

Computational Physics Group Assignment 1

Christopher Flower, Long Li, Shen Yan, Wenzhe Yu

March 24, 2016

1 Introduction

In this work, the 2D random walk, diffusion equation, and mixing of two gases are studied by Python programs.

2 2D Random Walk

A 2-dimensional random walker is a point continuously taking steps of unit length in $\pm x$ or $\pm y$ direction on a discrete square lattice. Random walkers taking up to 100 steps are simulated, then the properties of 2D random walk are investigated by averaging over 10^4 walks.

The average of x-coordinate $\langle x_n \rangle$, of the square of x-coordinate $\langle (x_n)^2 \rangle$, and of the square distance $\langle (r_n)^2 \rangle$ of random walks with different steps are plotted in figure 1 (a), (b), and (c), respectively. $\langle x_n \rangle$ oscillates around zero, consistent with the theoretical expectation. $\langle (x_n)^2 \rangle$, by contrast, increases with steps with a slope of 1/2.

Figure 1 (c) shows that $\langle (r_n)^2 \rangle$ increases with steps as well, with a slope of 1. This can be explained by the symmetry between $\langle (y_n)^2 \rangle$ and $\langle (x_n)^2 \rangle$, leading the slope of $\langle (r_n)^2 \rangle$ be $1/2 + 1/2 = 1$.

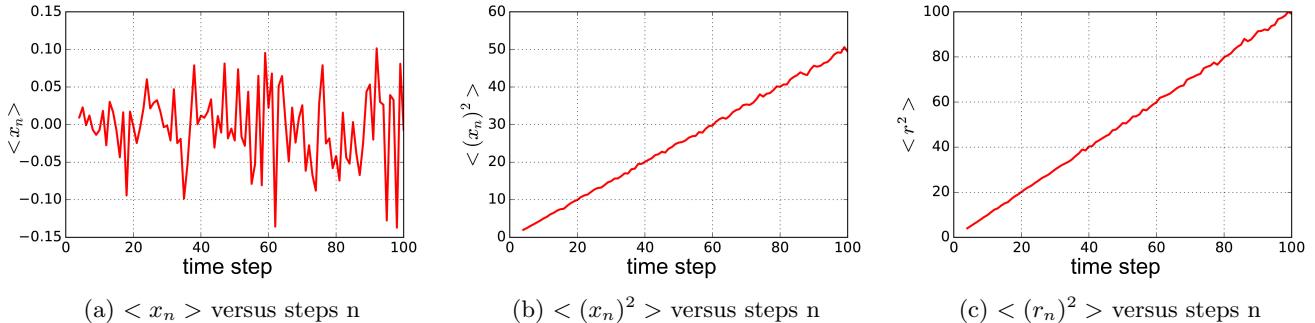


Figure 1: $\langle x_n \rangle$, $\langle (x_n)^2 \rangle$, and $\langle (r_n)^2 \rangle$ of 2D random walk.

3 Diffusion Equation

Consider the 1D Normal Distribution

$$\rho(x, t) = \frac{1}{\sqrt{2\pi\sigma^2(t)}} e^{-\frac{x^2}{2\sigma^2(t)}} \quad (1)$$

The spatial expectation value $\langle x^2(t) \rangle$ can be computed by

$$\begin{aligned}
\langle x^2(t) \rangle &= \int_{-\infty}^{\infty} x^2 \rho(x, t) dt \\
&= \frac{1}{\sqrt{2\pi\sigma^2(t)}} \int_{-\infty}^{\infty} x^2 e^{-\frac{x^2}{2\sigma^2(t)}} dx \\
&= \frac{1}{\sqrt{2\pi\sigma^2(t)}} \frac{1}{2} (\sqrt{2}\sigma(t))^3 \sqrt{\pi} \\
&= \sigma^2(t)
\end{aligned}$$

Here $\int_{-\infty}^{\infty} x^{2k} e^{-\frac{x^2}{a^2}} dx = \frac{(2k+1)!!}{(2k+1)2^k} a^{2k+1} \sqrt{\pi}$ ($k = 0, 1, 2, \dots$) is used.

