1 Problem Description

The objective of the HW4 is to solve the problem of random numbers, random walks, 1D diffusion and mixing of the gasses. The report will focus on the random numbers and various implementation of those. Detailed explanation for each section can be found in the later section of the report.

2 Solution to Random Numbers

Pseudo-Code:

1. *Generate uniform random numbers between 0 to 1 using random or numpy modules 1000 random number and 1000000 random number*
2. *To better visualize the distribution use histogram as we can change the interval size or 'bin' size*
3. *Convert uniformly distributed random numbers into gaussian distributed ones using the Box-Muller transformation.*
4. *for box-muller, split the data into 2 equal parts 'u1' and 'u2', Box-Muller formula to transform u1 and u2 into two sets of normally distributed random numbers, z1 and z2 and then finally combine the z1 and z2*
5. *Repeat step 2*

2.1 Generation of random numbers with distribution increasing resolution

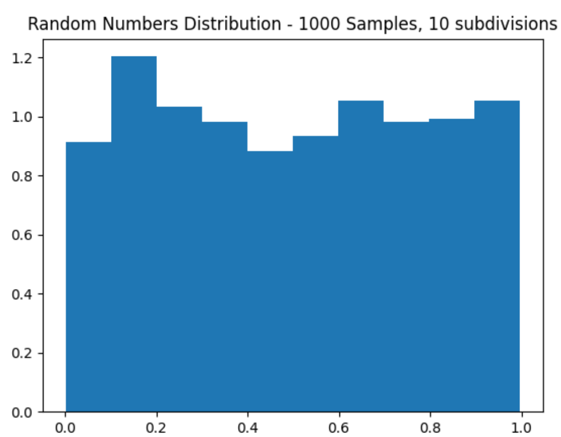

Figure 1: Random Numbers Distribution - 1000 Samples, 10 subdivisions

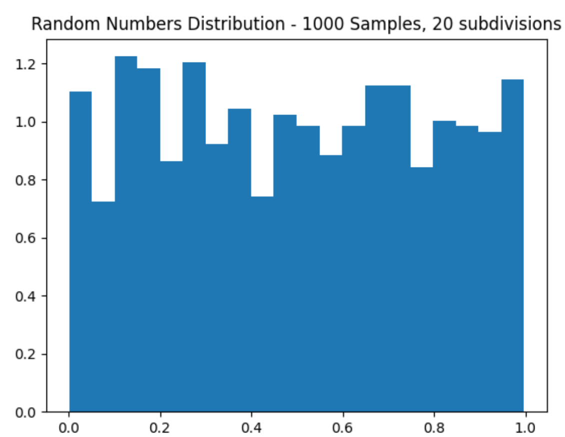

Figure 2: Random Numbers Distribution - 1000 Samples, 20 subdivisions

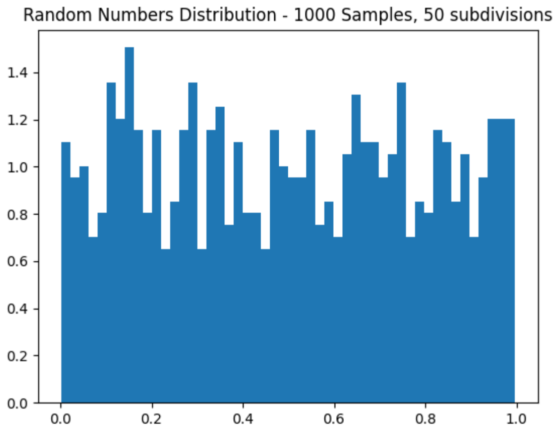

Figure 3: Random Numbers Distribution - 1000 Samples, 50 subdivision

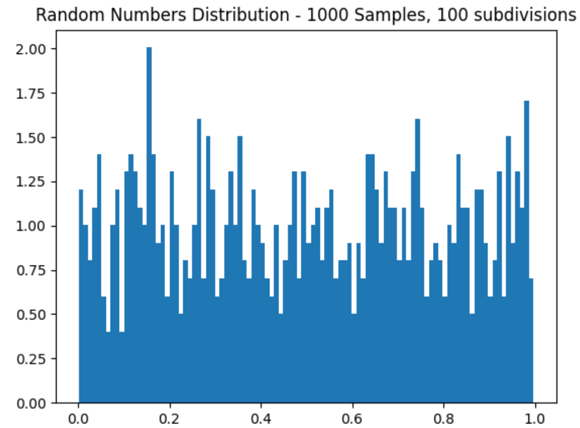

Figure 4: Random Numbers Distribution - 1000 Samples, 100 subdivisions

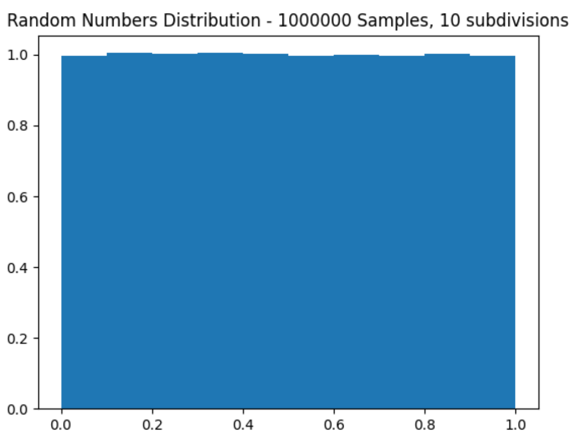

Figure 5: Random Numbers Distribution - 1000000 Samples, 10 subdivisions

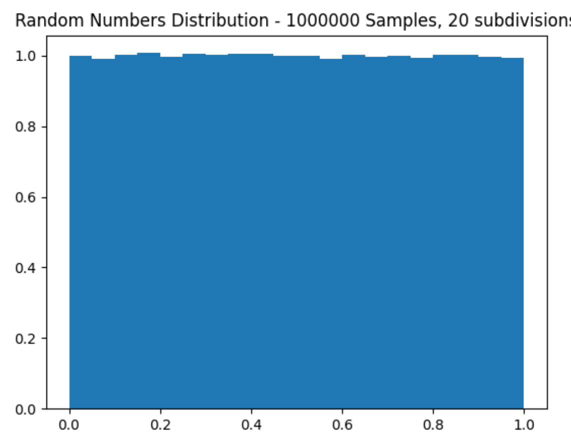

Figure 6: Random Numbers Distribution - 1000000 Samples, 20 subdivisions

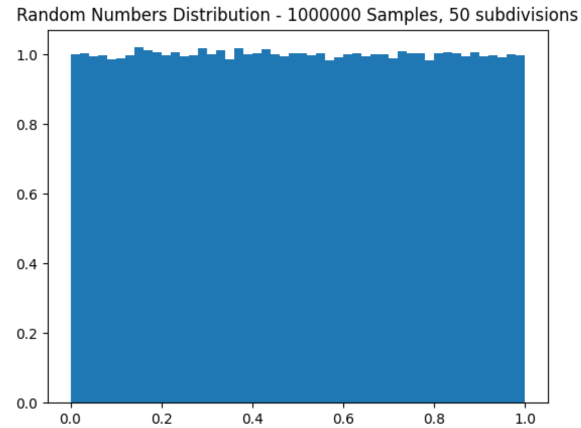

Figure 7: Random Numbers Distribution - 1000000 Samples, 50 subdivisions

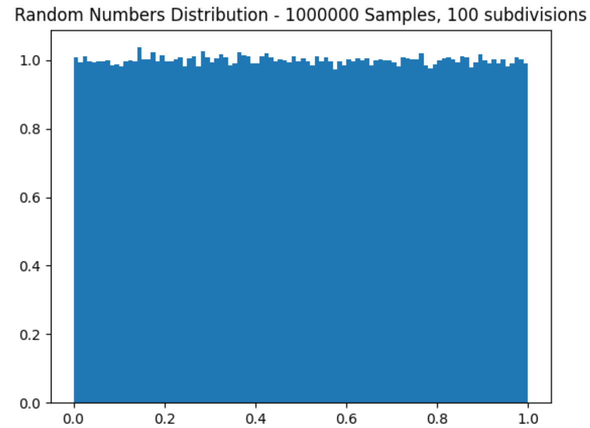

Figure 8: Random Numbers Distribution - 1000000 Samples, 100 subdivisions

