



## MU4PY115: M1 Project

# Disordered and quenched random systems

### Abstract

In this study, we explore disordered and quenched random systems through a combined numerical and analytical approach. Our investigation encompasses the study of random walks (both isotropic and anisotropic) and the Ising Model. By employing Monte Carlo simulations for the latter, our goal is to reproduce the analytical results through computational experiments. We aim to verify the accuracy of our simulations by comparing the results obtained numerically with the analytical predictions.

This project can be used as groundwork for more complex random systems or for application in the real world like the self-avoiding walk used in polymer physics or the correlated random walks used to model animal movements.

BESSERVE Maxime 28629795  
BOISTAY Lucas 3809291  
DONACHIE Samuel 28708507  
ROY Thomas 28711316  
YERLE Arsène 21106813

February 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Random Walk</b>	<b>3</b>
2.1	Overview . . . . .	3
2.2	Isotropic RW . . . . .	3
2.3	Link with diffusion . . . . .	6
2.4	The diffusion equation . . . . .	8
2.5	Anisotropic RW . . . . .	8
<b>3</b>	<b>Monte Carlo Methods</b>	<b>13</b>
3.1	Overview . . . . .	13
3.2	The Metropolis Algorithm . . . . .	14
3.3	The Ising Model . . . . .	14
3.3.1	Sample to sample fluctuation . . . . .	15
3.3.2	Results . . . . .	15
3.4	Different interaction constants $J$ . . . . .	17
3.4.1	Constant $J$ for the entire system . . . . .	17
3.4.2	Different $J$ for each pair of spins . . . . .	18
<b>4</b>	<b>Conclusion</b>	<b>19</b>

# 1 Introduction

In the realm of physics, disordered systems represent a fascinating area of study. These systems are characterized by the absence of correlations among constituent particles, resulting in each particle behaving independently and exhibiting random behavior. Despite the apparent chaos at the individual particle level, a remarkable phenomenon emerges when these systems consist of a large number of particles: their behavior becomes predictable through the principles of statistical physics. This project endeavors to explore and understand this intriguing aspect of disordered systems by employing two fundamental principles of physics: random walk and Monte Carlo methods. Through careful examination and analysis using these principles, we aim to uncover the underlying patterns and behaviors inherent in disordered systems, shedding light on their complex dynamics and contributing to our broader understanding of those random phenomena.

# 2 Random Walk

## 2.1 Overview

A random walk is a stochastic process, indicating it involves randomness. The goal of a random walk is to generate a path by taking numerous steps randomly selected from a probability distribution. Each position in the random walk is determined by the recurrent sequence:

$$X_{k+1} = X_k + R_k$$

with  $R_k$  the step taken and  $X_k$  the position after  $k$  steps. This random walk model is named a «memory-less process» because the position  $X_{k+1}$  is not influenced by the previous position  $X_k$ .

## 2.2 Isotropic RW

In the initial segment, we examine a one-dimensional random walk with discrete steps. Each step can take values of +1 or -1 with equal probabilities of 50%, ensuring equal chances of moving right or left. Additionally, we establish the following initial conditions and parameters for our model:

- $X_0$  for the initial position
- $t_0$  for the initial time
- $n$  : the number of steps
- $m$  : the number of random walk

Initially, we plot a random walk consisting of  $m = 1000$  random walks, each with  $n = 2000$  steps. Additionally, we set the initial position  $X_0 = 0$ .

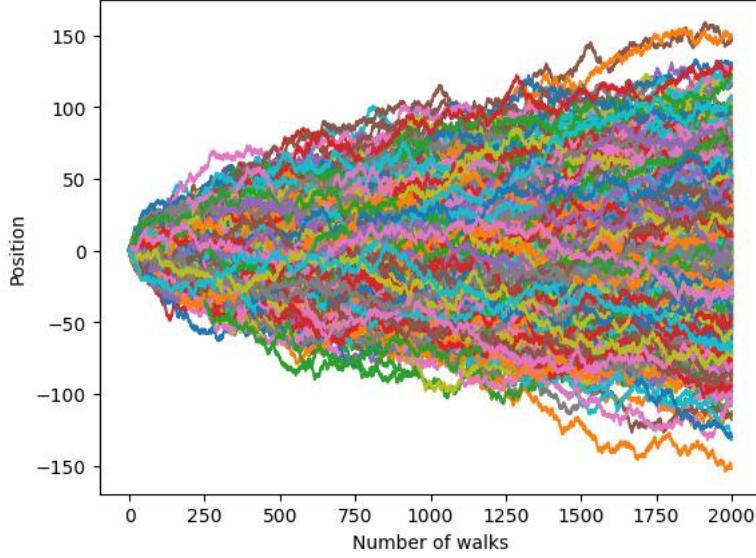


Figure 1: RW with conditions (1000 RW and 2000 steps)

We can observe various final positions for a given initial condition. The distribution of final positions demonstrates a certain symmetry between -60 and 60.

To extract statistical information from this result, our first step is to plot the histogram of final positions for a single random walk. The resulting graph is as follows:

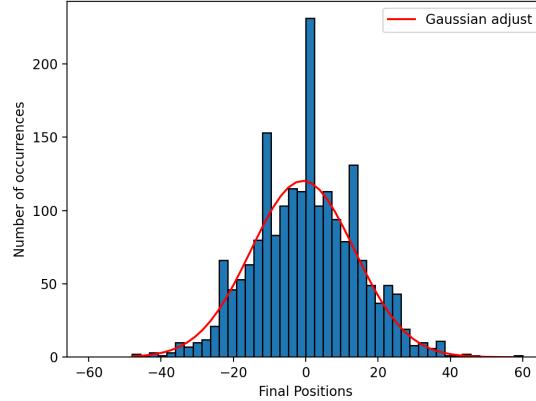


Figure 2: RW with conditions (1000 RW and 2000 lengh)

We observe a Gaussian distribution for the final positions. Therefore, we seek a distribution similar to:

$$N(x, n) = \frac{1}{\sqrt{2\pi V}} e^{-\frac{(x-\langle x \rangle)^2}{2V}}$$

Where  $\langle X \rangle$  and  $V$  are respectively the variance and the standard deviation, which depend at first glance on the number of steps  $n$ . In this histogram, the mean value of the final positions is approximately 0.04, which is close to the initial value ( $X_0 = 0$ ). This prompts the question of whether the mean value of the final positions is consistently equal to the initial position  $X_0$ .