The diffusion equation with a constant diffusion coefficient has the following form:

$$\frac{\partial u(r, t)}{\partial t} = D \nabla^2 u(r, t) \quad (2)$$

In one-dimensional case, equation (2) becomes

$$\frac{\partial u(r, t)}{\partial t} = D \frac{\partial^2 u(x, t)}{\partial x^2} \quad (3)$$

The first derivative in time and second derivative in space can be approximated by the finite difference:

$$\begin{aligned}
\frac{\partial u(r, t)}{\partial t} &= \frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} \\
\frac{\partial^2 u(x, t)}{\partial x^2} &= \frac{u(x + \Delta x, t) + u(x - \Delta x, t) - 2u(x, t)}{(\Delta x)^2}
\end{aligned}$$

Then equation (3) can be rewritten as

$$\frac{u(x, t + \Delta t) - u(x, t)}{\Delta t} = D \cdot \frac{u(x + \Delta x, t) + u(x - \Delta x, t) - 2u(x, t)}{(\Delta x)^2}$$

or equivalently,

$$u(x, t + \Delta t) = u(x, t) + D \cdot \Delta t \cdot \frac{u(x + \Delta x, t) + u(x - \Delta x, t) - 2u(x, t)}{(\Delta x)^2} \quad (4)$$

Starting from an initial box density profile, the 1D diffusion equation (3) is numerically solved based on equation (4). The density profile at later times is plotted in figure 2. The density are normally distributed, with different standard deviation σ . The value of σ at time t is obtained through Gaussian fitting. Figure 3 (a) shows the comparison between the fitted σ and the theoretical values calculated by $\sigma(t) = \sqrt{2Dt}$. The fitted values of σ agrees well with the theoretical values.

$\sigma(t) = \sqrt{2Dt}$ is verified by σ directly extracted from the maximum value of density u_{max} at $x = 0$.

$$\begin{aligned}
u_{max}(t) &= \rho(0, t) = \frac{1}{\sqrt{2\pi\sigma^2(t)}} \\
\Rightarrow \sigma(t) &= \frac{1}{\sqrt{2\pi u_{max}^2(t)}}
\end{aligned}$$

The maximum values u_{max} and derived *sigma* at time 0.2, 0.4, 0.6, 0.8, 1.0 are listed in table 1. The $\sigma(t)$ versus time t is plotted (log scale) in figure 3 (b) with a linear fit line, which has a slope of 0.478. This indicates that $\sigma(t) \propto \sqrt{t}$.

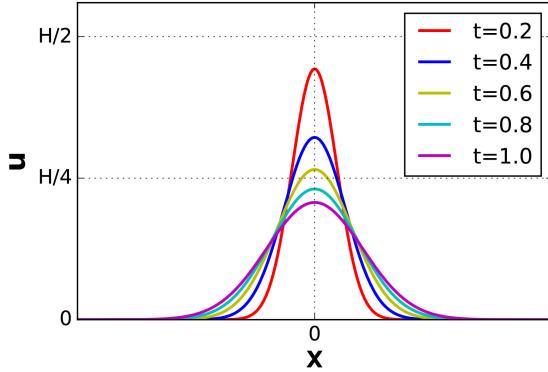


Figure 2: Solution of 1D diffusion equation at time 0, 0.2, 0.4, 0.6, 0.8, 1.0.

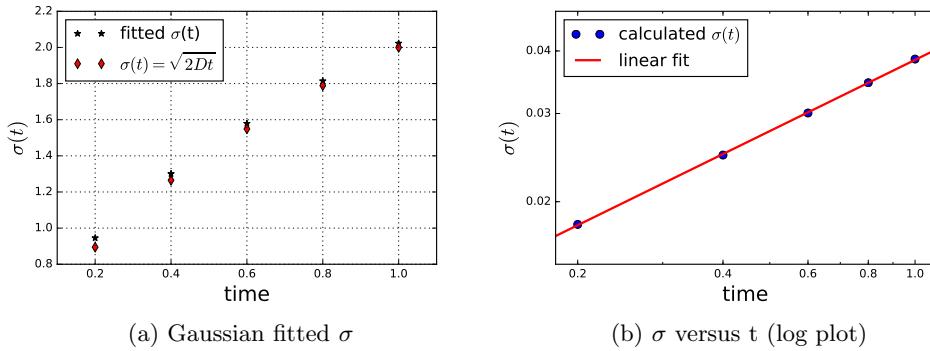


Figure 3: Comparison between calculated σ and $\sigma(t) = \sqrt{2Dt}$.

Table 1: u_{max} at time 0.2, 0.4, 0.6, 0.8, 1.0.

t	0.2	0.4	0.6	0.8	1.0
u_{max}	22.1449	16.0971	13.2666	11.5433	10.3539
$\sigma [\times 10^{-2}]$	1.8015	2.4783	3.0071	3.4560	3.8531

4 Mixing of two Gases

The mixing of two gases can be simulated with a number of random walkers. For simplicity, a 2D rectangular enclosure with dimensions 400×600 is studied. The initial profile is set up by fully populating the left third of the box with gas A and the right third with gas B.