2.2 Generation the Gaussian distributed random numbers

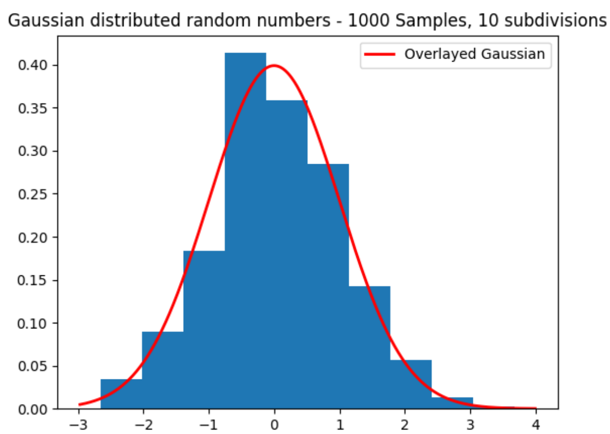

Figure 9: Gaussian distributed random numbers - 1000 Samples, 10 subdivisions

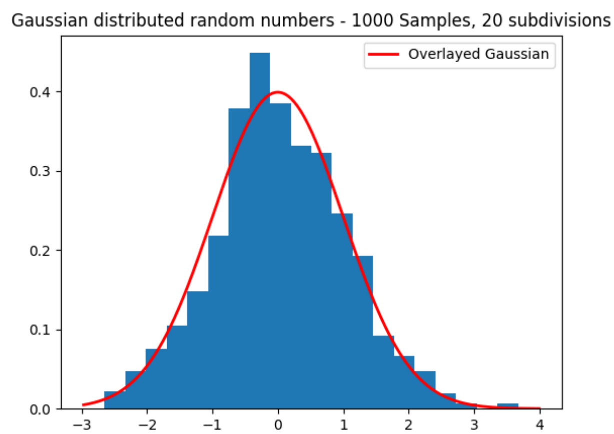

Figure 10: Gaussian distributed random numbers - 1000 Samples, 20 subdivision

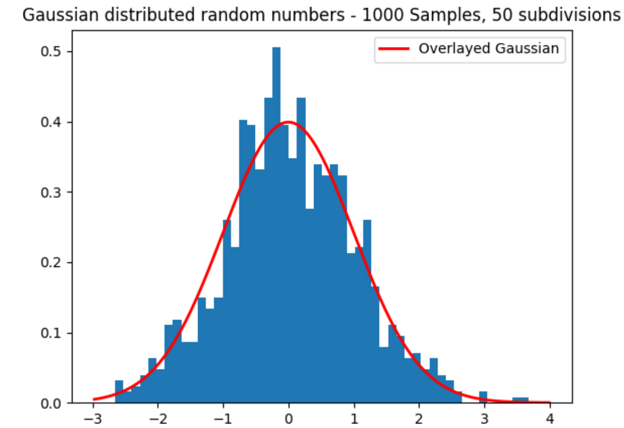

Figure 11: Gaussian distributed random numbers - 1000 Samples, 50 subdivisions

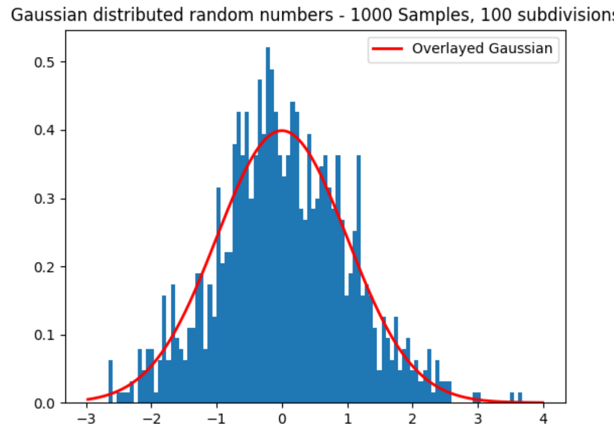

Figure 12: Gaussian distributed random numbers - 1000 Samples, 100 subdivision

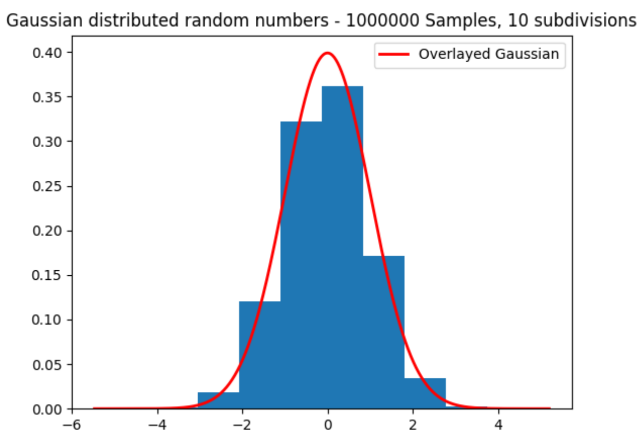

Figure 13: Gaussian distributed random numbers - 1000000 Samples, 10 subdivisions

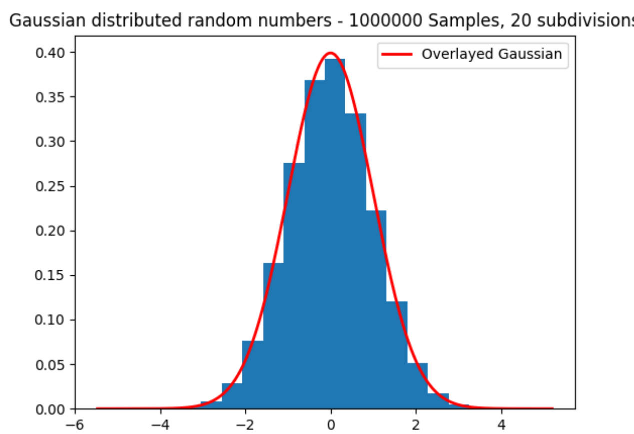

Figure 14: Gaussian distributed random numbers - 1000000 Samples, 20 subdivisions

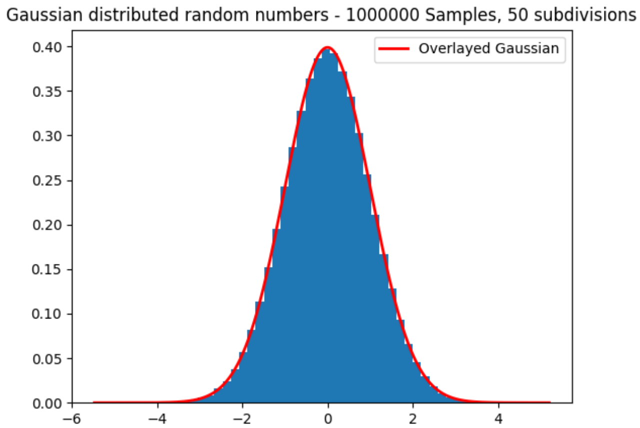

Figure 15: Gaussian distributed random numbers - 1000000 Samples, 50 subdivisions

