

2D Ising Model on Ferromagnetism and Antiferromagnetism

Theory and Simulations

John Andrew Kypriotakis | Computational Physics | May 11, 2023

Table of Contents

[Abstract 2](#_Toc134726606)

[Introduction 2](#_Toc134726607)

[Background 2](#_Toc134726608)

[Motivation 2](#_Toc134726609)

[Objectives 3](#_Toc134726610)

[The 2D Ising Model 4](#_Toc134726611)

[Ising Model basics 4](#_Toc134726612)

[Spin Interractions 4](#_Toc134726613)

[Hamiltonian and Partition Function 4](#_Toc134726614)

[Analytical Techniques 5](#_Toc134726615)

[Monte Carlo Simulations 5](#_Toc134726616)

[The Metropolis Algorithm 5](#_Toc134726617)

[Introduction 5](#_Toc134726618)

[The Metropolis Algorithm 5](#_Toc134726619)

[Application to the 2D Ising Model 6](#_Toc134726620)

[Efficiency and Improvements 7](#_Toc134726621)

[Limitations and Considerations 7](#_Toc134726622)

[Ferromagnetism in the 2D Ising Model 7](#_Toc134726623)

[Phenomenological Description 7](#_Toc134726624)

[Critical Temperature and Phase Transition 8](#_Toc134726625)

[Order Parameter 8](#_Toc134726626)

[Finite-Size Effects 8](#_Toc134726627)

[Experimental Observations 8](#_Toc134726628)

[Theoretical Predictions 9](#_Toc134726629)

[Antiferromagnetism in the 2D Ising Model 9](#_Toc134726630)

[Description and Characteristics 9](#_Toc134726631)

[Néel Temperature and Phase Transition 9](#_Toc134726632)

[Sublattice Magnetization 9](#_Toc134726633)

[Ground State Configurations 10](#_Toc134726634)

[Experimental Observations 10](#_Toc134726635)

[Theoretical Predictions 10](#_Toc134726636)

[Our Code 10](#_Toc134726637)

[Scope 10](#_Toc134726638)

[Ferromagnetic results 11](#_Toc134726639)

[References 11](#_Toc134726640)

# Abstract

The 2D Ising model is a powerful mathematical framework for studying magnetic properties in condensed matter physics. In this paper, we investigate the behavior of the 2D Ising model with a focus on ferromagnetism and antiferromagnetism. We analyze the underlying principles, theoretical foundations, and simulation techniques utilized to study these magnetic phenomena, in the context of the Ising model, with emphasis on Monte Carlo simulations. Additionally, we compare the key features and observable properties of ferromagnetic and antiferromagnetic systems in the 2D Ising model.

# Introduction

## Background

The study of magnetic materials and their properties has been a subject of significant interest in condensed matter physics. Understanding the behavior of magnetic systems is crucial for various technological applications, ranging from data storage devices to spintronics. The 2D Ising model, introduced by Ernst Ising in 1925, has emerged as a powerful mathematical framework for investigating magnetic phenomena in two-dimensional systems. The Ising model provides a simplified yet insightful description of magnetic behavior by considering a lattice of spins with interactions between neighboring spins.

## Motivation

Ferromagnetism and antiferromagnetism are two fundamental types of magnetic ordering observed in materials. Ferromagnetic materials exhibit a net magnetization in the absence of an external magnetic field, while antiferromagnetic materials have an intrinsic cancellation of magnetic moments between neighboring spins. The 2D Ising model offers a convenient platform to study these magnetic orders and explore their characteristics, phase transitions, and related properties. By investigating the behavior of the 2D Ising model for ferromagnetism and antiferromagnetism, we can gain valuable insights into the broader understanding of magnetic systems and their dynamics.

## Objectives

The main objective of this paper is to provide a comprehensive analysis of the 2D Ising model focusing on ferromagnetism and antiferromagnetism. Specifically, we aim to achieve the following goals:

* To present the theoretical foundations of the 2D Ising model, including the basic concepts, spin interactions, Hamiltonian, and partition function. We will discuss the analytical techniques used to study the model as well as the numerical simulation methods such as Monte Carlo simulations.
* To investigate ferromagnetism in the 2D Ising model, including the phenomenological description, critical temperature, order parameter, finite-size effects, and the comparison between experimental observations and theoretical predictions.
* To explore antiferromagnetism in the 2D Ising model, encompassing its description and characteristics.
* To conduct a numerical study of ferromagnetism and antiferromagnetism in the context of the 2D Ising model.
* To discuss the applications and extensions of the 2D Ising model beyond the scope of this paper, including its relevance in material science and engineering, quantum magnetism, and topological aspects.

Through these objectives, we aim to deepen our understanding of the 2D Ising model and its implications for magnetic behavior, phase transitions, and related phenomena. By comparing ferromagnetic and antiferromagnetic systems within the Ising model, we hope to contribute to the broader understanding of magnetic materials.

# The 2D Ising Model

## Ising Model basics

The Ising model is a lattice-based mathematical model that describes the behavior of a collection of interacting spins. In the two-dimensional (2D) Ising model, the spins are arranged on a lattice, typically a square lattice, where each lattice site contains a spin variable that can take two values: up or down (represented as +1 and -1, respectively). The spins interact primarily with their nearest neighbors, and the energy of the system depends on the orientation of neighboring spins.

## Spin Interractions

In the 2D Ising model, the interactions between spins are commonly described by the nearest-neighbor interaction, where each spin interacts only with its adjacent spins. The interaction between two spins is often quantified by an exchange coupling constant, J, which determines the strength of the interaction. If , the system favors alignment of neighboring spins, promoting ferromagnetic behavior. Conversely, if , the system tends to have antiferromagnetic interactions, where neighboring spins prefer to be anti-aligned.

## Hamiltonian and Partition Function

The Hamiltonian of the 2D Ising model describes the energy of the system and is defined as the sum of the interaction energies between neighboring spins. Mathematically, it is given by:

where and are indices denoting lattice sites, represents the spin variable at site , and is the exchange coupling constant.

The partition function, Z, is a central quantity in statistical mechanics and provides a framework for calculating various thermodynamic properties of the system. It is defined as the sum of the Boltzmann factors over all possible spin configurations:

where represents the set of all possible spin configurations and is the inverse temperature ( is the Boltzmann constant, and is the temperature).

## Analytical Techniques

The 2D Ising model can be solved analytically in certain special cases. One such case is the infinite lattice limit, where the model exhibits a phase transition at a critical temperature known as the Curie temperature . Various analytical techniques, such as mean-field theory, exact solutions for small lattices, and high-temperature expansions, have been employed to study the behavior of the 2D Ising model and investigate the critical properties near .

## Monte Carlo Simulations

Due to the complexity of the 2D Ising model, analytical solutions are limited to specific cases. Monte Carlo simulations provide a powerful numerical approach to study the model and explore its behavior for various system sizes and interaction strengths. In Monte Carlo simulations, random spin updates are performed according to the Metropolis algorithm or other variants, allowing the system to explore its configuration space and reach equilibrium. By averaging over many Monte Carlo steps, various thermodynamic quantities such as the magnetization, energy, and correlation functions can be calculated.