To verify this hypothesis, we run a number of random walks with the same conditions as the previous one but with different initial position at each run. Then, we calculate the mean value for each walk and plot the curve  $\langle X \rangle = f(X_0)$ . The resulting graph is as follows :

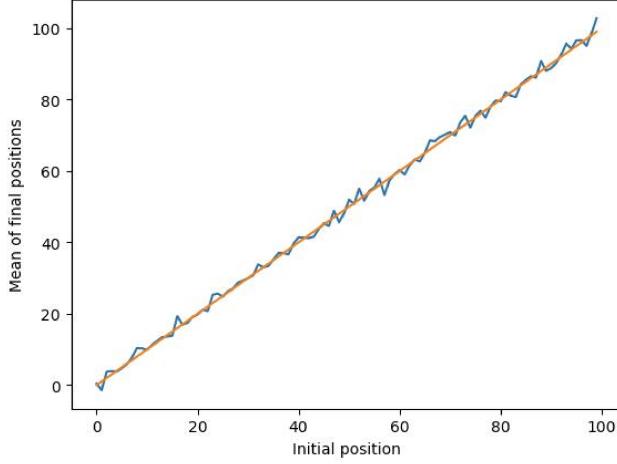


Figure 3: mean of final position VS initial conditions

The previous curve represents the identity function (confirmed by linear curve-fitting). The mean value of the final positions consistently equals the initial position. This finding holds significant implications within the realm of random walks.

Moving forward, we proceed to investigate the variance of the distribution. By fixing  $X_0 = 0$ , we plot histograms for various values of the number  $n$ . Subsequently, we calculate the variance and illustrate its evolution as a function of  $n$ .

The resulting graph is displayed below:

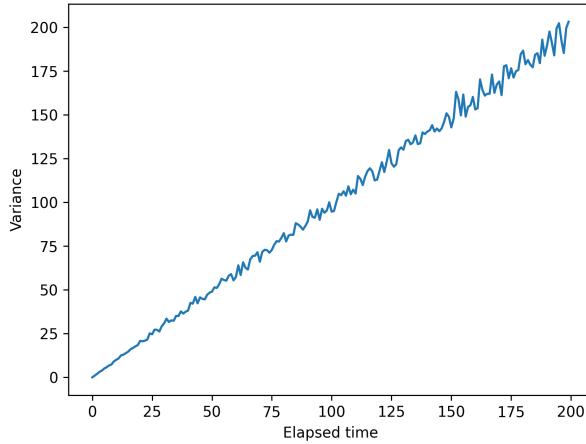


Figure 4: Variance VS time

This graph presents a linear trend. Instead of working with the number of steps  $n$ , we can work with time  $t$ . If we call  $\tau$  the time to complete one step, the total duration is equal to  $t = \tau n$ .

Einstein has shown the following relation between duration and the mean square of displacements:

$$\langle (X - X_0)^2 \rangle = 2dD(t - t_0)$$

With:

- $\langle (X - X_0)^2 \rangle$  : the mean square displacement between the final position and the initial position.
- $d$  : the dimension of the random walk.
- $\tau$  : the duration of one step.
- $D$  : the diffusion coefficient.

With the initial conditions ( $t_0 = 0, X_0 = 0$ ), the mean position is zero. Therefore, we can equate the variance to the mean square displacement. Additionally, in our simulation,  $d = 2$ . This simplifies the previous relation to:

$$V(X) = 2Dt$$

The random walk we obtain verifies this linear relation, and we can find the value of the diffusion coefficient.

Remark: If we denote  $l$  as the distance between two steps, the diffusion coefficient for the 1D model can be written as:

$$D = \frac{l^2}{2\tau}$$

The relation shows the link between the diffusion coefficient  $D$  and the parameters of the random walk. However, in our random walk, the value of the distance between the steps is equal to one. In fact, the position is given in units of  $l$ .

## 2.3 Link with diffusion

We can futher explore the connections between the statistic of random walk and the diffusion.

For many values of the number of steps  $n$ , we plot the histogram of final positions along with their Gaussian fit. We use normalization to fit a normal distribution.

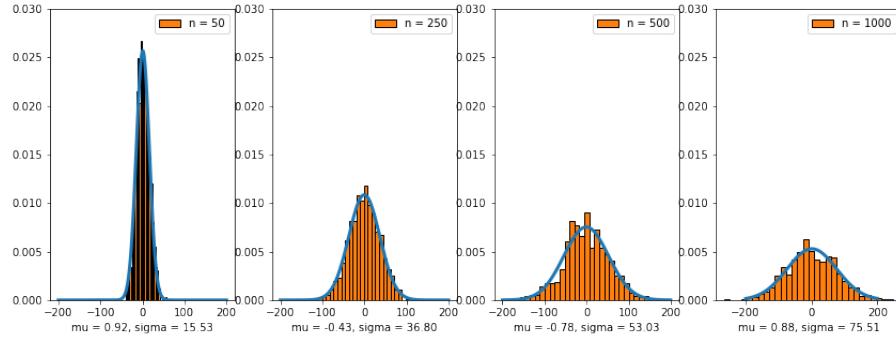


Figure 5: Normal distribution fit for different values of  $n$

As we have shown in the previous section, the width of the Gaussian increases because the standard deviation is proportional to the square root of  $n$ . Now, we plot the different fits for different values of  $n$  on the same graph.

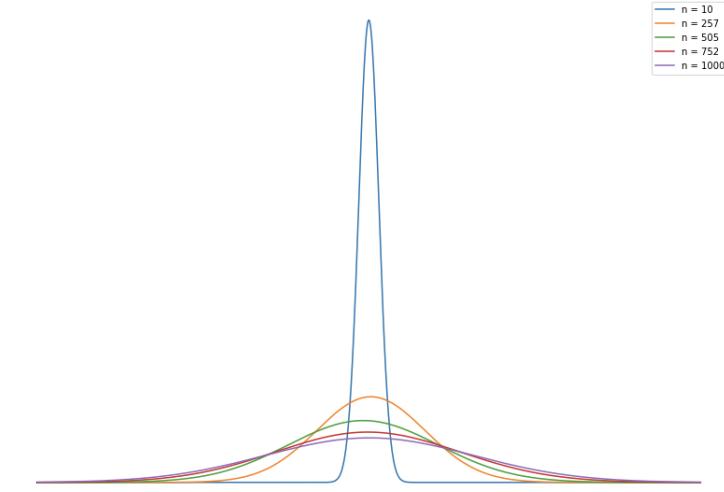


Figure 6: Normal distribution fit for different values of  $n$

This function of two variables  $n$  and  $t$  is similar to the Green's solution of the diffusion equation in one dimension.

$$G(x, t) = \frac{1}{\sqrt{4\pi D t}} e^{-\frac{(x-X_0)^2}{4Dt}}$$

To ensure the accuracy of this observation, we can plot the graph of the function  $G$  for different values of time  $t$  and compare it with our result. We obtained the following graph, and we noticed that the shape of these two curves is the same.

