Assignment No :6: 1-D Simulation of tube-light

Sakthi Harish D T (EE19B054)

April 7^{th}, 2021

1 Abstract:

In this assignment ,we will,

1. Simulate a tube-light in One-Dimension
2. Visualize the electron population density and intensity at every point along the length of the tube light
3. Print the values of intensity along the length of the tube-light.
4. compare the graphs obtained for some other values of threshold velocity and probability of collision.

2 Introduction:

In a tube-light, the electrons originate from the cathode and move towards the anode due to the applied electric potential. The electric field that has been setup due to the applied potential in between ends of the tube light accelerates the electrons and thus energizes it. Therefore if some electron crosses a threshold energy, then it can excite the atom which it collides and thus it can emit light while de-excitation. But there is some probability with which the electron collides with an atom.

We create our simulation environment as a array of size 'n', where the tube is divided into 'n' sections. In each time instant, 'M' electrons are injected into the environment. We run the simulation for 'nk' turns. The electrons are unable to excite the atoms till they posses a threshold velocity of 'u0'. Beyond this velocity, there is a probability 'p' in each turn that a collision will occur and an atom gets excited. The electron's velocity reduces to zero if it collides(inelastic). Here the parameters discussed are taken from the user (`sys.argv`), along with a set of default values.

3 Initializing Vectors:

We create vectors of size 'n*M' to hold the information of the electrons and initialize them to zero, this includes

1. Electron position (xx)
2. Electron velocity (u)
3. Displacement in current turn (dx)

In order to accumulate the information gathered in each loop of simulation, we create lists for the following

1. Intensity of emitted light(I)
2. Electron position(X)

3. Electron velocity(V)

For each turn, we record all electron positions and velocities in these arrays. If they had a collision, we also record that as emitted light. We do not know the length of these arrays. So we create them as lists and extend them as required. So, this is a bit slow.

4 Performing the iteration:

As mentioned above, we loop 'nk' times and update the electron position, velocity and apply the threshold conditions and update their values and finally retrieve the corresponding Intensity, position and velocity values of the electron along all sections of the tube light. This is done as per the following code: ***Python Code:***

```
for k in range(nk-1):
    ii= np.where(xx>0)        #finding the indices of positions greater than zero
    dx[ii]= u[ii]+ 0.5         #Updating the vectors
    xx[ii]= xx[ii]+ dx[ii]
    u[ii]= u[ii]+ 1

    anode_hit= np.where(xx[ii]>n)      #Finding the indices of electrons that reached anode
    xx[ii[0][anode_hit]]= 0        #Updating the values after hitting the anode
    u[ii[0][anode_hit]]= 0
    dx[ii[0][anode_hit]]= 0

    kk= np.where(u>= u0)        #Finding the indices of energetic electrons
    ll= np.where(np.random.rand(len(kk[0]))<= p)      #Vector of indices, containing indices of kk
    kl= kk[0][ll]               #Vector of indices of kk np.where collision occurs
    u[kl]= 0              #Updating the velocity of electrons suffering collision

    xx[kl]= xx[kl]- (dx[kl]* np.random.rand(len(kl)))     #randomly deciding np.where the collision
    I.extend(xx[kl].tolist())
    m= int(np.random.rand()*Msig)+ M

    free= np.where(xx==0)        #Finding free spaces in the grid
    n_inject= min((n*M)-len(free),m)

    xx[free[:n_inject]]= 1    #Updating the values of vectors corresponding to the injected electr
    u[free[0][:n_inject]]= 0    # (i.e) position= 1; velocity= 0 ; displacement= 0
    dx[free[0][:n_inject]]= 0
    X.extend(xx.tolist())
    V.extend(u.tolist())
```

5 Plotting the Graphs:

After we do the iterations, we finally plot the histogram plot of the Population of electrons and the Intensity plot, along with the Electron space phase plot. These are shown in the following figures:

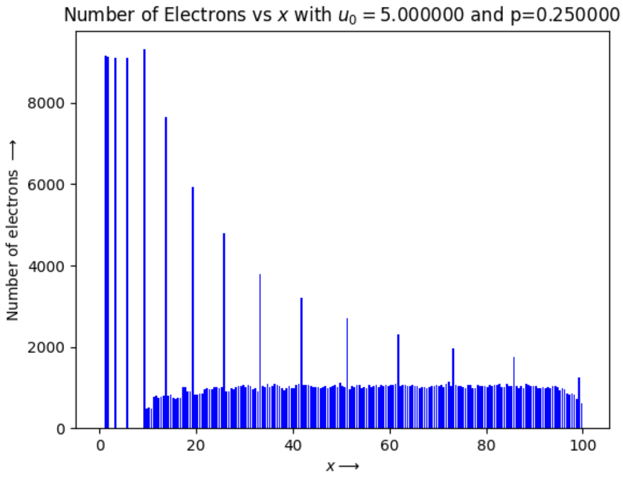

Figure 1: Population plot of the electrons

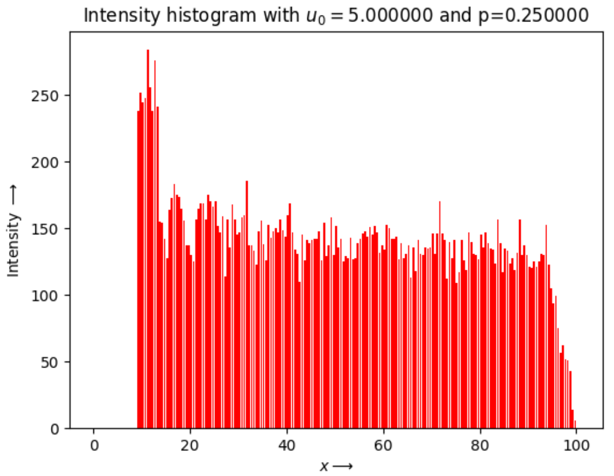

Figure 2: Intensity plot of the electrons

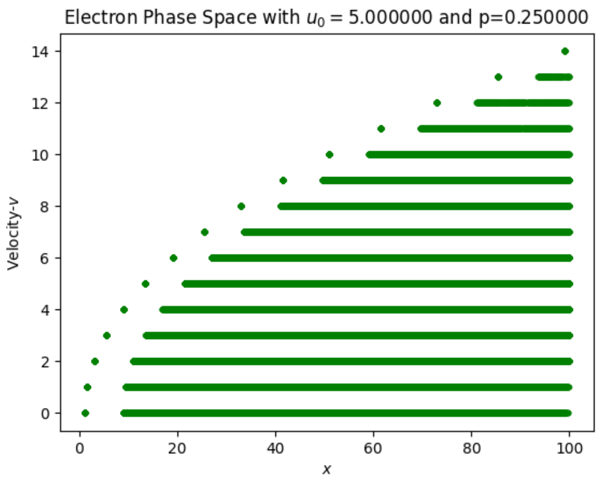

Figure 3: Electron space phase plot

We also make a table of the Intensity data against the `xpos`, this is as shown below:

```
Intensity Data:
 xpos   count

0.25   0.000000
0.75   0.000000
1.25   0.000000
1.75   0.000000
2.25   0.000000
2.75   0.000000
3.25   0.000000

..   ...    ...
97.75   51.000000
98.25   42.000000
98.75   29.000000
99.25   21.000000
99.75   5.000000
100.25   0.000000
```

6 Analysing with different set of values:

1. In this section we try to vary the parameters that decides the simulation pattern and visualize the same and seek out some inferences from the same.
2. One specific observation is that as the probability increases the graphs become more variate i.e. the superposition of the probability graphs become more separated. We also see that the maximum intensity also increases as the electrons get ionized more often.
3. Also as the cutoff velocity is increased the initial excitation happens at a higher value of x. This is because the electron has to travel longer distances to be able to reach cutoff frequency.