# The Metropolis Algorithm

## Introduction

In the study of statistical physics and Monte Carlo simulations, the Metropolis algorithm plays a pivotal role in sampling configurations and exploring the phase space of a system. Proposed by Nicholas Metropolis et al. in the 1950s, this algorithm revolutionized the field by providing an efficient way to simulate complex systems and calculate thermodynamic properties. This chapter focuses on explaining the principles and workings of the Metropolis algorithm, highlighting its significance in the context of the 2D Ising model for ferromagnetism and antiferromagnetism.

## The Metropolis Algorithm

The Metropolis algorithm is a Markov chain Monte Carlo (MCMC) method used to sample configurations from the Boltzmann distribution of a system. It allows us to approximate the equilibrium state of a system by iteratively updating the configuration according to a probabilistic rule. The algorithm consists of the following steps:

1. Start with an initial configuration of the system.
2. Choose a spin or a group of spins to update.
3. Propose a trial move, which flips the selected spin(s) to a new configuration.
4. Calculate the change in energy, , between the current configuration and the proposed configuration.
5. If , accept the move and update the configuration.
6. If , generate a random number, r, between 0 and 1.
   1. If accept the move and update the configuration.
   2. If , reject the move and keep the current configuration.
7. Repeat steps 2-6 for a sufficient number of iterations or until reaching equilibrium.

The algorithm ensures that detailed balance is satisfied, meaning that the system eventually reaches the Boltzmann distribution at the desired temperature. The acceptance probability, , allows for both uphill and downhill moves, allowing the system to explore higher-energy configurations and escape local energy minima.

## Application to the 2D Ising Model

The Metropolis algorithm is particularly well-suited for studying the 2D Ising model, which describes the behavior of spins on a lattice. In the Ising model, each spin can take two values: up (+1) or down (-1), representing the orientation of the magnetic moment. The energy of the system is determined by the interactions between neighboring spins.

To apply the Metropolis algorithm to the 2D Ising model, we initialize the lattice with random spin configurations. At each iteration, we select a spin randomly and propose a trial move by flipping its orientation. We then calculate the change in energy, ΔE, resulting from this move. If ΔE is negative, indicating a decrease in energy, we accept the move unconditionally. If ΔE is positive, we generate a random number and compare it with the acceptance probability. If the random number is smaller than the acceptance probability, we accept the move; otherwise, we reject it.

By repeatedly applying the Metropolis algorithm and updating the spin configurations, we can explore the phase space of the 2D Ising model and obtain statistical averages of physical quantities such as the magnetization, energy, and susceptibility. The algorithm allows us to study the system at different temperatures and observe phase transitions, including the ferromagnetic-paramagnetic transition and the antiferromagnetic behavior in the presence of competing interactions.

## Efficiency and Improvements

While the Metropolis algorithm is a powerful tool, it can be computationally expensive, especially for large systems. Several techniques have been developed to improve its efficiency, such as cluster algorithms (e.g., the Wolff algorithm) that update clusters of spins instead of individual spins, reducing the correlation time and accelerating convergence to equilibrium. Other techniques include parallel tempering, where multiple replicas of the system are simulated at different temperatures and occasionally exchanged, and the use of faster data structures and algorithms for efficient energy calculations.

## Limitations and Considerations

Although the Metropolis algorithm is widely used and highly effective, it is important to be aware of its limitations and considerations. One key aspect is the choice of the initial configuration, which can affect the convergence and exploration of the phase space. It is common practice to start with a random or thermalized initial configuration, but different choices may lead to different results.

Another consideration is the determination of the appropriate number of iterations needed to reach equilibrium and obtain accurate statistical averages. Convergence to equilibrium can be assessed by monitoring physical observables and their fluctuations over time. Autocorrelation analysis can provide insights into the correlation time of the system, helping to estimate the number of iterations required for independent samples.

Additionally, the choice of the lattice size and boundary conditions can influence the behavior of the system. The finite size of the lattice introduces finite-size effects, such as the rounding off of phase transitions. Periodic boundary conditions are commonly used to mimic an infinite lattice and reduce boundary effects.

# Ferromagnetism in the 2D Ising Model

## Phenomenological Description

Ferromagnetism is a magnetic ordering phenomenon characterized by the alignment of spins in the same direction, leading to a net magnetization even in the absence of an external magnetic field. In the 2D Ising model, ferromagnetic behavior occurs when the exchange coupling constant, J, is positive. The spins tend to align parallel to each other to minimize the energy and achieve a state of lower entropy.

## Critical Temperature and Phase Transition

One of the key features of ferromagnetism in the 2D Ising model is the occurrence of a phase transition at a critical temperature, . Below , the system exhibits spontaneous magnetization, while above , the magnetization disappears. The critical temperature depends on the exchange coupling constant, lattice size, and dimensionality of the system. The critical behavior near is described by power laws, and various critical exponents characterize different thermodynamic quantities.

## Order Parameter

In ferromagnetic systems, the order parameter serves as a measure of the degree of magnetization and the phase transition. In the 2D Ising model, the order parameter is defined as the average magnetization per spin, given by:

where N is the total number of spins and is the spin variable at site . The order parameter is non-zero below, indicating the presence of spontaneous magnetization, while it vanishes above .

## Finite-Size Effects

In finite systems, size effects play a crucial role in ferromagnetism. As the lattice size decreases, fluctuations become more prominent, and the critical temperature may shift. Finite-size scaling theory provides a framework to account for these size effects and extract the behavior in the thermodynamic limit. The study of finite-size effects in the 2D Ising model allows for a better understanding of how ferromagnetic properties manifest in different system sizes.

## Experimental Observations

Experimental investigations of ferromagnetic behavior in real materials have provided valuable insights and confirmation of the theoretical predictions of the 2D Ising model. Magnetic materials such as iron, nickel, and cobalt exhibit ferromagnetic properties and display a phase transition at a critical temperature. Experimental techniques, such as magnetization measurements, neutron scattering, and microscopy, have been employed to study the microscopic details of ferromagnetic materials and validate the theoretical understanding based on the 2D Ising model.

## Theoretical Predictions

The 2D Ising model has provided several theoretical predictions regarding ferromagnetism. The critical exponents characterizing the phase transition, such as the magnetization exponent and correlation length exponent, have been calculated using various analytical techniques. The scaling relations between these exponents provide insight into the universality of critical phenomena. The theoretical predictions of the 2D Ising model have served as a foundation for the study of ferromagnetism and have been verified and refined through experimental observations.

# Antiferromagnetism in the 2D Ising Model

## Description and Characteristics

Antiferromagnetism is a magnetic ordering phenomenon characterized by the alignment of spins in an alternating pattern, resulting in a net magnetization of zero. In the 2D Ising model, antiferromagnetic behavior occurs when the exchange coupling constant, J, is negative. The neighboring spins tend to align in opposite directions to minimize the energy and achieve a state of lower entropy.

## Néel Temperature and Phase Transition

Similar to ferromagnetism, antiferromagnetism in the 2D Ising model undergoes a phase transition at a critical temperature known as the Néel temperature, . Below the system exhibits antiferromagnetic ordering, where the spins align in an alternating fashion, forming antiparallel pairs. Above , thermal fluctuations disrupt the antiferromagnetic order, leading to a disordered state.