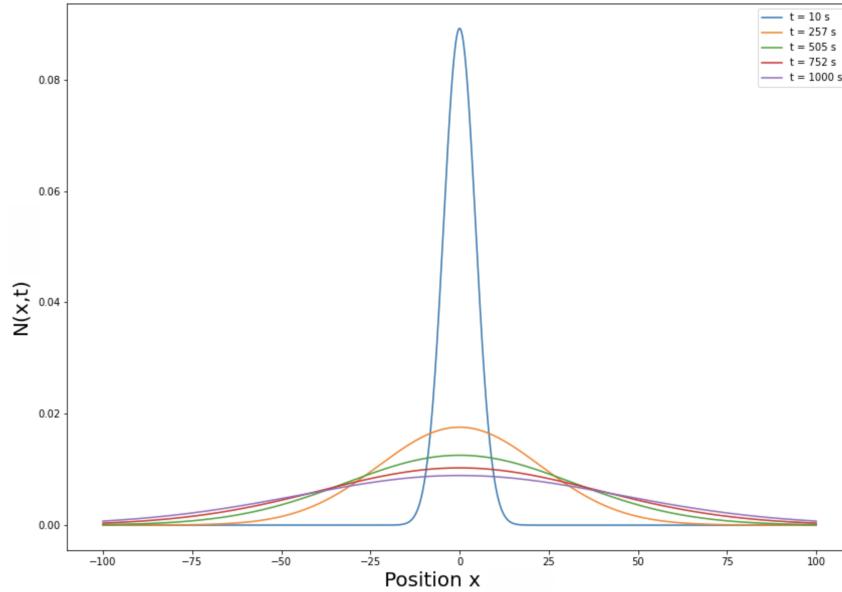


Figure 7:  $G(x, t)$  function of  $x$  for different values of time  $t$

Now, let's introduce the diffusion equation where the function  $G(x, t)$  is the fundamental solution.

## 2.4 The diffusion equation

We want to derive the diffusion equation from a random walk. First, we take a random walk, with  $i$  steps of length  $a$ , and  $N_i$  particles for each  $x_i$ . To get the diffusion equation, we have to go to the continuous limit such as we have  $N \gg 1$  positions in a length of  $[x, x+dx]$  with  $dx \ll 1$ . With that approximation, we can have a concentration of particles  $C_i$ .

With that you have the relation  $C(x_i) = \frac{\langle N_i \rangle}{a}$ . Let  $x_0$ ,  $x_1$  and  $x_2$  with  $N_0$ ,  $N_1$  and  $N_2$  particles. After a time  $\tau$ , all the  $N_1$  particles in  $x_1$  will leave and on average, half of the particles from  $x_0$  and  $x_2$  will go to  $x_1$ .

You have  $\langle N_1 \rangle(\tau) = \frac{1}{2}\langle N_0 \rangle + \frac{1}{2}\langle N_2 \rangle$

$$\begin{aligned} \frac{\partial C(x_1)}{\partial \tau} &= \frac{\langle N_1 \rangle(\tau) - \langle N_1 \rangle(0)}{a\tau} \\ \frac{\partial C(x_1)}{\partial \tau} &= \frac{\langle N_0 \rangle + \langle N_2 \rangle - 2\langle N_1 \rangle}{2a\tau} \\ \frac{\partial C(x_1)}{\partial \tau} &= \frac{C(x_0) + C(x_2) - 2C(x_1)}{2\tau} \\ \frac{\partial C(x_1)}{\partial \tau} &= \frac{a^2}{2\tau} \frac{C(x_1 - a) + C(x_1 + a) - 2C(x_1)}{a^2} \\ \frac{\partial C(x_1)}{\partial \tau} &= D \frac{\partial^2 C(x_1)}{\partial x^2} \end{aligned}$$

when  $a \rightarrow 0$  where  $D$  is the diffusion coefficient. We can see that  $\tau$  must not be too small in order to keep  $D$  finite, we thus have a condition between the space and time grid.

## 2.5 Anisotropic RW

To finish our work on random walks, we then decided to change one of the parameter of the studied model: the probability of advancing or taking a step back.

For each position  $X_k$ , there will be a fixed probability to move forward or backwards. This probability will be given randomly at the beginning of the simulation, we can choose whatever probability distribution we want.

For anisotropic random walk, there also is a diffusion equation.

So, we then take the same hypothesis as before, but with a probability of  $p$  and  $q=1-p$  we have:

$$\langle N_1 \rangle(\tau) = p\langle N_0 \rangle + (1-p)\langle N_2 \rangle$$

Then, same as before we have:

$$\begin{aligned} \frac{\partial C(x_1)}{\partial \tau} &= \frac{\langle N_1 \rangle(\tau) - \langle N_1 \rangle(0)}{a\tau} \\ \frac{\partial C(x_1)}{\partial \tau} &= \frac{p\langle N_0 \rangle + (1-p)\langle N_2 \rangle - \langle N_1 \rangle}{a\tau} \\ \frac{\partial C(x_1)}{\partial \tau} &= \frac{(\frac{1}{2} + p)\langle N_0 \rangle + (\frac{1}{2} - p)\langle N_2 \rangle - \langle N_1 \rangle - \frac{1}{2}\langle N_0 \rangle + \frac{1}{2}\langle N_2 \rangle}{a\tau} \end{aligned}$$

$$\frac{\partial C(x_1)}{\partial \tau} = \frac{\langle N_0 \rangle + \langle N_2 \rangle - 2\langle N_1 \rangle}{2a\tau} - \frac{(p - \frac{1}{2})(\langle N_2 \rangle - \langle N_0 \rangle)}{a\tau}$$

We recognize the first term which is  $D \frac{\partial^2 C(x_1)}{\partial x^2}$  when  $a \rightarrow 0$ . We can also write  $p - \frac{1}{2}$  as  $2(p - q)$

$$\begin{aligned} \frac{\partial C(x_1)}{\partial \tau} &= D \frac{\partial^2 C(x_1)}{\partial x^2} - \frac{(p - q)(\langle N_2 \rangle - \langle N_0 \rangle)}{2a\tau} \\ \frac{(p - q)(\langle N_2 \rangle - \langle N_0 \rangle)}{2a\tau} &= \frac{(p - q)(C(x_2) - C(x_0))}{2\tau} = \frac{a(p - q)}{\tau} \frac{(C(x + a) - C(x - a))}{2a} \end{aligned}$$

When  $a \rightarrow 0$ ,

$$\frac{(C(x_1 + a) - C(x_1 - a))}{2a} = \frac{\partial C(x_1)}{\partial x}$$

We thus have:

$$\frac{\partial C(x_1)}{\partial \tau} = D \frac{\partial^2 C(x_1)}{\partial x^2} - v \frac{\partial C(x_1)}{\partial x}$$

$v = \frac{a(p-q)}{\tau}$  can be interpreted as the bias velocity, and  $\frac{\partial C(x_1)}{\partial x}$  as the convective term. We notice that  $\frac{v}{D}$  diverges as  $\frac{1}{a}$  in the continuum limit. To fix this issue,  $p-q$  must be proportional to  $a$ . This equation is also called drift diffusion equation.