Five versions of code with increasing mixing speed, v0, v1, v2, v3, and v4, are discussed here.

v0: A grid point is selected randomly from the 400×600 region, to perform a one-step random move. The move will be accepted if the destination is not occupied by another particle and is inside the box; it will be rejected if the destination is occupied or is outside the box. This process is iterated for sufficient times until a stable configuration is achieved.

According to our observation, this method works, but is extremely slow at the beginning of the gases mixing. Initially, only those particles that lie on the boundary of gas A and vacuum, and the boundary of gas B and vacuum, are free to move. The ratio of movable particles to all particles is $800/240000 = 1/300$. Each free particle only has one direction to move, i.e. only moving towards the vacuum is allowed, which has a probability of $1/4$. Therefore, at the beginning stage, the probability of a successful move is $1/300 \times 1/4 = 1/1200$. This low probability explains the low speed of gases mixing.

v1: An obvious improvement of v0 is creating a list of available sites, and randomly pick a site only from the list. After each successful move, the list of available sites is updated: $[i_{old}, j_{old}]$ moves to $[i_{new}, j_{new}]$ successfully,

then add [i_new, j_new] to the list, and remove [i_old, j_old] from the list. Theoretically, 1/3 iterations should be saved compared to v0, although it takes long time to fully mix the two gases as well.

v2: In a more aggressive strategy, the initial list of available sites only contains those 800 particles on the boundary. Then the list is gradually enlarged due to the largest distance of the movement of particles, until the list include all grid points. Another trick used in v2 is that after a grid point is selected, it always moves one step to a randomly chosen direction, unless no direction is free to move.

v3: The locations of all the particles are stored in set 1, while the locations with available neighbor sites are stored in set 2. Then a more precise list of available sites is the intersection of set 1 and set 2, which always guarantees that the chosen site is a particle with available neighbor sites.

v4: Further acceleration of the mixing process is achieved by increasing the step-size per move. For instance, if a particle is allowed to move 100 steps, instead of only 1 step, the two gases can be mixed in a few minutes. However, this method no longer makes physical sense. Thus v3 is not used for the analysis of gases configurations and linear population densities.

The configurations after different number of random moves are plotted in figure 4. In figure 4 (f), A and B gases are fully mixed. The status of mixing is more visible by the linear population densities, as shown in figure 5. A and B gases are separated at the beginning (figure 5 (a)). Finally the distribution of each gas is virtually uniform along the x direction (figure 5 (f)).

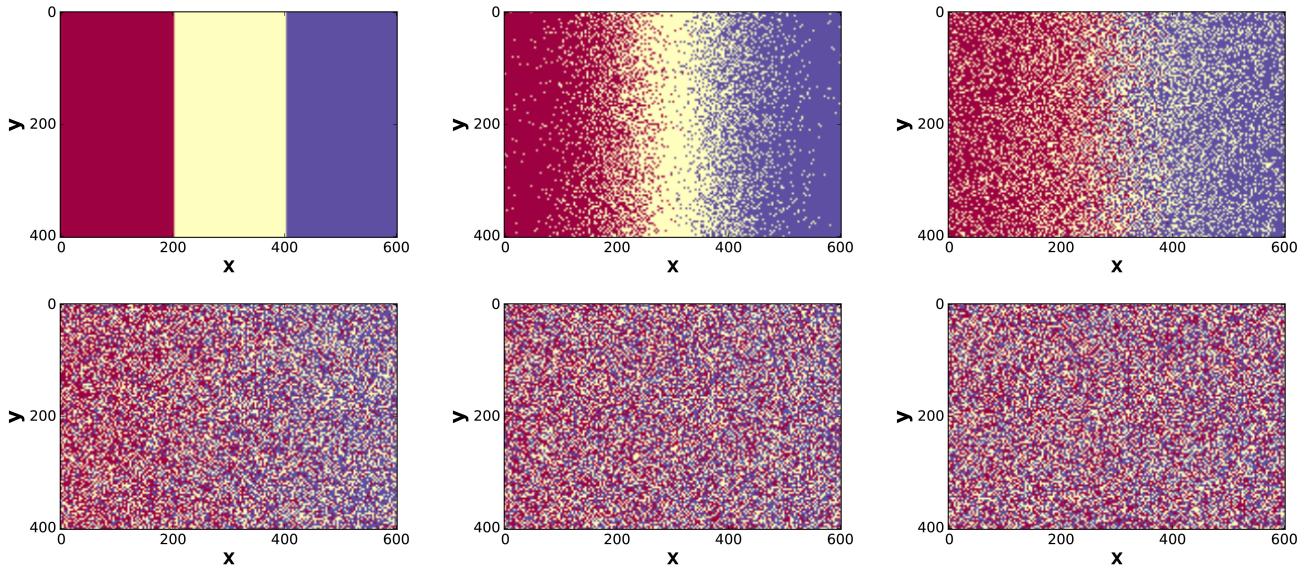


Figure 4: Gases mixing configurations after different number of steps.