## Sublattice Magnetization

In antiferromagnetic systems, the sublattice magnetization serves as a measure of the degree of antiferromagnetic order. In the 2D Ising model, the lattice can be divided into two sublattices, and the sublattice magnetization is defined as the difference in magnetization between the two sublattices. It quantifies the imbalance in spin orientations between the sublattices and provides information about the antiferromagnetic order parameter.

## Ground State Configurations

The ground state configurations of the 2D Ising model with antiferromagnetic interactions exhibit a highly degenerate nature. There are multiple ground state configurations that satisfy the antiferromagnetic order, where spins on one sublattice are aligned opposite to the spins on the other sublattice. These ground state configurations have specific spin patterns, such as checkerboard or stripe-like arrangements, depending on the lattice geometry and boundary conditions.

## Experimental Observations

Antiferromagnetism has been observed in various real materials, such as manganese oxide compounds and certain types of iron-based superconductors. Experimental techniques, including neutron scattering, X-ray diffraction, and magnetic resonance imaging, have provided insights into the microscopic details of antiferromagnetic ordering and its characteristic features. The behavior observed in these materials aligns with the theoretical understanding based on the 2D Ising model for antiferromagnetism.

## Theoretical Predictions

The 2D Ising model has yielded several theoretical predictions concerning antiferromagnetism. Analytical calculations have been performed to determine critical exponents associated with the antiferromagnetic phase transition, as well as to investigate the effects of various lattice geometries and boundary conditions on the ground state configurations. These theoretical predictions provide a basis for understanding the nature of antiferromagnetism and have been validated through experimental observations in real materials.

# Our Code

## Scope

Our code aims to use the simple Metropolis algorithm to numerically compute the Ising model of both a ferromagnetic (code: Appendix A) and an antiferromagnetic (code: Appendix B) material. Our lattice will be either orthogonal (4 nearest neighbors) or hexagonal (6 nearest neighbors). The results will be plots of energy vs temperature and magnetization vs temperature (for the steady state) and snapshots of the lattice in various timestamps. Through them we aim to show the of the system, as well as the progression of magnetization over time. To simplify the code, we choose a unit system where both and are equal to 1.

## Ising Model

The Ising model class has the following components:

def \_\_init\_\_(

self, temperature, lattice, iterations, mesh="ortho", prop="ferro", plots=True

):

self.temperature = temperature

self.lattice = lattice

self.iterations = iterations

if plots:

if plots == "animate":

self.plots = True

self.animate = True

else:

self.plots = True

else:

self.plots = False

if mesh == "ortho":

self.mesh = "ortho"

elif mesh == "hex":

self.mesh = "hex"

else:

raise Exception("Incorrect mesh")

if prop == "ferro":

self.prop = "ferro"

elif prop == "antiferro":

self.prop = "antiferro"

else:

raise Exception("Incorrect property")

This is the class init where all parameters are set. We allow for ferromagnetic and ferromagnetic material as well as hexagonal or orthogonal mesh.

The crystal will be a numpy 2D array, with all lattice positions initially set to 1 or -1 randomly. The Metropolis algorithm will run and change this crystal.

For any given crystal state the energy will be calculated as:

def E(self, crystal, lattice):

energy = 0

for i in range(len(crystal)):

for j in range(len(crystal)):

S = crystal[i, j]

nb = (

crystal[(i + 1) % lattice, j]

+ crystal[i, (j + 1) % lattice]

+ crystal[(i - 1) % lattice, j]

+ crystal[i, (j - 1) % lattice]

)

if self.mesh == "hex" and ((i + 1) % 2 == 0):

nb += (

crystal[(i - 1) % lattice, (j - 1) % lattice]

+ crystal[(i + 1) % lattice, (j - 1) % lattice]

)

elif self.mesh == "hex":

nb += (

crystal[(i + 1) % lattice, (j + 1) % lattice]

+ crystal[(i - 1) % lattice, (j + 1) % lattice]

)

if self.prop == "ferro":

energy += -nb \* S

elif self.prop == "antiferro":

energy += nb \* S

if self.mesh == "hex":

return energy / 6.0

elif self.mesh == "ortho":

return energy / 4.0

Here we have summed the nearest neighbors for every lattice point, then multiply by the point value, multiplying by 1 or -1 (depending on whether the material is ferromagnetic or antiferromagnetic see chapter “The 2D Ising Model”).

The nearest neighbors in the hexagonal mesh case were identifying depending on the x-axis position of the lattice point. If that were an even multiple of the lattice constant, the nearest neighbors were selected to be . If the point was in an x position, odd multiple of the lattice constant the nearest neighbors were . This is visualized on the following graphic:

A picture containing pattern, symmetry, square, design

Description automatically generated

For any given crystal state the total magnetization is:

def M(self, crystal):

mag = sum(crystal)

return mag

Essentially the sum of the crystal is the magnetization with all lattice points as +1 or -1 being added.

Each step of the metropolis algorithm is:

def mcStep(self, crystal, lattice, beta):

for i in range(lattice):

for j in range(lattice):

a = randint(0, lattice)

b = randint(0, lattice)

s = crystal[a, b]

nb = (

crystal[(a + 1) % lattice, b]

+ crystal[a, (b + 1) % lattice]

+ crystal[(a - 1) % lattice, b]

+ crystal[a, (b - 1) % lattice]

)

if self.mesh == "hex" and ((a + 1) % 2 == 0):

nb += (

crystal[(a - 1) % lattice, (b - 1) % lattice]

+ crystal[(a + 1) % lattice, (b - 1) % lattice]

)

elif self.mesh == "hex":

nb += (

crystal[(a + 1) % lattice, (b + 1) % lattice]

+ crystal[(a - 1) % lattice, (b + 1) % lattice]

)

if self.prop == "ferro":

metric = 2 \* s \* nb

elif self.prop == "antiferro":

metric = -2 \* s \* nb

if metric < 0:

s \*= -1

elif rand() < exp(-metric \* beta):

s \*= -1

crystal[a, b] = s

return crystal

Essentially, for the selected lattice point, we sum the nearest neighbors and then multiply by the lattice spin and, again, 1 or -1 (depending on the material modelled). The choice is then made, based on the metropolis algorithm to flip, or not, the chosen point’s spin.

To the class, a wrapper is being added to iterate over the selected iterations and plot the results appropriately.