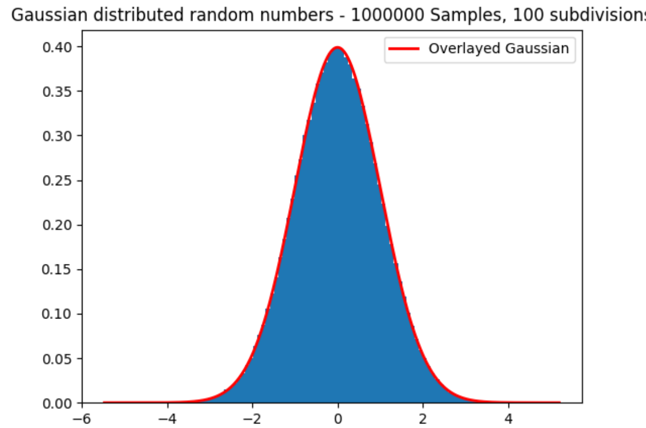

Figure 16: Gaussian distributed random numbers - 1000000 Samples, 100 subdivisions

3 2D Random Walk

Pseudo-Code:

1. *Give equal probability to move up-down-left-right in a 2D, start teh walk from (0,0)*
2. *For steps less than 3 do only one trials for find the probability of moving and then move*
3. *Use the end coordinate of the previous step as an starting point for the next step*
4. *For steps more than 3, take average of 10000 such probability*
5. *From each walk we will get set of average values for x and y coordinates*
6. *Plot x vs steps and x^2 vs steps.*
7. *For the mean square distance from the starting point, calculate the value of $x^2 + y^2$.*

3.1 Average of $< x_n >$ and $< x_n >^2$ up to $n = 100$

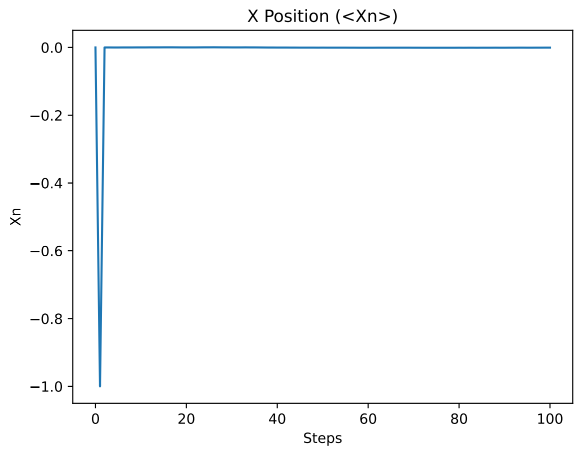

Figure 17: Average of x_n

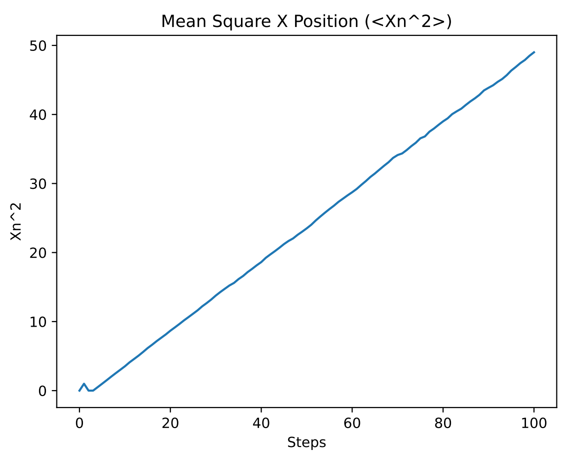

Figure 18: Square of the Average of x_n

3.2 Mean square distance from the starting point $< r^2 >$ up to $n = 100$

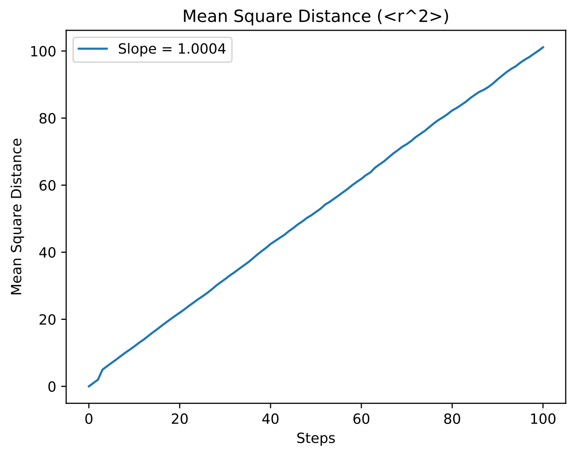

Figure 19: Mean square distance from the starting point $< r^2 >$

As shown in Figure 19, the slope of the mean squared distance is approximately 1. And as we know:

$$r^2 = 4Dt \tag{1}$$

In this equation:

- r represents the root mean square displacement
- D is the diffusion constant

We can find the value of diffusion constant by dividing the slope with 4, hence the value for diffusion constant = 0.5 (unit length)^2/steps

4 Solution to Diffusion Equation

1. *Define the diffusion function with parameters D, boundary limit, initial spread, grid spacing, time interval, total number of iterations.*
2. *Create a initial grid within boundary using grid spacing*
3. *Initialize density profile (ρ) with values 1 inside initial_spread and 0 otherwise.(like a drop of min in the centre of tea cup)*
4. *Iterate over steps:*
5. *At each time step, look at each grid point, compare it with its neighbors, and adjust its value accordingly, simulating diffusion over time.*
6. *Define a normal distribution which accepts the standard deviation given in the paper*

4.1 Analytical Derivation

Given the 1D Normal Distribution Given the probability density function for a one-dimensional normal distribution:

$$\rho(x,t) = \frac{1}{\sqrt{2\pi\sigma^2(t)}} \exp\left(-\frac{x^2}{2\sigma^2(t)}\right) \tag{2}$$

We calculate the expectation value of x^2:

$$\langle x(t)^2 \rangle = \int_{-\infty}^{\infty} x^2 \rho(x,t)\, dx \tag{3}$$

We perform a substitution:

$$u = \frac{x}{\sqrt{2}\sigma(t)}, \quad du = \frac{dx}{\sqrt{2}\sigma(t)}, \quad x^2 = 2\sigma^2(t)u^2 \tag{4}$$

The integral becomes:

$$\langle x(t)^2 \rangle = 2\sigma^3(t) \int_{-\infty}^{\infty} \frac{u^2}{\sqrt{\pi}} \exp(-u^2)\, du \tag{5}$$

We solve the integral using integration by parts. Let $v = u$ and $dw = ue^{-u^2}\, du$, then $dv = du$ and $w = -\frac{1}{2}e^{-u^2}$. The integral is:

$$\int u^2 e^{-u^2}\, du = u\left(-\frac{1}{2}e^{-u^2}\right) - \int \left(-\frac{1}{2}e^{-u^2}\right) du \tag{6}$$

The first term vanishes at the limits, and the second term integral is $\sqrt{\pi}$, so:

$$\langle x(t)^2 \rangle = 2\sigma^3(t)\frac{\sqrt{\pi}}{2} = \sigma^2(t) \tag{7}$$

This shows that $\langle x(t)^2 \rangle = \sigma^2(t)$.

4.2 Solution to diffusion equation using the finite difference form with a diffusion constant D = 2

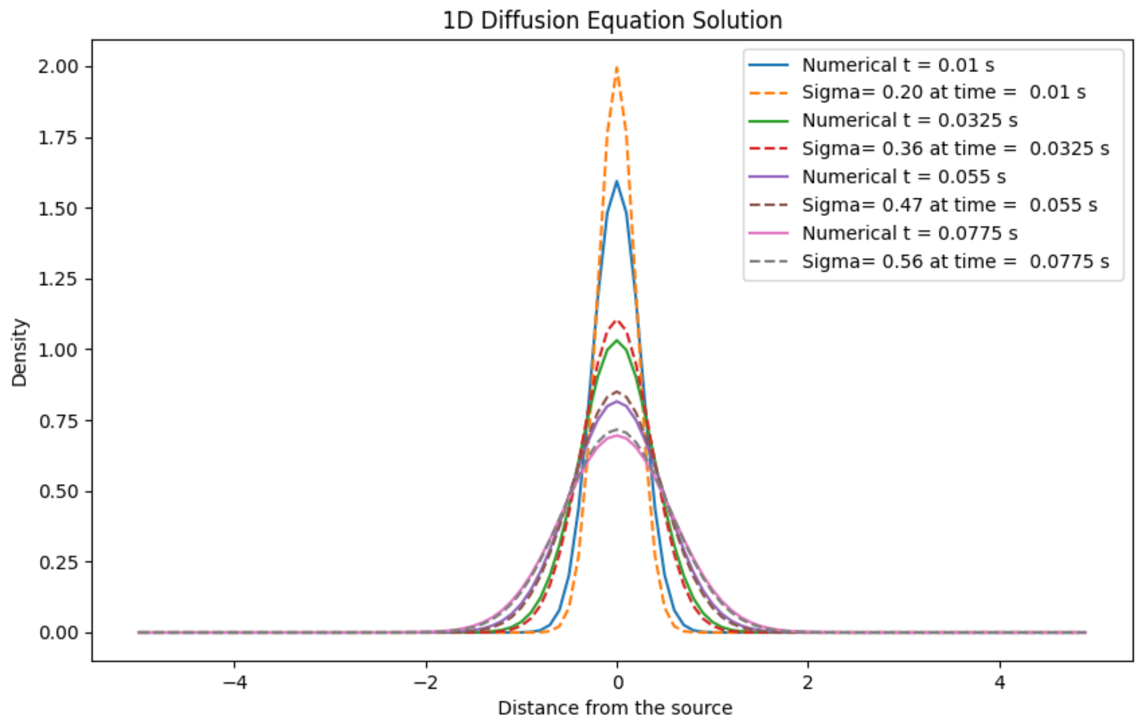

Figure 20: 1D Diffusion Equation Solution

From Figure 20, we can see that the after t = 0.055 seconds the density profile corresponds to a normal distribution.