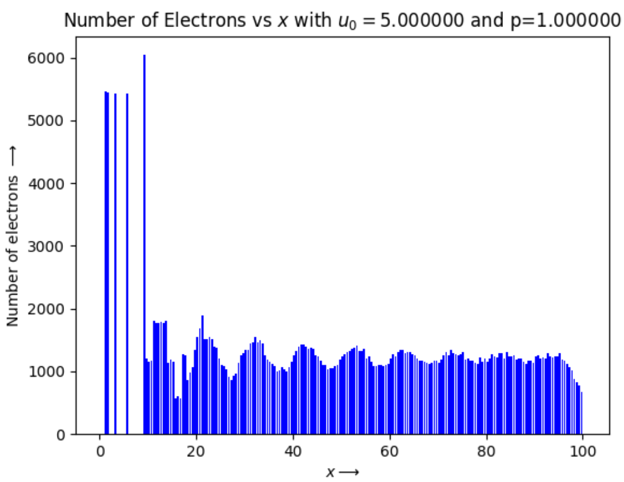

Figure 4: Population plot of the electrons

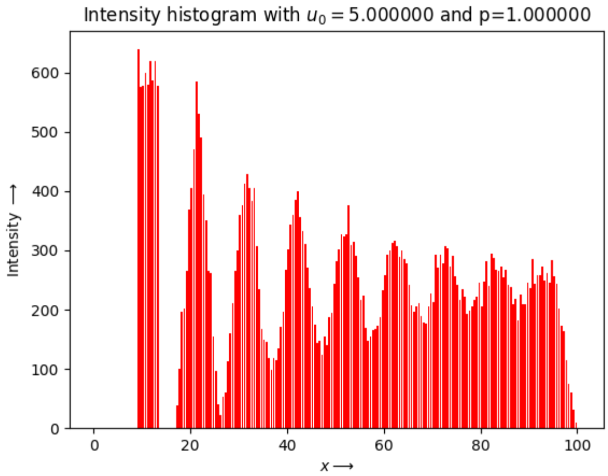

Figure 5: Intensity plot of the electrons

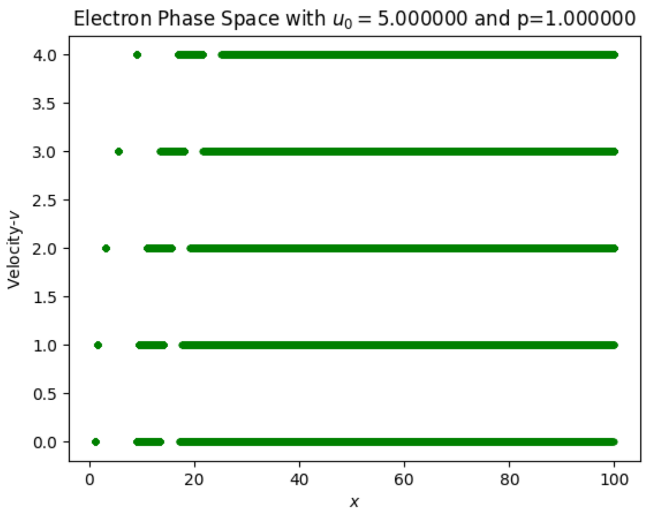

Figure 6: Electron space phase plot

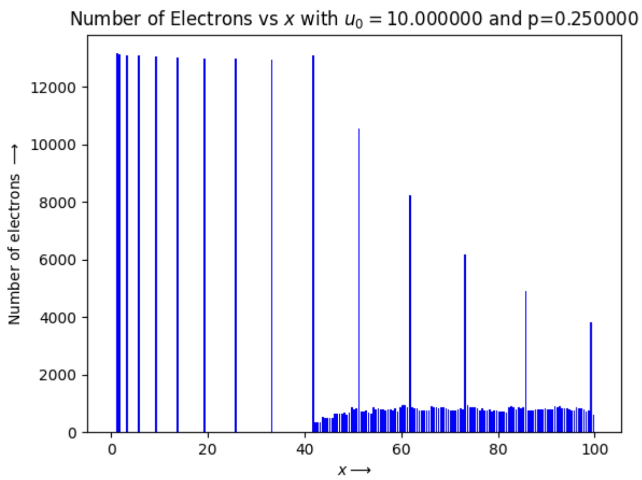

Figure 7: Population plot of the electrons

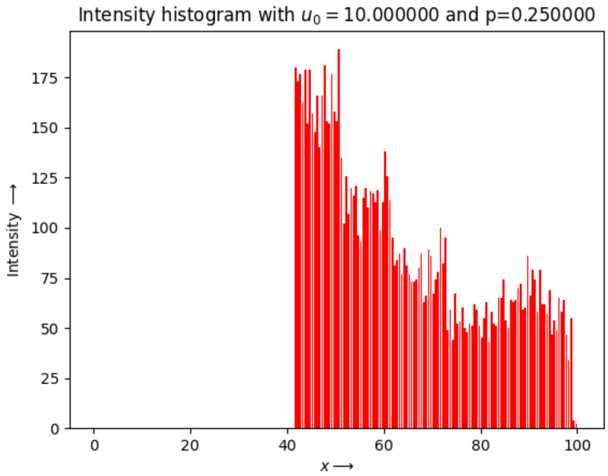

Figure 8: Intensity plot of the electrons

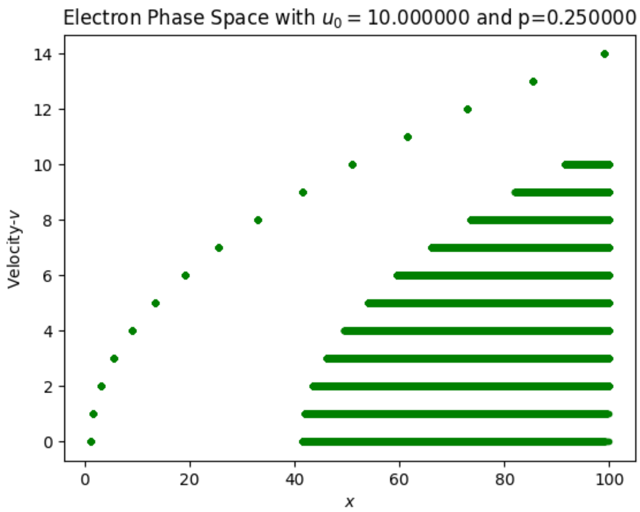

Figure 9: Electron space phase plot

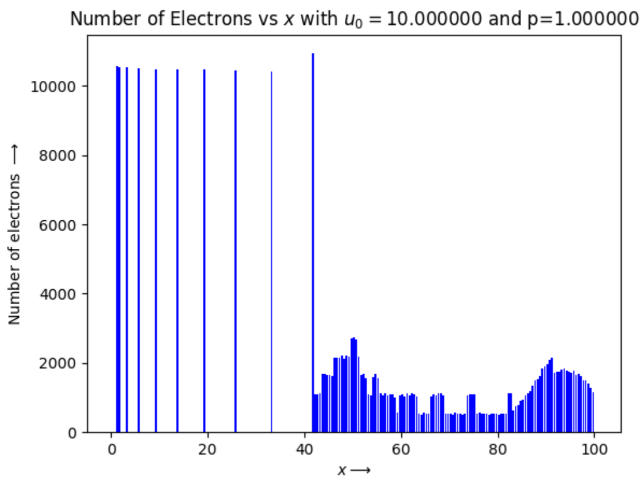

Figure 10: Population plot of the electrons

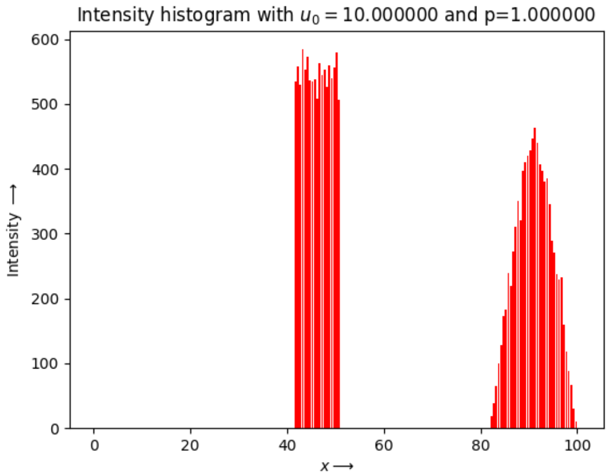

Figure 11: Intensity plot of the electrons

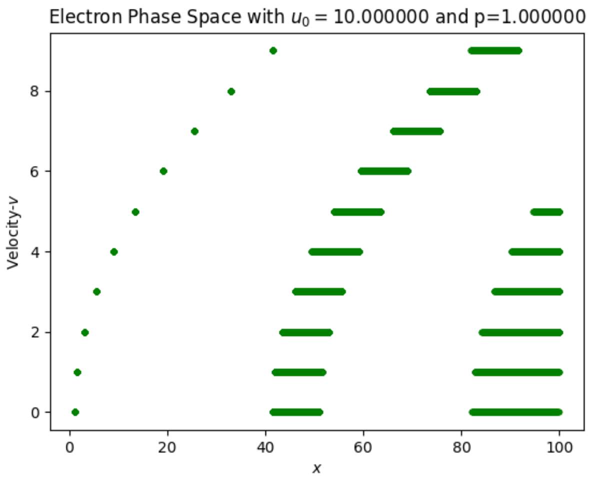

Figure 12: Electron space phase plot

7 Conclusion:

1. From the Intensity plot of the electrons(for p=0.25; u_0=5), we see that it reaches a maximum at around x=15 and stays like that for few bins and then it decreases.
2. This is because of the fact that the electron comes to rest after collision with other atom. So it has to start regaining energy from zero to be able to excite the atom for emitting light.
3. From the Electron phase space, we see that there are certain specified values of velocity that can occur at a particular value of x, thus we could say that the velocities are quantized.
4. We have analysed a 1-D tubelight using python and visualized it using histograms.