For our simulation, we'll employ a box probability distribution ranging from 0 to 1. This entails assigning each position a probability  $p$  of moving forward and  $(1-p)$  of taking a step back, where  $p$  can assume any value between 0 and 1 with equal likelihood of occurrence.

Other than altering the probability of moving from one position to another, we keep all other parameters unchanged.

Let's proceed to plot the potential paths taken with these specified parameters.

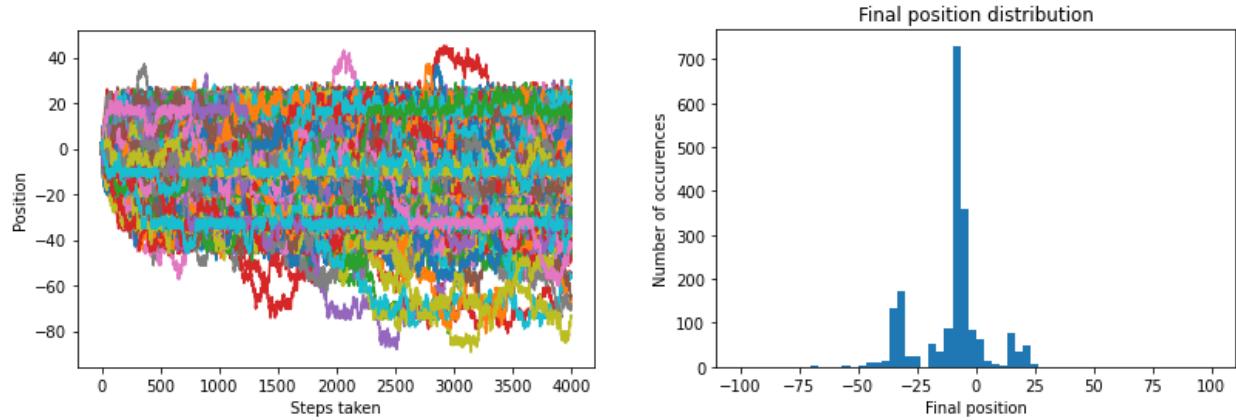


Figure 8: Anisotropic RW with 4000 steps and 5000 runs (with on the left all of the path explored and on the right the final position distribution)

When comparing the positions observed in our simulation to those of the isotropic random walk, a distinct trend emerges. We observe that the positions are notably more concentrated around the initial position. This deviation can be attributed to the unique characteristics of our chosen probability distribution. For example, we can have certain positions within the system that would exhibit

a disproportionately high likelihood of moving backward by one step.

Consequently, these positions effectively act as barriers, hindering the progression of many paths beyond their location. Moreover, upon closer examination of the final position distribution, we note a significant departure from the Gaussian distribution observed in our initial analysis for the same reasons mentioned earlier.

Furthermore, it has been given to us that the variance should be a function of  $\log^4(t)$ .

That's where we had a problem, the first few times we plotted it, we didn't have the right function at all:

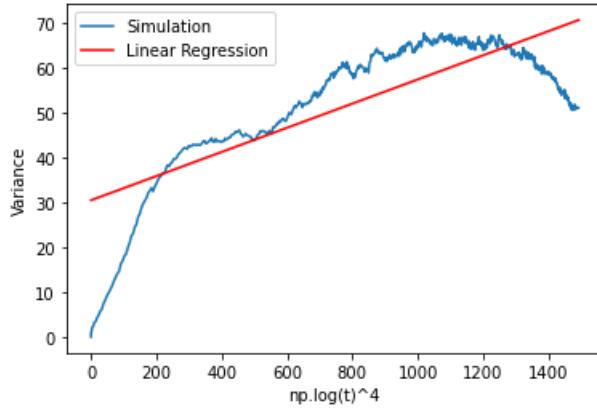


Figure 9: Variance of an ARW with the following conditions : 1000 RW and 500 steps

But, after a few tries, we somehow got some perfect runs like the following one:

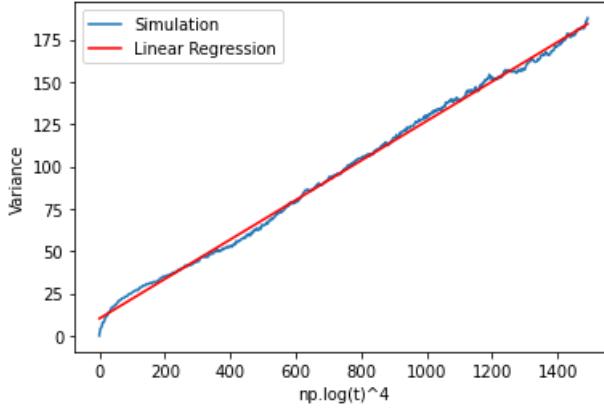


Figure 10: Variance of an ARW with the following conditions : 1000 RW and 500 steps

So, what was the issue? We suspected that the problem stemmed from the randomness of certain paths:

For example, if at a given position there was a very high probability of moving forward, but the next position had a very high probability of taking a step back, it would effectively block a path between those two positions. In another scenario, if there was a consistently high probability of moving forward, but at a particular position, the opposite was true, then the path would be unable to proceed further.

If we encounter such scenarios, the variance may exhibit anomalies due to these extreme cases. This led us to consider two solutions: either finding a probability distribution that better suits our needs, thereby abandoning the box distribution, or prolonging the simulation and increasing the number of runs to statistically eliminate these anomalies. Additionally, we realized that this problem could also be viewed as random diffusion in a physical context, which provides a more intuitive understanding.

Indeed, here, the variance will only exhibit behavior as  $\log^4(t)$  when it stabilizes. Therefore, to achieve this, we need to apply our linear regression not from the beginning, but after a certain number of steps. That's why we decided to do the linear regression in function of  $\log^4(t)$  but this time starting from a certain point in time. We chose approximately 500 steps but we also decided to do more runs and longer ones.