The entire class is as follows:

from numpy import exp, meshgrid, sum, linspace, diff, ones

from numpy.random import rand, randint

import matplotlib.pyplot as plt

from matplotlib import colors

import matplotlib.patches as mpatches

from matplotlib.collections import PatchCollection

class Ising:

# initialization

def \_\_init\_\_(

self, temperature, lattice, iterations, mesh="ortho", prop="ferro", plots=True

):

self.temperature = temperature

self.lattice = lattice

self.iterations = iterations

if plots:

if plots == "animate":

self.plots = True

self.animate = True

else:

self.plots = True

else:

self.plots = False

if mesh == "ortho":

self.mesh = "ortho"

elif mesh == "hex":

self.mesh = "hex"

else:

raise Exception("Incorrect mesh")

if prop == "ferro":

self.prop = "ferro"

elif prop == "antiferro":

self.prop = "antiferro"

else:

raise Exception("Incorrect property")

# energy of the crystal

def E(self, crystal, lattice):

energy = 0

for i in range(len(crystal)):

for j in range(len(crystal)):

S = crystal[i, j]

nb = (

crystal[(i + 1) % lattice, j]

+ crystal[i, (j + 1) % lattice]

+ crystal[(i - 1) % lattice, j]

+ crystal[i, (j - 1) % lattice]

)

if self.mesh == "hex" and ((i + 1) % 2 == 0):

nb += (

crystal[(i - 1) % lattice, (j - 1) % lattice]

+ crystal[(i + 1) % lattice, (j - 1) % lattice]

)

elif self.mesh == "hex":

nb += (

crystal[(i + 1) % lattice, (j + 1) % lattice]

+ crystal[(i - 1) % lattice, (j + 1) % lattice]

)

if self.prop == "ferro":

energy += -nb \* S

elif self.prop == "antiferro":

energy += nb \* S

if self.mesh == "hex":

return energy / 6.0

elif self.mesh == "ortho":

return energy / 4.0

# magnetization of crystal

def M(self, crystal):

mag = sum(crystal)

return mag

# monte carlo steps

def mcStep(self, crystal, lattice, beta):

for i in range(lattice):

for j in range(lattice):

a = randint(0, lattice)

b = randint(0, lattice)

s = crystal[a, b]

nb = (

crystal[(a + 1) % lattice, b]

+ crystal[a, (b + 1) % lattice]

+ crystal[(a - 1) % lattice, b]

+ crystal[a, (b - 1) % lattice]

)

if self.mesh == "hex" and ((a + 1) % 2 == 0):

nb += (

crystal[(a - 1) % lattice, (b - 1) % lattice]

+ crystal[(a + 1) % lattice, (b - 1) % lattice]

)

elif self.mesh == "hex":

nb += (

crystal[(a + 1) % lattice, (b + 1) % lattice]

+ crystal[(a - 1) % lattice, (b + 1) % lattice]

)

if self.prop == "ferro":

metric = 2 \* s \* nb

elif self.prop == "antiferro":

metric = -2 \* s \* nb

if metric < 0:

s \*= -1

elif rand() < exp(-metric \* beta):

s \*= -1

crystal[a, b] = s

return crystal

# simulate the crystal

def simulate(self):

E = 0

M = 0

crystal = 2 \* randint(2, size=(self.lattice, self.lattice)) - 1

if self.plots and (not self.animate) and self.mesh == "hex":

f = plt.figure(figsize=(15, 15))

self.configPlotHex(f, crystal, 0, self.lattice, 1)

elif self.plots and (not self.animate) and self.mesh == "ortho":

f = plt.figure(figsize=(15, 15))

self.configPlot(f, crystal, 0, self.lattice, 1)

elif self.plots and (self.animate) and self.mesh == "hex":

self.savePlotHex(crystal, 0, self.lattice)

elif self.plots and (self.animate) and self.mesh == "ortho":

self.savePlot(crystal, 0, self.lattice)

for i in range(self.iterations):

self.mcStep(crystal, self.lattice, 1.0 / self.temperature)

if self.plots and (not self.animate) and self.mesh == "hex":

if i == 1:

self.configPlotHex(f, crystal, i, self.lattice, 2)

if i == 4:

self.configPlotHex(f, crystal, i, self.lattice, 3)

if i == 32:

self.configPlotHex(f, crystal, i, self.lattice, 4)

if i == 100:

self.configPlotHex(f, crystal, i, self.lattice, 5)

if i == self.iterations - 1:

self.configPlotHex(f, crystal, i, self.lattice, 6)

elif self.plots and (not self.animate) and self.mesh == "ortho":

if i == 1:

self.configPlot(f, crystal, i, self.lattice, 2)

if i == 4:

self.configPlot(f, crystal, i, self.lattice, 3)

if i == 32:

self.configPlot(f, crystal, i, self.lattice, 4)

if i == 100:

self.configPlot(f, crystal, i, self.lattice, 5)

if i == self.iterations - 1:

self.configPlot(f, crystal, i, self.lattice, 6)

elif self.plots and (self.animate) and self.mesh == "hex":

self.savePlotHex(crystal, i, self.lattice)

elif self.plots and (self.animate) and self.mesh == "ortho":

self.savePlot(crystal, i, self.lattice)

E += self.E(crystal, self.lattice)

M += self.M(crystal)

if self.plots:

plt.show()

return [E, M]

# configure hex plot

def configPlotHex(self, f, crystal, i, lattice, n\_):

nx = lattice

ny = lattice

cmap = colors.ListedColormap(["black", "white"])

bounds = [0, 1, 2]

norm = colors.BoundaryNorm(bounds, cmap.N)

x = linspace(0, 1, nx)

y = linspace(0, 1, ny)

dx = diff(x)[0]

patches = []

for k in x:

for n, j in enumerate(y):

if n % 2:

polygon = mpatches.RegularPolygon([k - dx / 2.0, j], 6, 0.6 \* dx)

else:

polygon = mpatches.RegularPolygon([k, j], 6, 0.6 \* dx)

patches.append(polygon)

collection = PatchCollection(patches, cmap=cmap, norm=norm, alpha=1.0)

sp = f.add\_subplot(3, 3, n\_)

sp.set\_xticks([])

sp.set\_yticks([])

sp.add\_collection(collection)

collection.set\_array(crystal.ravel())

sp.set\_title("Time=%d" % i)

sp.axis("tight")

# make a single plot

def configPlot(self, f, crystal, i, lattice, n\_):

X, Y = meshgrid(range(lattice), range(lattice))

sp = f.add\_subplot(3, 3, n\_)

plt.setp(sp.get\_yticklabels(), visible=False)

plt.setp(sp.get\_xticklabels(), visible=False)

plt.pcolormesh(X, Y, crystal, cmap=plt.cm.binary)

plt.title("Time=%d" % i)

plt.axis("tight")

# configure hex plot

def savePlotHex(self, crystal, i, lattice):

nx = lattice

ny = lattice

cmap = colors.ListedColormap(["black", "white"])

bounds = [0, 1, 2]

norm = colors.BoundaryNorm(bounds, cmap.N)

x = linspace(0, 1, nx)

y = linspace(0, 1, ny)

dx = diff(x)[0]

patches = []

for k in x:

for n, j in enumerate(y):

if n % 2:

polygon = mpatches.RegularPolygon([k - dx / 2.0, j], 6, 0.6 \* dx)

else:

polygon = mpatches.RegularPolygon([k, j], 6, 0.6 \* dx)

patches.append(polygon)

collection = PatchCollection(patches, cmap=cmap, norm=norm, alpha=1.0)

fig, ax = plt.subplots(1, 1)

ax.set\_xticks([])

ax.set\_yticks([])

ax.add\_collection(collection)

collection.set\_array(crystal.ravel())

ax.set\_title("Time=%d" % i)

ax.axis("tight")

fig.savefig(f"Animation/{i:04d}.png")