5 Mixing of two Gases

Pseudo-Code:

1. *Define a 2D array and give gas A a vlaue of 1 and gas B value of -1 and empty as 0*
2. *store the indices of gas A,gasB and empty seperatly*
3. *randomly select gas A or gas B*
4. *once selected, randomly select the direction in which gas can move, note 4 directions have equal probability*
5. *check if that space is empty*
6. *if empty swap the value of 2D array and also the the 2 tuples*
7. *if not empty, discard and start again*

5.1 Set up a 2D grid with 60x40 grid size and run it for longer duration

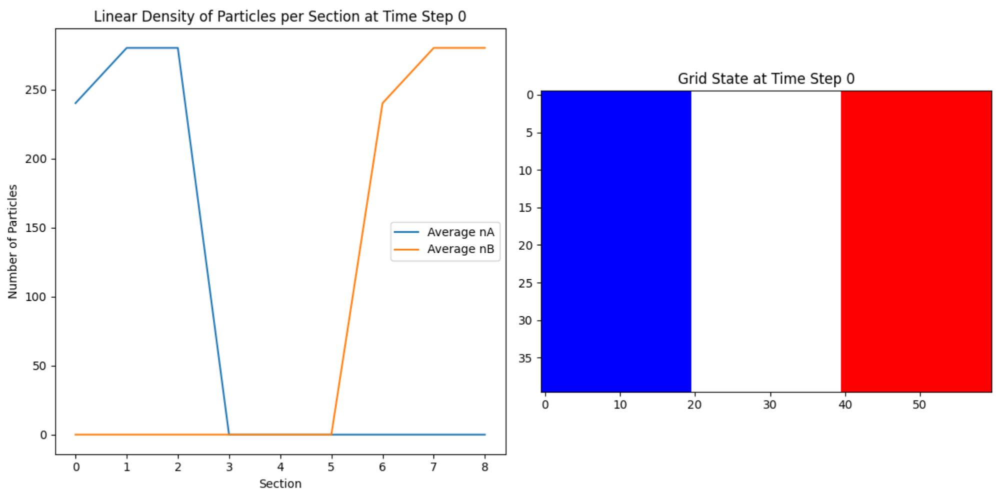

Figure 21: Linear Density and the Grid after 0 Steps

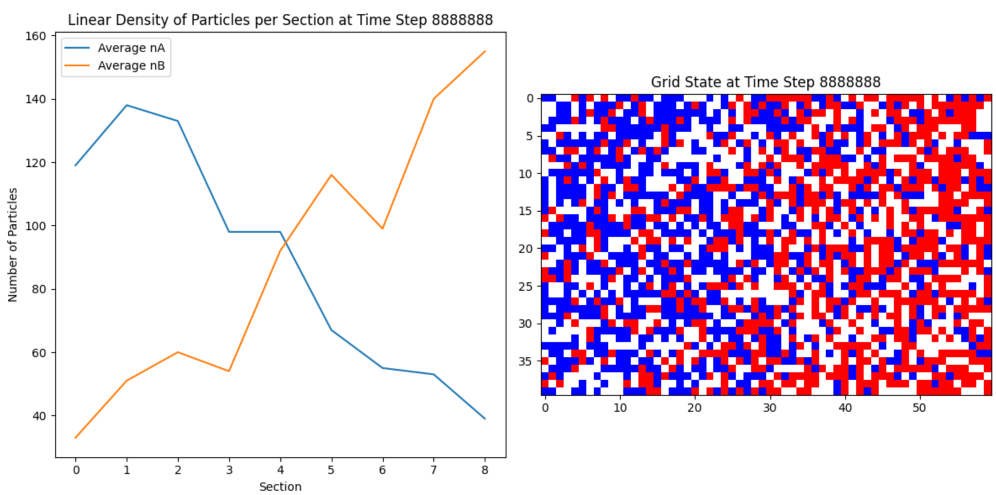

Figure 22: Linear Density and the Grid after 8888888 Steps

5.2 Plot a linear density plot with grid images at some time interval

