Probability Final Project Report

Name: Ma'aaz Arshad Shaheen ID: ms05176
Section: L3

Github Link: https://github.com/Maaaz-Shaheen/Probability-Final-Project-Random-Walker

App link: https://prob-project-ms05176.netlify.app/

Task1:

Our model for this task was used to calculate the expected distance from the starting point after a number of steps. The default values for our model are 10000, each of 100 steps, with step size being a discrete random variable with values -1 and 1, each with equal probabilities. As the number of simulations increased, the expected value of the mean got closer to 0, as can be seen in the histogram Fig.1. below. If the probabilities are changed to favour one direction, the histogram shifts in the particular direction(Fig 2). The correctness of my implementation can be verified by looking at the absolute distance from the initial point. For the absolute distance with equal probabilities, we can use the formula in Fig 3 and we get the value 7.97… mathematically. If we calculate the expected absolute distance, then as we increase the number of simulations, we get closer to 7.9.

Task2:

One assumption made in this task is that the dots meet not when they are at the same point only, but when they have crossed each other too. The probabilities and step sizes are the same as task 1. The distance between the points is assumed to be 5 here. If we increase the absolute distance between them, the time for them to meet increases, however, except for a few outliers, they generally meet around the region. If the probabilities are increased in one direction, it does not seem to have an effect on the meeting times. But shifts the meeting place in the direction of the greater probability. The number of simulations is also kept at a 1000 here.

Task 3:

The particle was placed in a Cartesian plane at origin. It's step size is discrete from 0, 0.5 and 1. The orientation was to be taken as discrete from [0 – 2pi] so the values 0, (½)pi, pi, and (3/2)pi were assumed, each with an equal probability. The motion of the particle was decide using vectors, with a random value being chosen from the step size and the orientation and then converted to Cartesian values, with each value being added to the particles previous x and y position. The model of re-entry was decided to be in a manner similar to the Snake game, where exiting from the circular region at x,y would place you at the corresponding -x,-y region. Here the circular region was assumed to be spherical in thought, which means a particle existing the circle is not actually exiting but is moving on a spherical object, where the next point it is transferred to is just represented on the opposite end of the circle. Keeping in line with this analogy, the correct distance the point the particle needed to be transferred to must also keep in mind the distance it has travelled already. Which has been factored for in the code.

Task 4:

Not much was changed except the step size was chosen from a continuous range from [0 - 1]. This is verified in a similar degree as task 1 also.

Task 5/6:

Not much was changed except the step size was chosen from a continuous range from [0 - 1] and angle was chosen from a range of [0 – 2pi). 2pi is not included as the orientation of 2pi and 0 are the same, leading to an unfair bias in that direction, removing uniformity.

Task 7:

Not much was changed except the step size was chosen from a discrete range from [0 - 1] i.e -1, 0, and 1, while orientation was kept continuous.

Task 8:

The majority of the details were carried over from task 5. However, the initial position of the nodes was carried out using the formulas in fig 4. This is due to the fact that we wished to assume a truly uniform randomness for the initial point of the two nodes. Just simply choosing a random value from [0, 1] and from [0, 2pi] would result in more values closer to the center, which is why these formulas were used. As this simulation could run for very long per iteration, an upper limit of 100,000 steps was bound. And any simulation that did not have the nodes meet within this time was discarded. And the remaining expected value is calculated from the results that bore fruit. I could not find a way to verify this. The paper that I did find (reference 3 below) was extremely dense and required topics not studied yet. However, I believe the simulation does run correctly and would yield the correct result.

Figures:

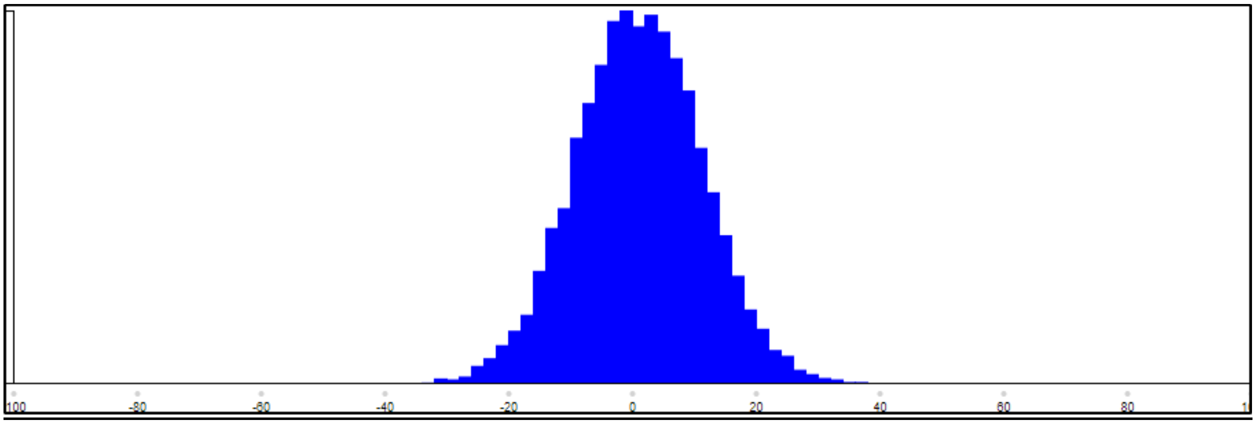

Fig. 1

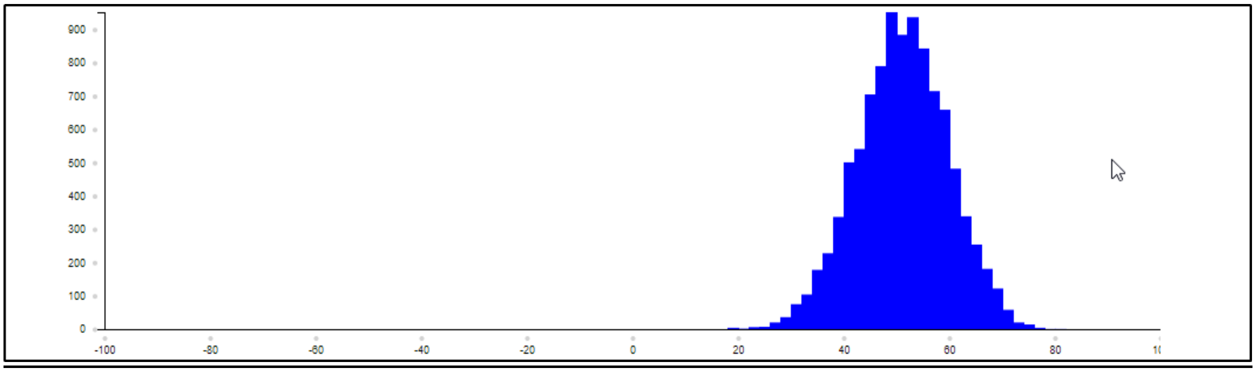

Fig. 2

$$\langle d_N \rangle \sim \sqrt{\frac{2N}{\pi}},$$

Fig. 3

More visualisations can be found on the website. For example, line graphs showing paths and random walker.

References:

1. https://mathworld.wolfram.com/RandomWalk1-Dimensional.html
2. https://stackoverflow.com/a/50746409
3. https://www.researchgate.net/figure/Encounter-probability-of-two-random-walkers-initially-placed-at-x-1-x-2-where-x-1_fig3_231110695