In doing so, we managed to repeatedly identify the expected curve:

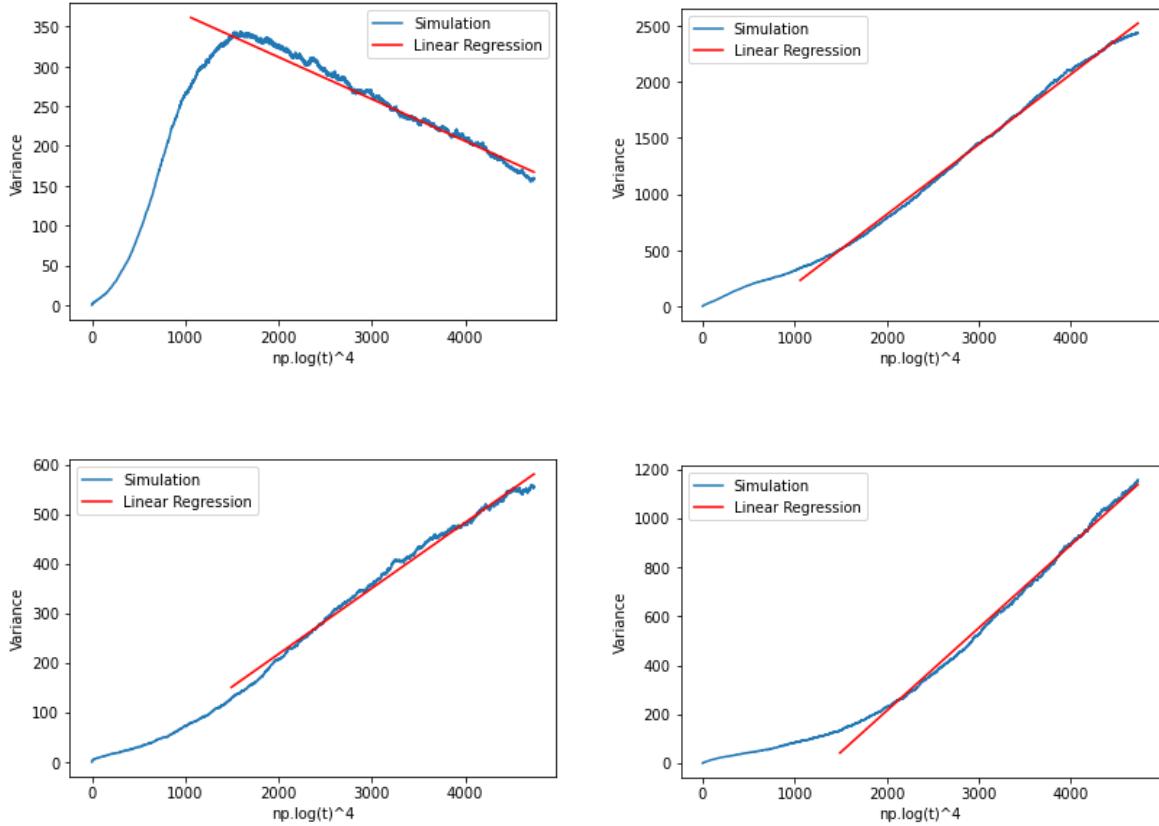


Figure 11: Variance of 4 different ARW as a function of  $\log^4(t)$  with the following conditions: 5000 RW and 5000 steps

Those simulations took approximately 5 minutes each due to the high number of steps and runs. Reducing the number of steps should keep the model unaffected, but the curve could be less smooth in some runs, making it harder to fit to a function.

This decision reduced the time taken for the simulation to a mere few seconds, and we still managed to obtain very satisfying results, as the following figures can demonstrate.

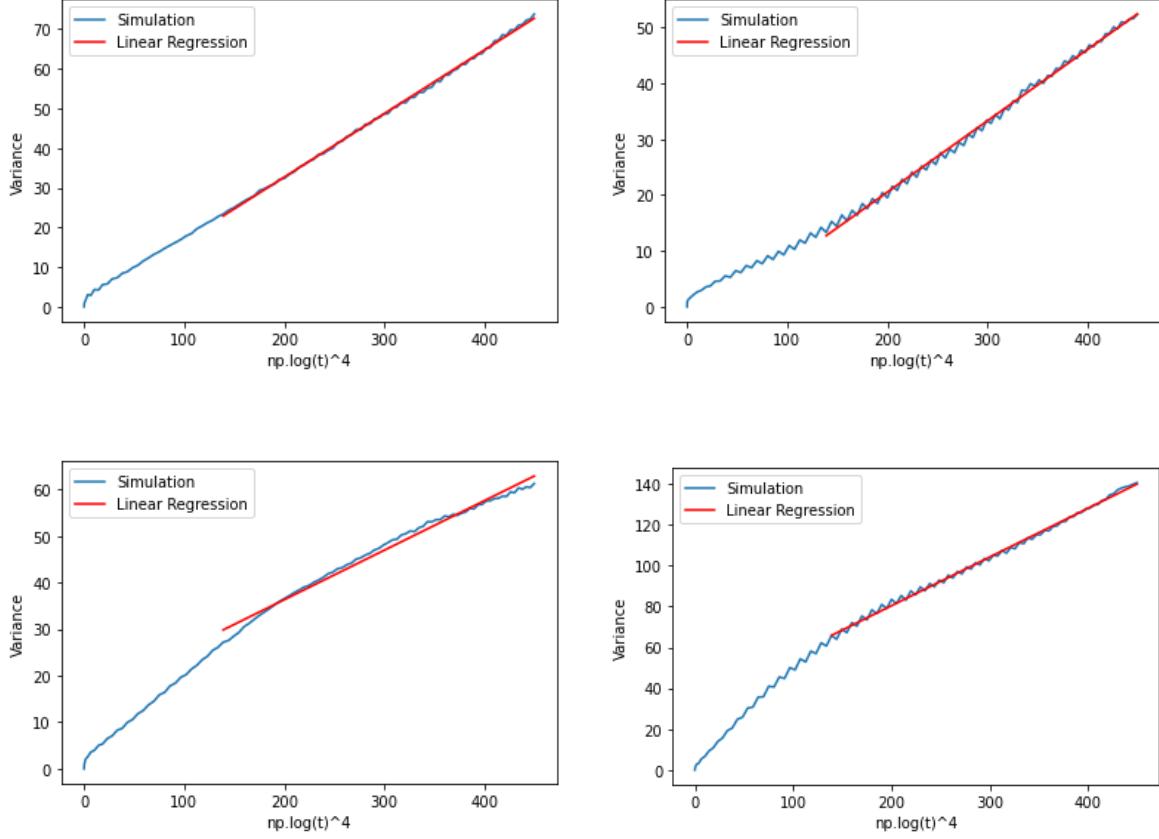


Figure 12: Variance of 4 different ARW as a function of  $\log^4(t)$  with less resources consumed: 5000 RW and 100 steps

Let's still acknowledge the fact that all of this depends on randomness. It's normal to encounter an occasional unusual simulation that may require far more steps to stabilize or far more runs to obtain the expected curve.