# make a single plot

def savePlot(self, crystal, i, lattice):

fig, ax = plt.subplots(1, 1)

ax.set\_xticks([])

ax.set\_yticks([])

X, Y = meshgrid(range(lattice), range(lattice))

ax.pcolormesh(X, Y, crystal, cmap=plt.cm.binary)

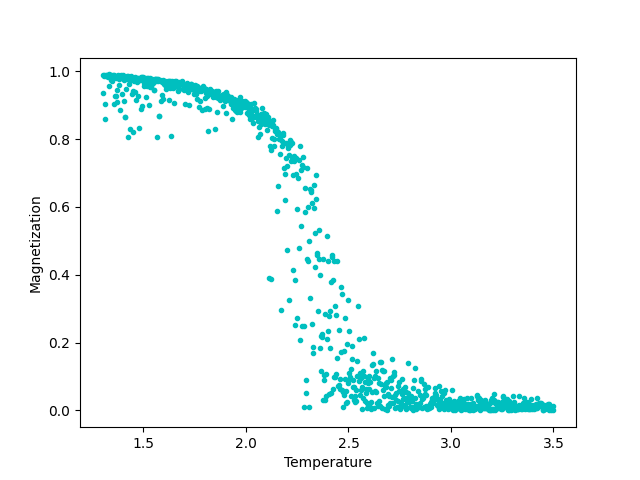
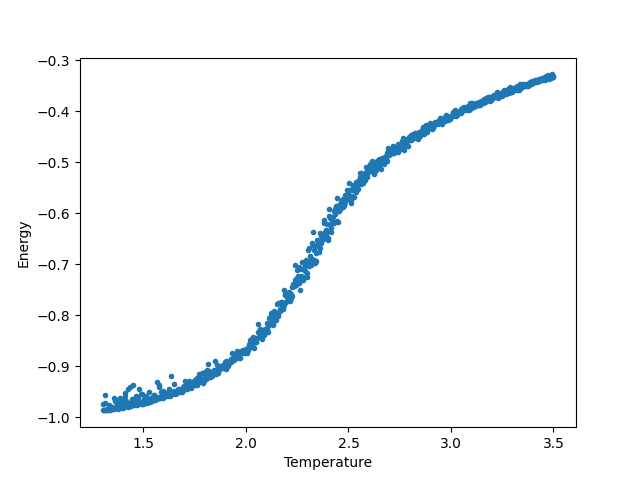
ax.set\_title("Time=%d" % i)

ax.axis("tight")

fig.savefig(f"Animation/{i:04d}.png")

## Ferromagnetic Results

For the ferromagnetic material, the resulting Energy vs Temperature and Magnetization vs Temperature plots, in an orthogonal lattice were:



We clearly see a phase transition at a Temperature of around 2.25 (A.U.). This is the for the system. Both curves are sigmoid, as expected.

Snapshots of the crystal at different times throughout the MCMC are shown below for 2 different temperatures (above and below ), showing in image, the failure to magnetize beyond .

A picture containing map, text

Description automatically generated

Snapshot of the crystal at temperature 1.3A.U. (below ). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. Overtime all the lattice points align their spins with only thermal fluctuations remaining.

A picture containing pattern, black and white, monochrome, fabric

Description automatically generated

Snapshot of the crystal at temperature 3.0A.U. (above ). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. No matter the number of iterations allowed, the lattice will never reach a state of full alignment, like in the lower temperature.

The equivalent plots and images in the hexagonal lattice are shown below:

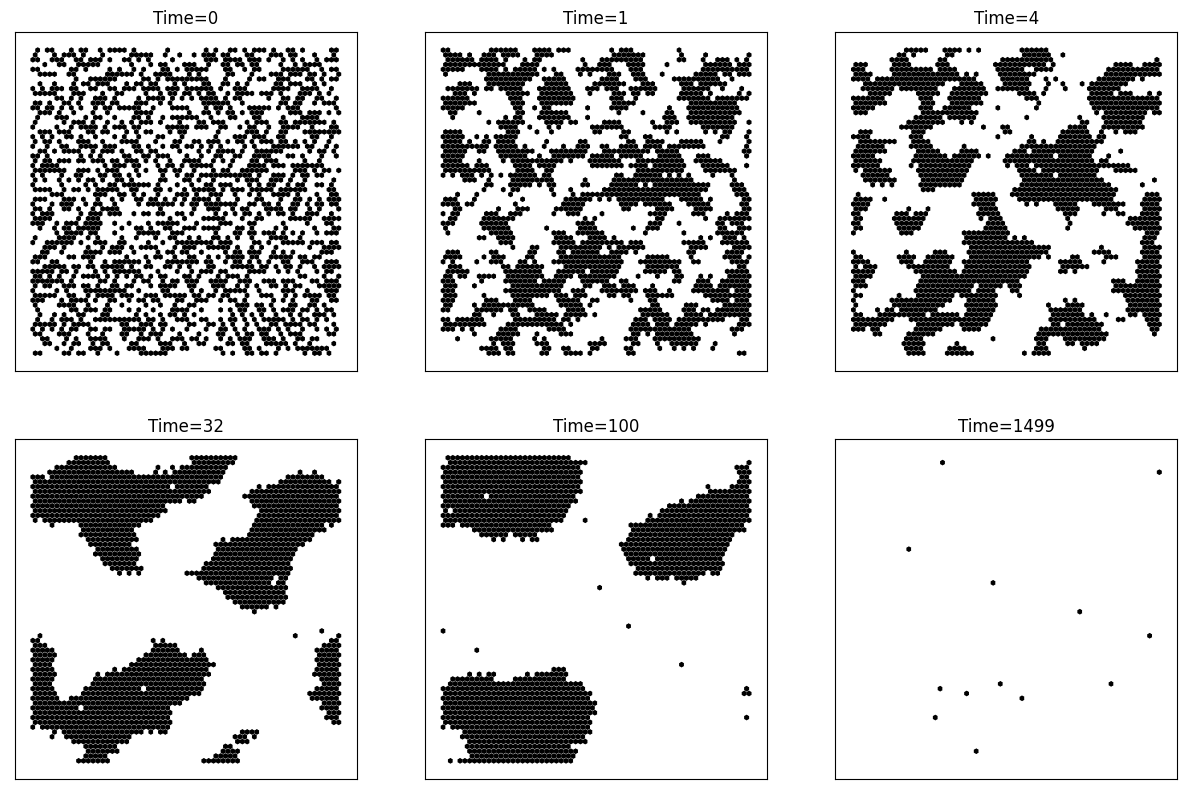
A picture containing text, diagram, plot, screenshot

Description automatically generatedA picture containing text, screenshot, diagram, map

