1. **Overview and goals**

The basic goals are the following:

1. Build up a reliable program to simulate the behavior of diffusion based on random walk theory.
2. Predict some performances of diffusion which are still not fully clear.
3. **Instructions**
   1. Set up
      1. Under your *src* directory, please create a new directory like *project.*
      2. Please copy the *canvas.go* and *simulator.go* file and put them into the directory above.
   2. Simulations

2.2.1 “FTCS” (forward time, central spacing) method.

The goal of “FTCS” mode is to provide a numerical analysis of the change in compositions within the samples. We will pick up some points between 0 and 2 and monitor the evolvement of the concentration profile.

You can run the program with following command:

**go run canvas.go simulator.go ftcs numNodes numSteps c**

where:

• ftcs is the mode, note that “FTCS” or “Ftcs” would be fine

• numNodes is the number of checking points, which should be an integer>1 (Report example takes numNodes=50)

• numSteps is the number of time steps, should be an integer>1

(Report example takes numSteps=5000)

• c is the boundary concentration, like c=0 or c=0.5

Note that you should ensure the FTCS method is stable. If you see the following message: “Error: it's not a stable FTCS method, probably wrong!”, you may consider taking a greater numSteps.

For a given numSteps, the program will select some steps which are in the range of numSteps, and output some txt files recording the concentration profile in the form “checking point: concentration” at those steps.

You will get the information stating if the system is steady after “numSteps” steps.

Extra points: you may change the constants like the **length** of x axis, the **end time** and the **diffusion coefficient** to build your own systems. Also you might be happy to set your own standard for steady state in CheckSteady() method

2.2.2

To run the pattern diffusion, you can use the following command:

**go run canvas.go simulator.go mode numParticles numSteps step1 step2…**

where:

• You need to choose your mode from [thin, thick, surface, slab] or you may get an error message.

• numParticles is the number of random walkers in a stack, should be an integer >0.

• numSteps is the steps you want the walkers to perform, should be an integer >0.

• step1… is the steps a walker can move, like –a +a.

In terms of the calculations of changing positions, the output will be a series of time steps and the mean square distance. These data are analyzed by a method called linear regression analysis. And the result is presented in the “numerical slope” and “theoretical slope” part.

“max moving distance” will record the maximum distance the particles have travelled, which is proportional to time step and step size in theory.

“max occurrence”: the max probability that a particle can stay in certain place.

“max displacement”: the furthest place certain particles can travel to, different from distance.