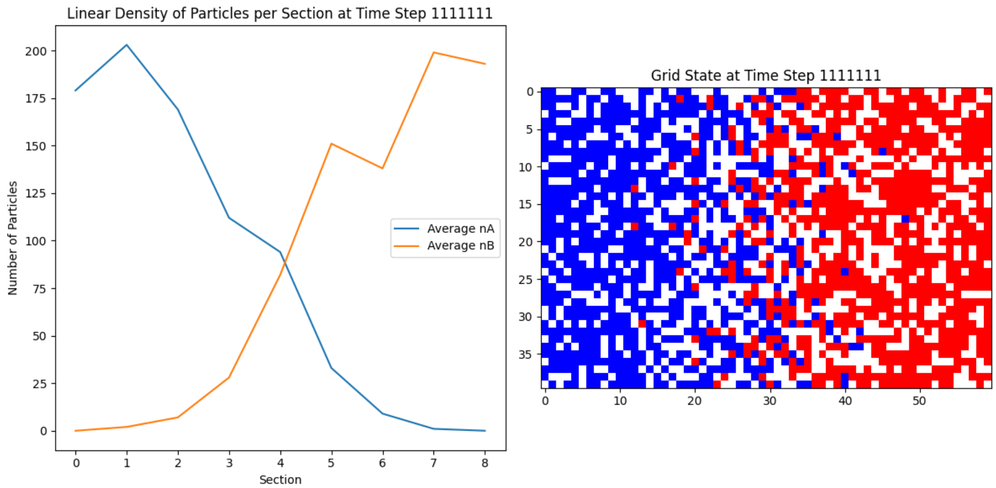

Figure 23: Linear Density and the Grid after 1111111 Steps

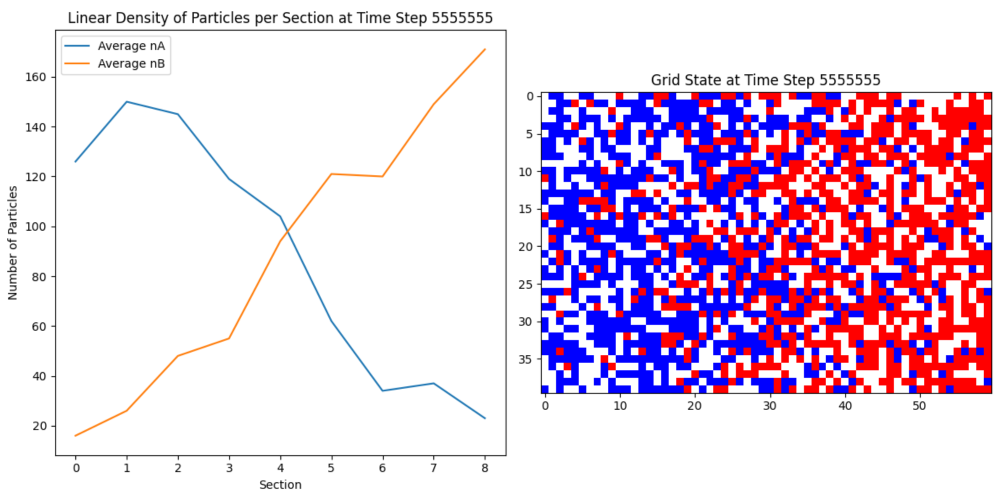

Figure 24: Linear Density and the Grid after 5555555 Steps

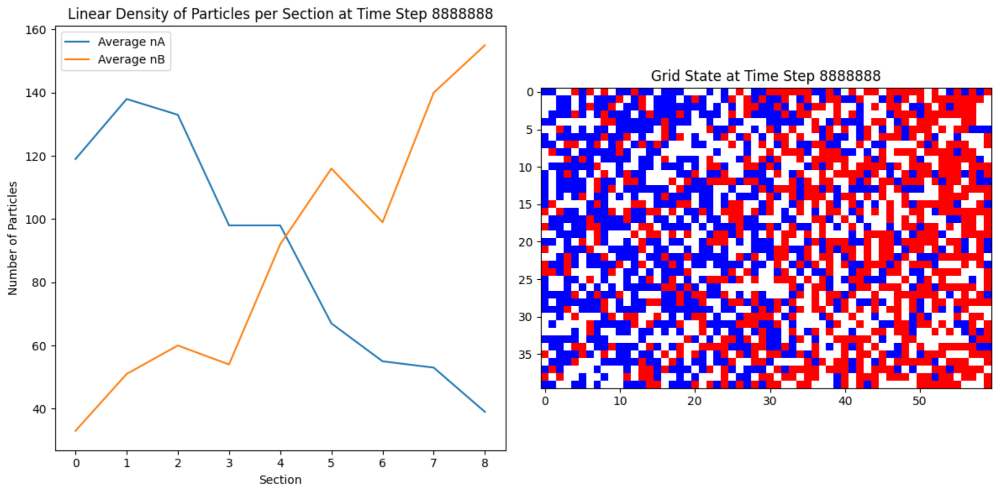

Figure 25: Linear Density and the Grid after 8888888 Steps

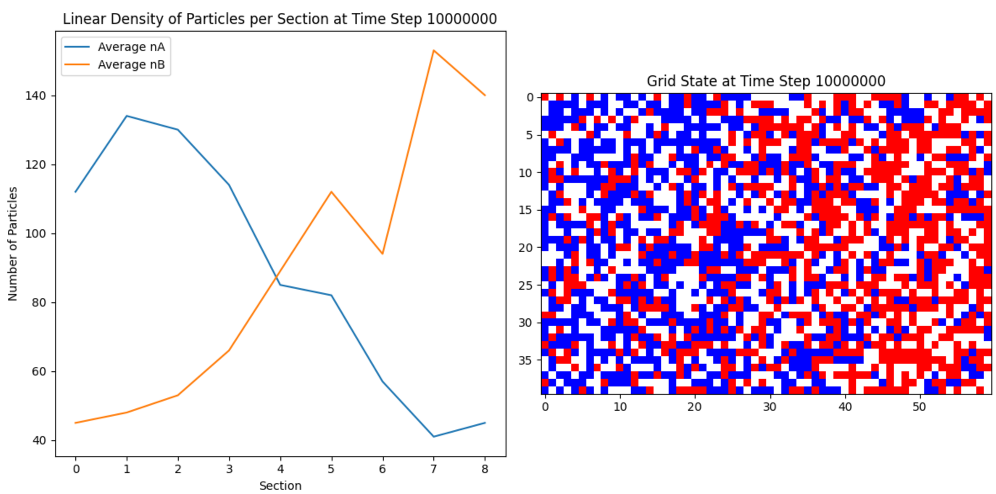

Figure 26: Linear Density and the Grid after 10000000 Steps

5.3 Average the densities over 100 trials for added accuracy and replot the densities

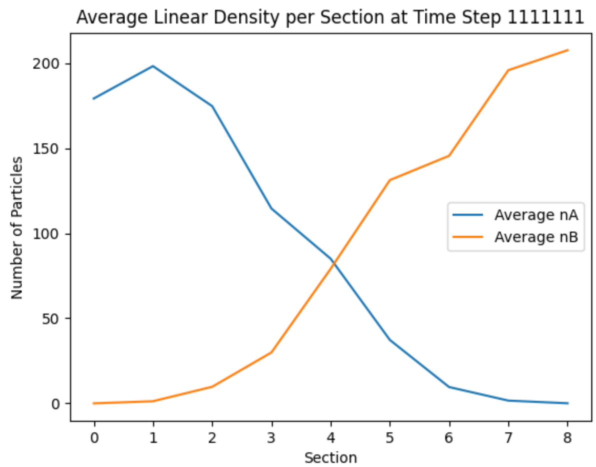

Figure 27: Average Linear Density and the Grid after 1111111 Steps

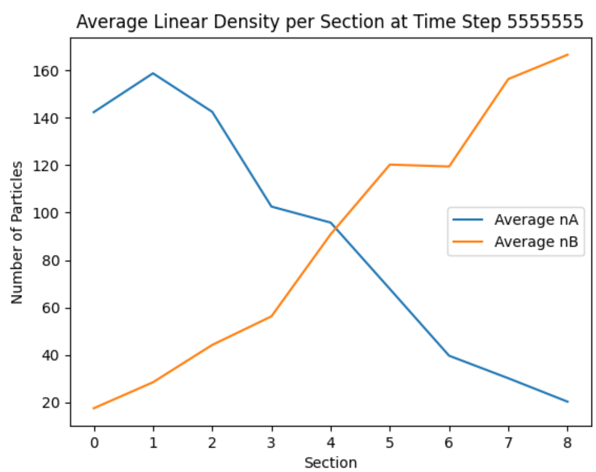

Figure 28: Average Linear Density and the Grid after 5555555 Steps

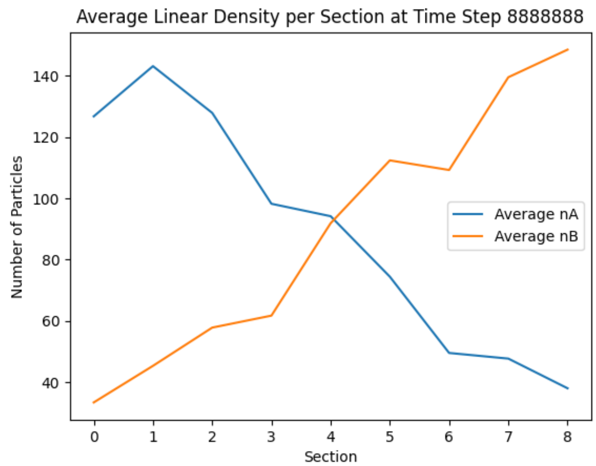

Figure 29: Average Linear Density and the Grid after 8888888 Steps

6 Conclusion

1. From Figure 10 and Figure 14 we observed that for getting a smooth Gaussian distribution the number of elements needs to be on higher side.
2. Figure 21 and Figure 22, we can say the the simulation of gasses mixing is a memory intensive job and we observed almost homogeneous mixture around the end of 10^7 iterations

Contribution

1. Thanks to Alison for helping me with the diffusion problem, specifically for making me understand finite difference algorithm and also for initially i took x^2 same as r^2, alison pointed out the mistake. And for cross checking out plots
2. Thanks to Sravya for helping me find the problem in my random walk algorithm. And for cross checking out plots

References

[1] Gaussian Random Number Generator. Available at: `https://www.tspi.at/2021/08/17/gaussianrng.html`