In principle, two fully mixed gases should be uniformly distributed within the box. Due to the stochastic nature of the mixing simulation, the uniform distribution is never obtained by one single trial. The linear population densities averaged over 100 trials, illustrated in figure 6, show good agreement with the uniform distribution. The densities will gradually approach the perfect uniform distribution with increasing number of trials.

5 Conclusions

In this work, the 2D random walk, diffusion equation, and mixing of two gases are studied by Python programs. $\langle x_n \rangle$ of 2D random walk oscillates around zero, while $\langle (x_n)^2 \rangle$ and $\langle (r_n)^2 \rangle$ increases linearly with steps.

The solutions of 1D diffusion equation with an initial box density profile are normal distributions. The standard variance σ of the normal distribution is proportional to the square root of time \sqrt{t} .

In the gases mixing simulation, two gases are fully mixed after a large number of iteration steps. The final linear population density of each gas averaged over 100 simulations is a uniform distribution.

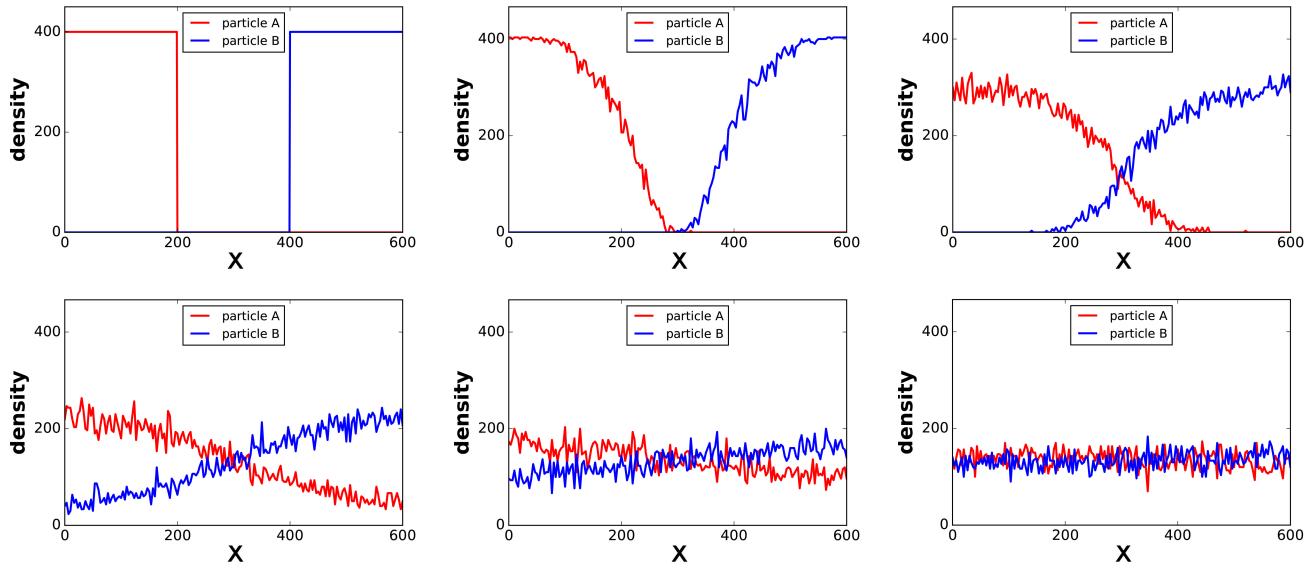


Figure 5: Linear population densities after different number of steps.

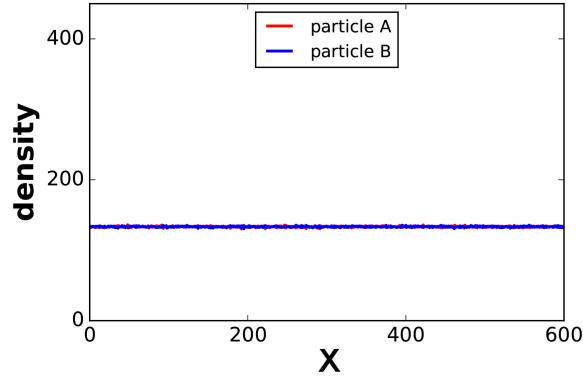


Figure 6: Linear population densities averaged over 100 trials.

The source code of this work is downloadable online at github.com/vyu16/PHY566-DUKE.

References

- [1] Notes of ***Computational Physics*** by Prof. S.A. Bass.