Additionally, we managed to find a sweet spot between the resources consumed by the code (in this case, the time taken to run it) and obtaining satisfying results. For this purpose, we required at least a few thousand random walks (in this case, 5000) of just a few hundred steps (here, 100). Finally, we need to start fitting the curve at approximately 30% of the data.

To conclude this section, it becomes evident that analyzing disordered systems necessitates the development of optimized codes capable of executing a large number of operations efficiently, especially regarding the next part of our project on disordered systems: the Ising Model using the Monte Carlo methods.

## 3 Monte Carlo Methods

Monte Carlo (MC) methods/simulations, are a class of computational algorithm that relies on random events.

### 3.1 Overview

MC methods are wildly used in sciences and especially here in simulation for uncertainty processes. Principle of MC methods vary but follow the same kind of dynamic :

- You choose a domain of inputs.
- A probability distribution will generate these inputs.
- Use these random inputs to deterministically compute a certain value.

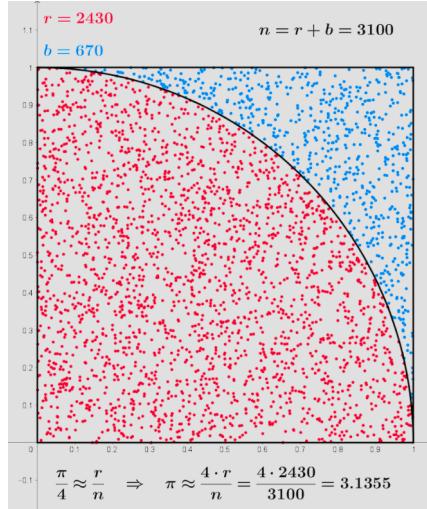


Figure 13: An example of MC simulation to compute the value of  $\pi$ . Points are added randomly on the surface, and by counting the number of point inside the quarter of the sphere and in total you get an approximation of  $\pi$ .

The core idea behind Monte Carlo (MC) methods is to leverage numerous random simulations to explore the potential outcomes of complex processes or systems. By aggregating these outcomes, Monte Carlo methods can approximate solutions to problems that are challenging or even impossible to solve analytically.

In the realm of physics, MC methods find widespread application in statistical physics and quantum mechanics. They are particularly valuable in scenarios where deterministic solutions are impractical due to the sheer number of interacting particles or degrees of freedom within a complex system. For instance, in statistical mechanics, MC simulations play a crucial role in elucidating the properties of gases, liquids, and solids across different temperatures and pressures. By simulating the random movements and interactions of particles, these simulations provide valuable insights into the behavior of such systems.

### 3.2 The Metropolis Algorithm

The Metropolis Algorithm holds a special place in computational physics due to its efficiency in sampling from complex probability distributions. It has played a pivotal role in studying phase transitions, including the melting of solids, the magnetization of materials, and the superconducting transition. By occasionally accepting worse states and thus escaping local minima, the algorithm can explore the energy landscape of physical systems more comprehensively than deterministic algorithms.

Let's dig into the process of the Metropolis Algorithm :

- Initialization: Choose an arbitrary point  $x_t$  to be the observation point in this scenario<sup>1</sup>.
- Generate  $x'$ : From  $x_t$ , we generate a new point  $x'$  from a distribution  $g(x'|x)$ .
- Calculate an acceptance ratio  $\alpha$ .
- Generate a random number  $i \in [0, 1]$ .
  - If  $i \leq \alpha$ , we accept the new candidate as  $x_{t+1} = x'$ .
  - If  $i > \alpha$ , reject the new candidate and  $x_{t+1} = x_t$ .

Now that we understood Monte Carlo methods and the Metropolis Algorithm, let us build an Ising Model.

### 3.3 The Ising Model

The Ising Model (IM) is a statistical physic model to model interacting two-states particles. In the case of a ferromagnetic model, the IM can simulate the phase transition of the material around a certain temperature, the Curie temperature.

We choose to form an  $N \times M$  matrix that can be seen as a grid. Each case  $i$  of this grid represent a spin  $\sigma_i$  that can be up ( $\uparrow, \sigma_i = 1$ ) or down ( $\downarrow, \sigma_i = -1$ ).

The magnetization  $m$  and energy  $\epsilon$  of each particle is then simply

$$m_i = \mu\sigma_i \tag{1}$$

$$\epsilon_i = -J \sum_{j \in V} \sigma_i \sigma_j \tag{2}$$

with  $V$  the neighbour of the  $i$ -particle set.

To get the total energy and magnetization, we then sum on every particle in our system, getting :

$$M = \sum_i m_i = \mu \sum_i \sigma_i \tag{3}$$

$$E = \sum_i \epsilon_i = -J \sum_i \sum_{j \in V} \sigma_i \sigma_j \tag{4}$$

---

<sup>1</sup>It is possible to have a density probability to choose  $x_t$  depending on the previous point chosen, but here we will stay with the simple random point from the whole system.

### 3.3.1 Sample to sample fluctuation

Let us consider the following scenario: we have a magnet with ferromagnetic alignment, and it is initially at a temperature of  $T = 0K$ . At time  $t_0$ , we immerse this magnet in a hot bath. The introduction of thermal energy from the bath will induce a change in the spin orientations. How will this affect the variation in energy?

This problem can be addressed using the Ising model and the Metropolis algorithm. Assuming we begin with all spins aligned upwards, the initial energy of the system is minimized, given by  $\epsilon_i = -NJ$ , where  $N$  represents the number of spins.

When the magnet is dived in the bath, the increase of the temperature will give some energy to the system to flip a spin. Let  $x$  be the system before the flip of one spin,  $x'$  the system after the flip<sup>2</sup>, and  $\Delta E = E(x') - E(x)$  our acceptance ratio, which is the variation of the energy, and  $i$  the random number.

Application of the Metropolis algorithm: if  $\Delta E < 0$ , we reject the flip, if  $\Delta E > 0$ , we calculate  $e^{\beta\Delta E}$ : if  $e^{\beta\Delta E} < A$ , we reject the flip and if  $e^{\beta\Delta E} > A$ , we accept the flip. On repeating the operation for all the spins, we can see the evolution of the global energy.