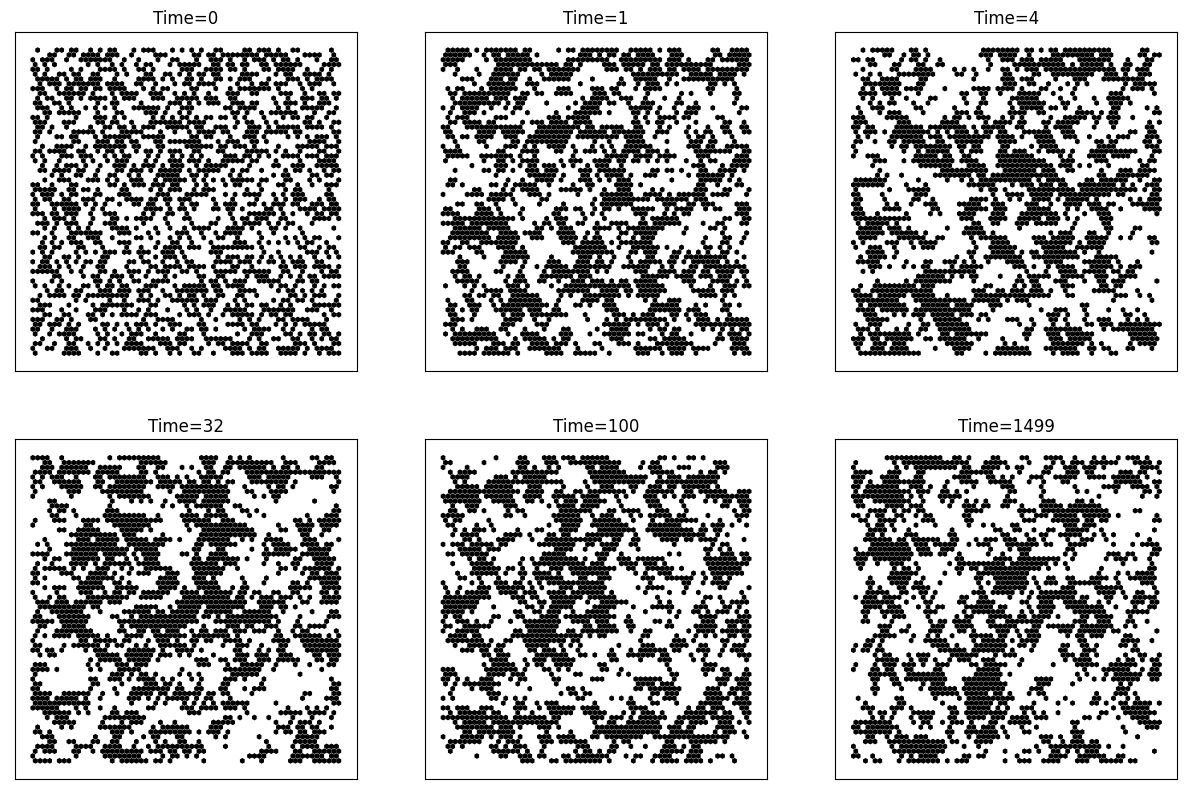
Description automatically generated

We again see a phase transition, this time at a Temperature of around 3.75A.U. This is the for the system. Both curves are sigmoid, as expected.

Snapshots of the crystal at different times throughout the MCMC are shown below for 2 different temperatures (above and below ), showing in image, the failure to magnetize beyond .



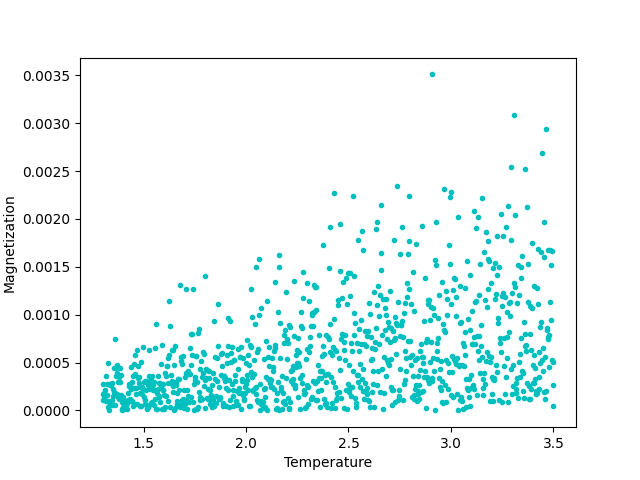
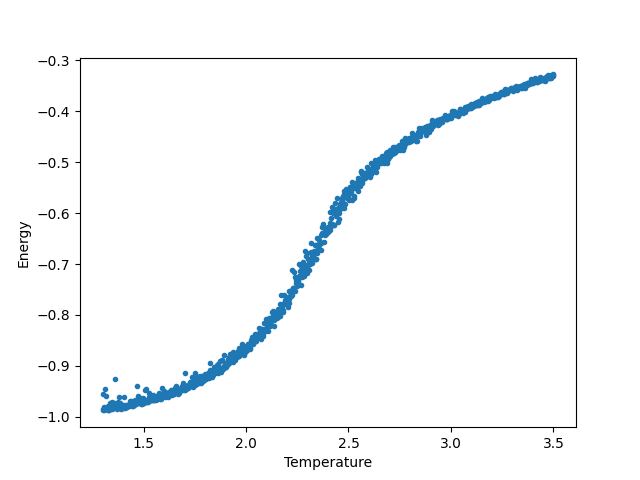
Snapshot of the crystal at temperature 2A.U. (below ). Each black pixel represents a point in the lattice with spin up and each white one a point with spin down. Overtime all the lattice points align their spins with only thermal fluctuations remaining.



Snapshot of the crystal at temperature 5.0A.U. (above ). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. No matter the number of iterations allowed, the lattice will never reach a state of full alignment, like in the lower temperature.

