

2D random walk

CONTENTS

[Summary](#)[Learning objectives](#)[Some Physics regarding the 2D random walk](#)[Exercise 1: 2D random walk](#)[Exercise 2: how far does the particle get?](#)[Acknowledgements/References](#)

Summary

This challenge focuses on implementing a 2D random walk using a discrete grid system. Inspired by a concept from the Feynman Lectures on Physics, the exercise simulates the movement of a particle that is "kicked" to a new position on a 2D grid at each time step. The model uses a von Neumann neighbourhood, meaning the particle can move to one of the four adjacent cells ('up', 'right', 'down', or 'left') or ('north', 'east', 'south', 'west') in each step. The simulation incorporates periodic boundary conditions, allowing the particle to wrap around the edges of the grid, creating the effect of an infinite space. You will then measure how far the particle gets from where it began, averaged over many walks.

LEARNING OBJECTIVES

- Understand 2D Random Walks:
 - Gain understanding of random walks in two-dimensional space and how they differ from 1D random walks.
- von Neumann Neighbourhood:
 - Learn about the von Neumann neighbourhood in grid-based models, where a particle can move to one of four adjacent positions.
- Periodic Boundary Conditions in 2D:
 - Implement periodic boundary conditions in a 2D grid, understanding how they allow for a seamless, infinite space by wrapping around edges.
- Visual Representation of Particle Movement:

- Create visual outputs that represent the path of a particle in a 2D grid, using symbols and possibly colours to enhance the representation.
- Apply Random Walk Concepts to Physical Phenomena:
 - Understand how the random walk model can be used to represent real-world phenomena, such as the movement of particles undergoing Brownian motion.
- Measure a Random Walk:
 - Measure how far the particle travels from its origin and compare the result, averaged over many walks, with the number of steps taken.

By the end of this challenge, you should have a practical understanding of 2D random walks, the ability to implement and visualize them, and an appreciation for how these models can represent complex physical behaviours.

Some Physics regarding the 2D random walk (continuous model)

Look at Figure 1, taken from the *Feynman Lectures on Physics* (see acknowledgements).



Figure 1: A 2D random walk of 36 steps of length 1.

The figure represents the path that a randomly moving particle might take inside a 2D space. This particular ‘model’, as you can see from reading Feynman’s 41st lecture, of volume 1 in his classic 3-volume series, assumes a continuous range of angles of collision (“bombarded on all sides”), that kick the particle around. As Feynman says in that lecture:

“Let us consider how the position of a jiggling particle should change with time, for very long times compared with the time between “kicks.” Consider a little Brownian movement particle which is jiggling about because it is bombarded on all sides by irregularly jiggling water molecules. Query: After a given length of time, how far away is it likely to be from where it began?”

In this challenge, we are not going to implement a continuous model. It will be a discrete **2D space, a grid...** again, not unlike the line of cells in the 1D random walk (or the Nagel and Rasmussen car model, if you have met it), but a 2D grid for a wandering particle. Let's rephrase the problem: consider a particle in a 2D grid, centred at an origin (the middle of the grid). At each time step, the particle is 'kicked' to a new position of length $l = 1$. As we are in a grid, and as the length can only be 1, we can picture the possible moves the particle can take from its current position - i, j :

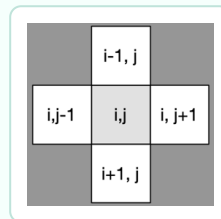


Figure 2: von Neumann 'neighbourhood'.

As time unfolds, the 'particle' (use a 'graphic' for that), output on a square grid of size L , will be seen to drift as the following example shows:

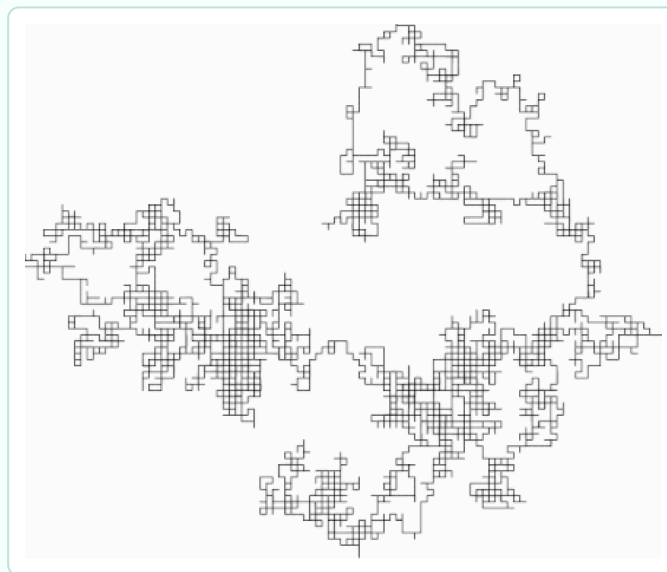


Figure 3: Accumulative result of a random walk constrained to a von Neumann 'hop' at time t .

Figure 3 was drawn with lines, from a much longer walk than you need here. Your console output will be blockier, and a short walk is enough to see the particle drift.

Exercise 1: 2D random walk

Create a script called `random_walk_2D.py`. Implement a 2D random walk, with von Neumann neighbourhood constraints, and output your results to the console. As a starting point, use a grid of size $L = 30$, start the particle in the middle of the grid and take 100 steps, marking every cell that the particle visits. Use whatever symbols you want to represent the particle and the empty space, and use colours, if you like (colour needs a third-party package, such as `colorama`, which is installed with `pip`). Implement it however you like, but ensure that you implement periodic boundaries – i.e., a boundary condition that simulates the lattice as if it extends infinitely in all directions. Instead of treating the lattice as a finite space with edges, a periodic boundary condition enforces the idea that the lattice "wraps around" itself. In other words, if you move beyond one edge of the lattice, you re-enter from the opposite edge.

Exercise 2: how far does the particle get?

Return to Feynman's question: after a given number of steps, how far away is the particle likely to be from where it began? Alongside the wrapped position that you draw, keep track of the particle's *real* displacement from the origin, in both directions (as you did for the real distance in the 1D random walk). At the end of a walk of N steps, calculate the square of the distance from the origin. Repeat this for many walks (1000 is a good number to start with, and there is no need to draw them all) and calculate the average of the squared distances. How does the average compare with N ? What happens as you increase the number of walks?

You should find that the average is close to N : about 100 for walks of 100 steps. Do not expect a single walk to be close to N , though: its squared distance can be anywhere from 0 to several times N , so the average only settles down over many walks. With 1000 walks of 100 steps, expect an average within about 10 of 100.

Acknowledgements/References

- 1) Figure 1: https://www.feynmanlectures.caltech.edu/I_41.html
- 2) Feynman Lectures on Physics: <https://www.feynmanlectures.caltech.edu/>
- 3) Landau, R.H, Paez, M. J., Bordeianu, C.C (2007) *Computational Physics: Problem Solving with Computers*. WILEY-VCH, Weinheim. See chapter 11.1 for random walk description from a computational physics standpoint.