### 3.3.2 Results

We start from a random lattice (Figure 14) and then compute the Metropolis Algorithm.

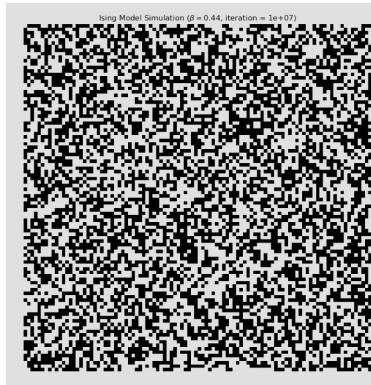


Figure 14: The starting lattice of the IM (random values for  $\sigma_i$ ). This is a  $100 \times 100$  grid.

We can detect different behaviour of the spins depending on the  $\beta$  value and the time we are looking at the system. But whatever these parameters, the system tends to create “clusters” with the same spin orientations (see Figures 15 and 16). These clusters can grow, shrink or even move through the system, and it seems like their behaviour depends mostly on the temperature.

The goal is now to do the same simulation for different temperature values. As every simulation is independent, we can use multiprocessing to do each simulation. We then used *multiprocessing* from the *Pool* Python library to lower the computation time<sup>3</sup>.

---

<sup>2</sup>The  $g(x'|x)$  distribution is a uniform probability density, meaning that whatever previous point we chose on the last iteration, all points can be choose now.

<sup>3</sup>The multiprocessing will separate each simulation to different core of the CPU to make them run at the same time, lowering the computation time.

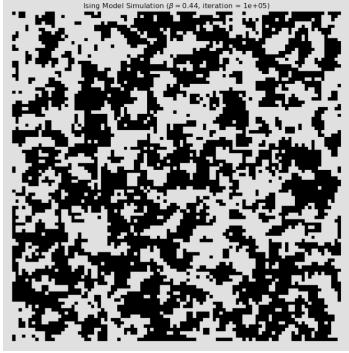


Figure 15: Start of cluster formation.

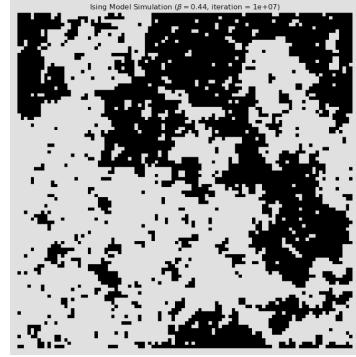


Figure 16: More advanced clusters with a strong differentiation.

We can now measure the average energy  $\langle E \rangle$  and magnetisation  $\langle M \rangle$  at the end of every simulation (here we took 100 values in the 10'000 lasts iterations, with around  $10^6$  total iterations) for different values of temperature  $T$ . We directly see a phase transition, and we calculate the derivative of the magnetization<sup>4</sup> to get the critical temperature  $T_c$ <sup>5</sup> (see Figure 17). The computed critical temperature known, we can then plot the magnetization graph as in Figure 18.

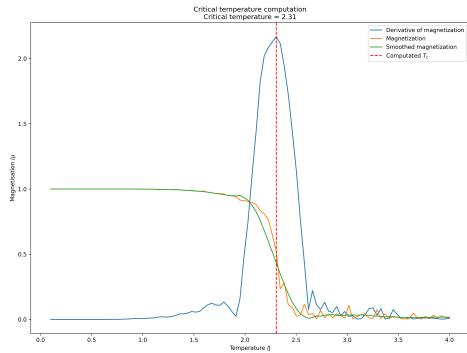


Figure 17: The computed critical temperature of our system. We can see the magnetisation (orange line) and the derivative of the magnetization (blue line) that gives the critical temperature.

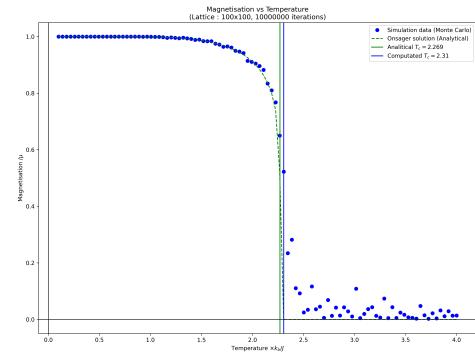


Figure 18: The phase transition with the critical temperature (analytically and computed).

As we can see in the previous figures, our simulation closely approximates the analytical  $T_c$  value.

A key aspect of the approximation stems from the discretization of space into a lattice. While increasing the lattice size can mitigate this effect, it also leads to longer iteration times<sup>6</sup>. An interesting avenue for further investigation would be to explore the relationship between the critical temperature approximation and the size of the lattice.

<sup>4</sup>The derivation is done by smoothing out the magnetization curve with a Savgol Filter for better results.

<sup>5</sup>Another way to get  $T_c$  would have been to get the maximum susceptibility, but in our simulation, the susceptibility was way too fluctuating for any use.

<sup>6</sup>If we imagine a  $10 \times 10$  lattice, flipping a spin will change 1% of the lattice, while flipping a spin in a  $100 \times 100$  will only change 0.001%. We then need more iteration to approach the same “real time”.

## 3.4 Different interaction constants $J$

### 3.4.1 Constant J for the entire system

For this next part, we first decided to change the constant  $J$  and see what would happen if we changed it beforehand. We gave it multiple different values ranging from 0 to 2 and plotted the energy of the system and magnetisation depending as functions of temperature.

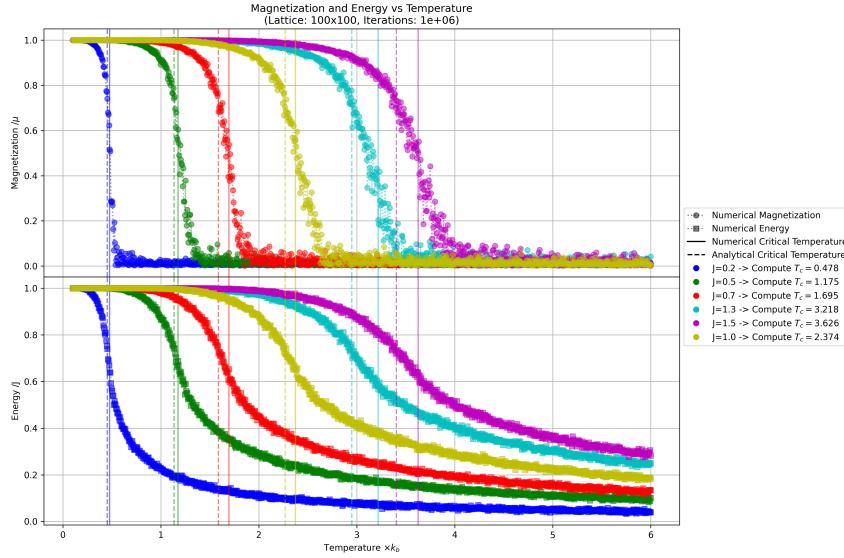


Figure 19: Energy and magnetization graphs for different values of  $J$ .