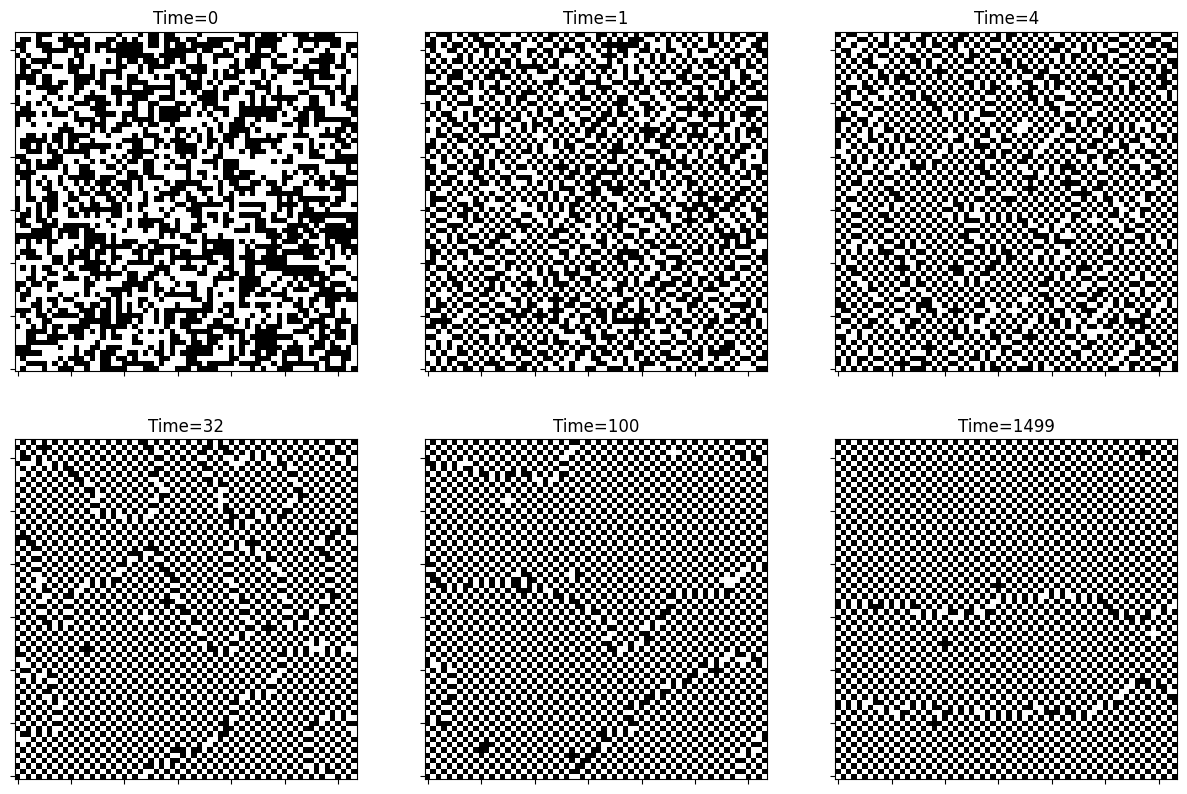
## Antiferromagnetic Results

For the antiferromagnetic material, the resulting Energy vs Temperature and Magnetization vs Temperature plots, in an orthogonal lattice were:

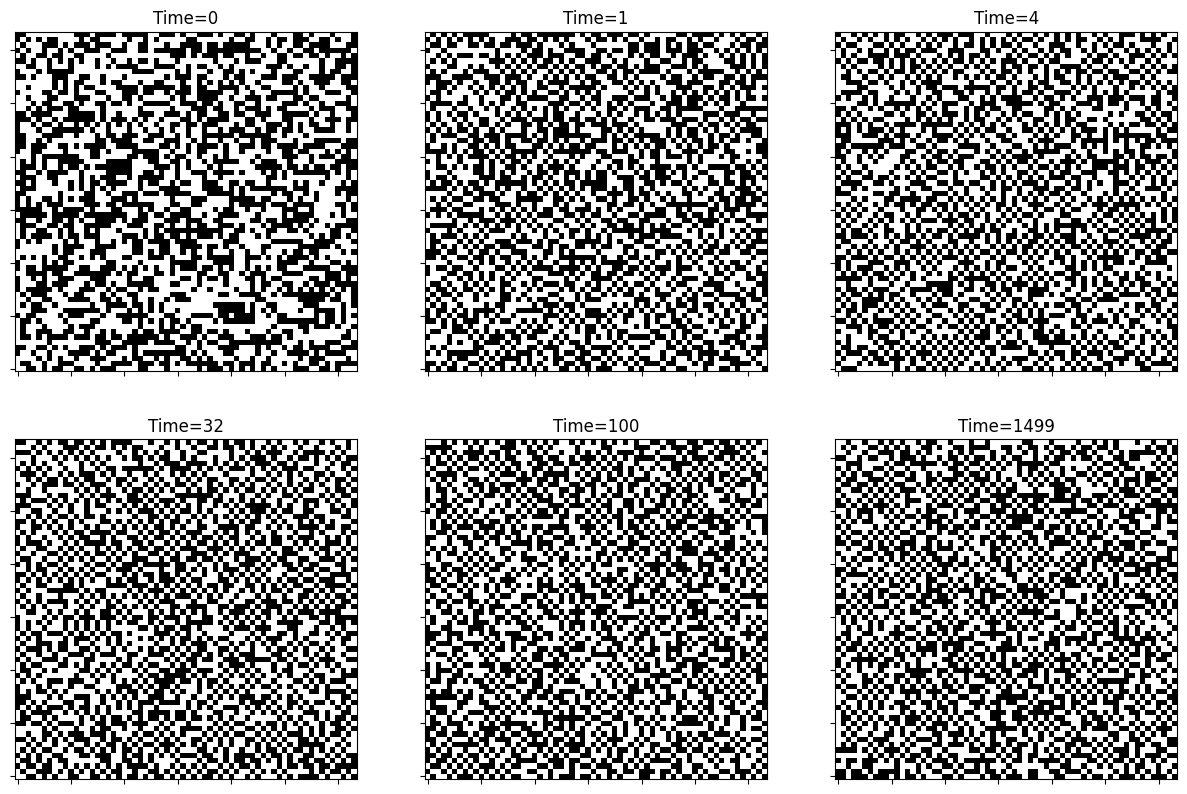


We see a phase transition at a Temperature of around 2.25 (A.U.). This is the for the system. Only the Energy curve is sigmoid, therefore only that can provide us with an estimate for .

Snapshots of the crystal at different times throughout the MCMC are shown below for 2 different temperatures (above and below ), showing in image, the failure to present antiferromagnetic property beyond .

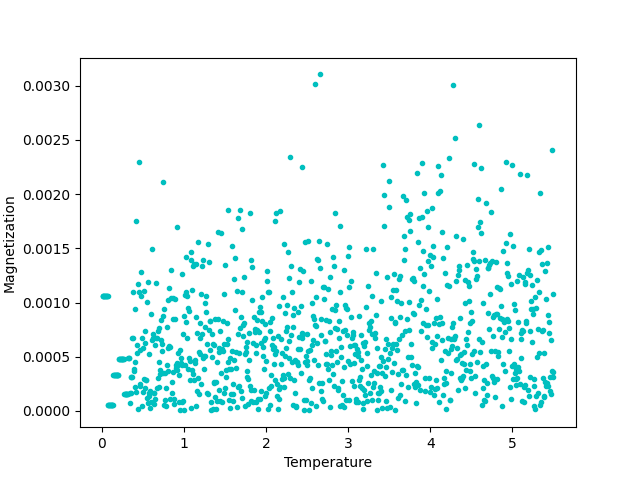
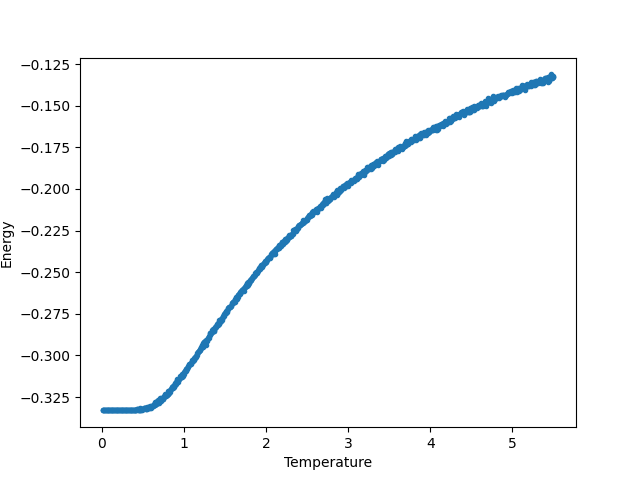


Snapshot of the crystal at temperature 1.3A.U. (below ). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. Overtime all the lattice points create a checkerboard pattern, inversely aligning their spins with only thermal fluctuations remaining. The fluctuations themselves are more patterned than in the ferromagnetic scenario, due to the nature of the phenomenon.



