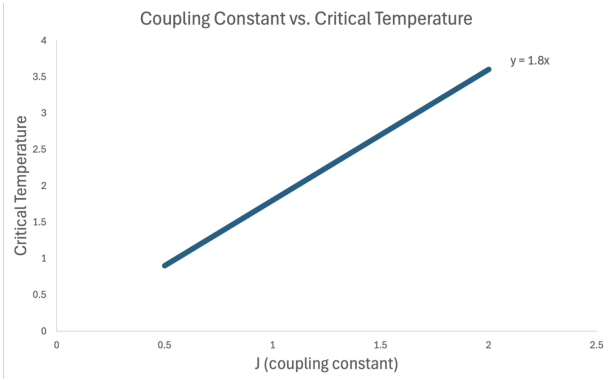

FIG. 13: The figure above shows the relationship between the coupling constant and critical temperature values shown in Table II.

Based on Figure 13, we can show that the relation ship between the critical temperature and coupling constant of a lattice is proportional. Based on the other variables in our simulations this resulted in a slope of 1.8. This slope value is not constant and is subject to change depending on the number of spin states a point could take on, as well as the overall size of the lattice itself. However such a well defined slope value should be investigated further.

However, since the goal of the data analysis was to show that our simulation functions properly we did not investigate this relationship further.

V. DISCUSSION AND CONCLUSIONS

Based on the results of our study, we conclude that our Potts model simulation performs as intended, enabling an effective investigation of lattice properties while maintaining flexibility through customizable parameters. The primary objective of this work was to develop a robust and reliable model capable of simulating spin interactions on a lattice and verifying its consistency with established theoretical expectations. Specifically, we sought to demonstrate that the lattice would converge to an equilibrium energy state influenced by factors such as lattice size and environmental temperature. Additionally, we successfully identified the critical temperature at which the lattice undergoes a phase transition, validating our simulation's accuracy. To achieve these goals, we implemented a suite of Python functions capable of performing necessary calculations, recording data, and visually displaying simulations. This allowed us to observe spin dynamics in real time and confirm that our results aligned with theoretical predictions. The ability to directly visualize and analyze lattice behavior was instrumental in ensuring the validity of our model.

Looking ahead, this project presents numerous opportunities for further development and application. One potential avenue involves optimizing the algorithm to improve computational efficiency. For instance, instead of

flipping individual spins one at a time, we could implement cluster algorithms, such as the Wolff or Swendsen-Wang methods, to accelerate convergence, particularly near the critical temperature. Another promising direction is adapting the model to address specific scientific domains, such as magnetism, solid-state physics, cellular interactions, and digital signal processing. By tailoring our implementation to these areas, we could explore more targeted phenomena and potentially contribute to practical advancements in these fields. We also see potential in modifying the algorithm to accommodate different lattice geometries. For example, transitioning from a cubic lattice to honeycomb or triangular lattices could provide insights into how lattice structure influences system behavior and critical phenomena. Finally, with our current framework, we aim to delve deeper into the variables that affect the critical temperature of the lattice. Investigating factors such as lattice size, spin multiplicity, and coupling constants could provide a more comprehensive understanding of phase transitions and the interplay between system parameters. In conclusion, our work demonstrates that the Potts model is a versatile and powerful tool for studying lattice dynamics and critical phenomena. The foundation established here provides a solid platform for future research and applications across various scientific disciplines.

VI. ACKNOWLEDGMENTS

This experiment was completed collaboratively by Carter Chapman, Dev Singh, and Florenica Nardone, each lab group member working directly with the code for the simulation and data collection. Guidance regarding the project was provided by professor Joaquin Drut. The Abstract, Introduction, Data Collection and Analysis, Discussion section were written by Carter Chapman, Florencia Nardone wrote the methods, and Dev Singh wrote the theoretical background.

[1] Potts Model Wikipedia Page
https://en.wikipedia.org/wiki/Potts_model
[2] Hyperphysics: Characteristic X-Rays
http://hyperphysics.phy-astr.gsu.edu/hbase/quantum/xrayc.html
[3] A Review of the Potts Model
https://scholar.rose-hulman.edu/cgi/viewcontent.cgi?article=1194&context=rhumj