We immediately noticed a trend between the chosen value of  $J$  and the critical temperature obtained with the simulations. After plotting it it becomes clear that it's a linear function : That's expected,

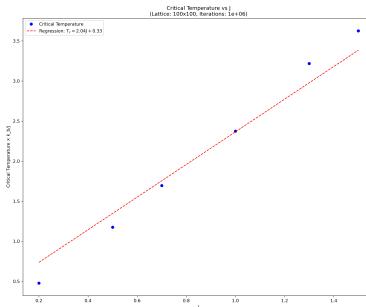


Figure 20: Linear Regression on the critical temperature depending on  $J$  values.

as the critical temperature and energy depend linearly on  $J$ . Therefore, changing  $J$  in the system will result in proportional changes in both the energy and the critical temperature.

The next step is more complex: we have decided to vary  $J$  for each bond. This means that for every pair of particles,  $J_{i,j}$  will be entirely random (selected from a specific range).

### 3.4.2 Different J for each pair of spins

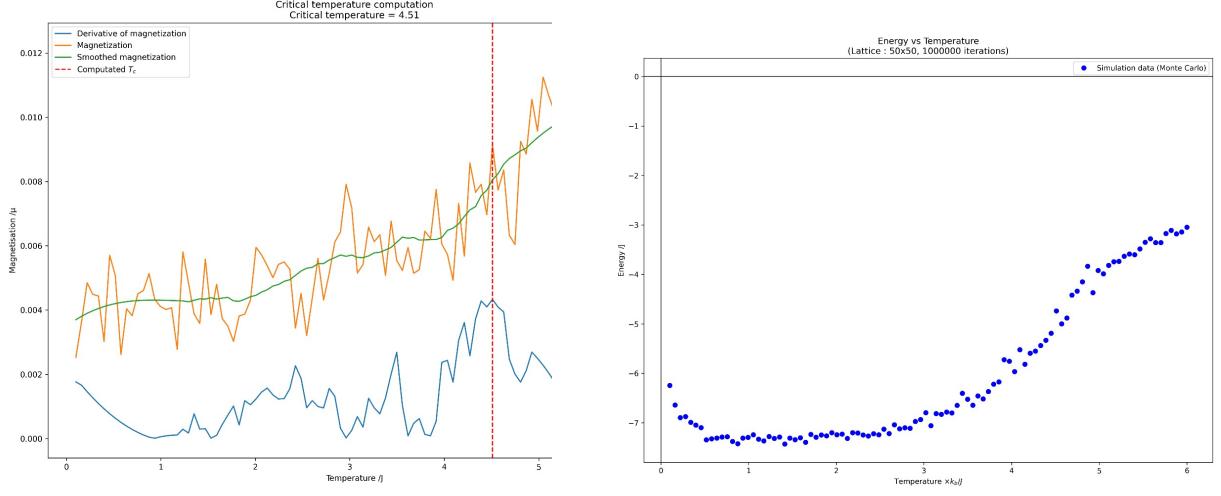


Figure 21: Magnetization (on the left) and Energy (on the right) both represented as functions of Temperature with  $J$  being different for each pair of spins

Based on the observations from the previous figures, we can suggest the following hypothesis: We posit that disorder alters the critical temperature's value. This alteration likely occurs due to the varied interactions resulting from the random coefficients of interaction  $J$  between spins. In contrast to the classical case, where magnetization diminishes at low temperatures due to spin alignment, here we observe that magnetization remains close to 0 regardless of temperature. Even though there is a slight increase in magnetization at higher temperatures, the value still hovers around 0.01, whereas previously we observed a decrease in magnetization with increasing temperature. Additionally, the plots reveal slower phase transitions compared to the classical scenario.

## 4 Conclusion

To summarize, our exploration of random walks, both isotropic and anisotropic, alongside the Ising model, has yielded valuable insights into the behavior of random systems across various parameter changes. Through numerical simulations, we have observed various patterns and dynamics emerging from these models.

In addition to our numerical simulations, we enriched our study with analytical analysis, providing a more comprehensive perspective on random walks and the Ising model. By combining numerical and analytical techniques, we gained deeper insights into the underlying principles governing these systems. Our analytical analysis enabled us to derive theoretical insights and mathematical expressions that complemented our numerical findings.

Towards the end of our study, we encountered some observations for which we did not have a complete understanding. Regrettably, we were unable to thoroughly investigate these findings or verify our hypotheses through analytical study due to time constraints. With additional time, we believe we could have gained a better understanding and provided more insightful commentary on these observations.

Furthermore, our analytical results served as valuable benchmarks for validating the accuracy of our numerical simulations. By comparing the predictions of our analytical models with the outcomes of our numerical experiments, we verified the robustness of our computational methods and identified areas for improvement.

Moving forward, we can leverage our analytical findings to refine and extend our numerical simulations. Insights gained from analytical solutions can guide the selection of appropriate parameter ranges or initial conditions, enhancing the efficiency and accuracy of our simulations.

One notable takeaway from our study is the significant computational resources and time required for such simulations. This underscores the importance of efficient algorithms and computational techniques in tackling complex systems like random walks and the Ising model. Exploring optimization strategies, such as parallel computing techniques and leveraging specialized hardware like GPUs with CUDA cores, could expedite our simulations.

Furthermore, our project paves the way for further investigation into various aspects of random systems. Exploring the effects of different parameter variations, incorporating real-world data, extending our analysis to higher dimensions or non-equilibrium scenarios, and investigating the impact of external influences hold promise for yielding intriguing results.

In conclusion, while our project has laid a solid foundation for understanding random systems through numerical simulations, there is still much to explore and refine. By addressing the challenges of computational complexity and venturing into new avenues of research, we can deepen our understanding of the fascinating dynamics of disorganized and random systems.

## References

<https://arxiv.org/pdf/0803.0217.pdf>

[http://micro.stanford.edu/~caiwei/me334/ME346A\\_LectureNotesWin2012.pdf](http://micro.stanford.edu/~caiwei/me334/ME346A_LectureNotesWin2012.pdf)