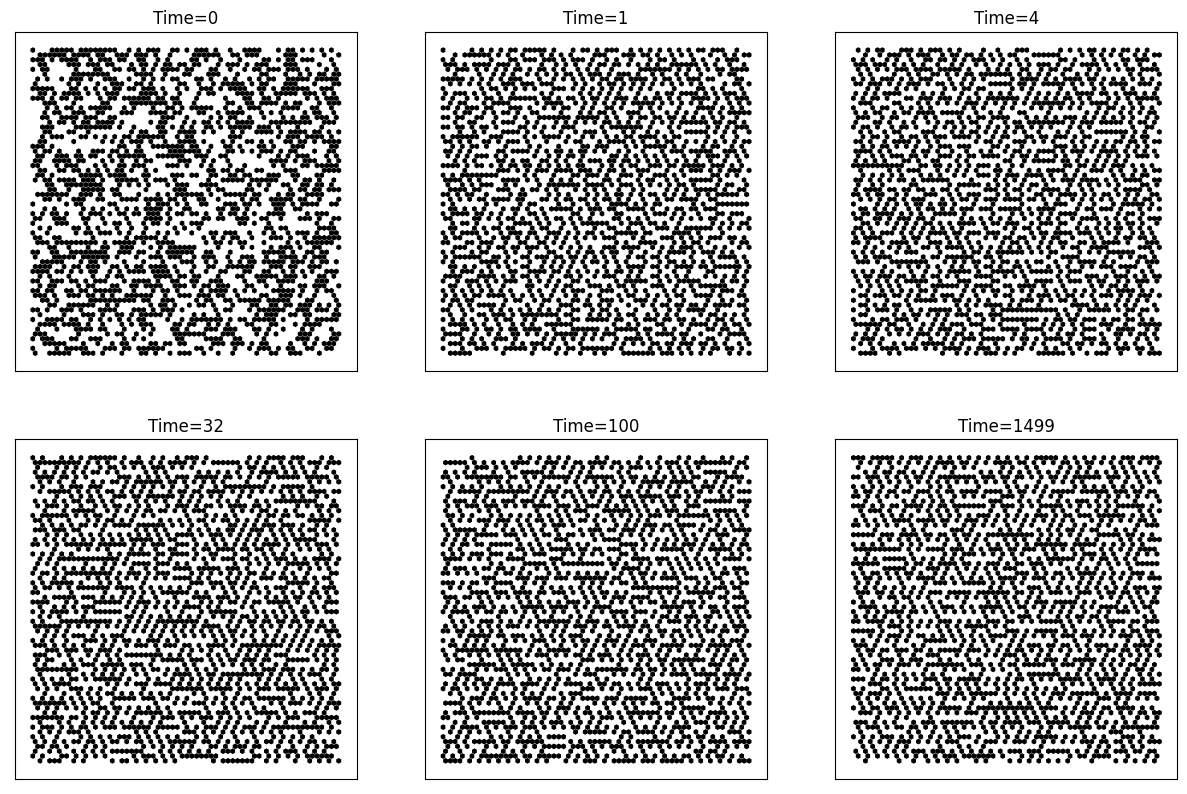
Snapshot of the crystal at temperature 3.0A.U. (above ). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. No matter the number of iterations allowed, the lattice will never reach a state of almost pure checkerboard pattern, like in the lower temperature.

The equivalent plots and images in the hexagonal lattice are shown below:



We again see a phase transition, this time at a Temperature of around 0.7A.U. This is the analogue of the for the system. Neither of the curves is sigmoid, which hints us to the fact that the antiferromagnetism is only loosely defined in the hexagonal lattice. This is more easily visualized at the snapshots provided below.

Snapshots of the crystal at different times throughout the MCMC are shown below for 2 different temperatures (above and below “”), showing in image, the problem of creating checkerboards in a hexagonal lattice.



Snapshot of the crystal at temperature 0.1A.U. (below “”). Each black pixel represents a point in the lattice with spin up and each white one a point with spin down. Overtime a fractal-like pattern does emerge.



Snapshot of the crystal at temperature 5.0A.U. (above “”). Each black pixel represents a point in the lattice with spin down and each white one a point with spin up. No matter the number of iterations allowed, the lattice will never reach a state of fractal-like pattern, like in the lower temperature.

## Extra Analysis of the Hexagonal Lattice Antiferromagnetism

Analyzing further the hexagonal lattice antiferromagnetic simulation, we get the following plot for the energy of the crystal:

A picture containing text, screenshot, plot, line

Description automatically generated

We see that having fit both sides of the curve with a line, they intercept at a temperature of 0.69A.U. This is a different temperature than the expected Néel (3.75A.U.), since the lattice cannot harvest the full potential of antiferromagnetism, due to its shape. In other words, there is no way to tile the space in black and white hexagons with only alternating black and white colors between edges.

## Discussion

# References

1. Ising, E. (1925). Beitrag zur Theorie des Ferromagnetismus. Zeitschrift für Physik, 31(1), 253-258.
2. Onsager, L. (1944). Crystal statistics. I. A two-dimensional model with an order-disorder transition. Physical Review, 65(3-4), 117-149.
3. Yeomans, J. M. (1992). Statistical Mechanics of Phase Transitions. Oxford University Press.
4. Binder, K., & Heermann, D. W. (2010). Monte Carlo Simulation in Statistical Physics: An Introduction (5th ed.). Springer.
5. Goldenfeld, N. (1992). Lectures on Phase Transitions and the Renormalization Group. Addison-Wesley.
6. Chaikin, P. M., & Lubensky, T. C. (1995). Principles of Condensed Matter Physics. Cambridge University Press.
7. Sachdev, S. (2011). Quantum Phase Transitions (2nd ed.). Cambridge University Press.
8. Chayes, L., Chayes, J., Kotecký, R., & Ruelle, D. (1982). A mathematical theory of Griffiths phases: The case of one-dimensional quantum systems. Communications in Mathematical Physics, 79(3), 581-607.
9. Selke, W. (1988). The Ising Model in the critical region. Progress in Physics, 49(11), 517-608.
10. Hohenberg, P. C., & Halperin, B. I. (1977). Theory of dynamic critical phenomena. Reviews of Modern Physics, 49(3), 435-479.
11. Swendsen, R. H., & Wang, J.-S. (1987). Nonuniversal critical dynamics in Monte Carlo simulations. Physical Review Letters, 58(2), 86-88.
12. Wolff, U. (1989). Collective Monte Carlo updating for spin systems. Physical Review Letters, 62(4), 361-364.
13. Baxter, R. J. (1982). Exactly solved models in statistical mechanics. Academic Press.
14. Kosterlitz, J. M., & Thouless, D. J. (1973). Ordering, metastability and phase transitions in two-dimensional systems. Journal of Physics C: Solid State Physics, 6(7), 1181-1203.
15. Kardar, M. (2007). Statistical Physics of Fields. Cambridge University Press.