Tubelight Simulation

Rishabh Adiga ee19b135

April 2021

1 Introduction

In this assignment, we model a tubelight as a one dimensional space of gas in which electrons are continually injected at the cathode and accelerated towards the anode by a constant electric field.We simulate the tubelight over a given number of turns(timesteps) starting with no electrons initially. The electrons can ionize atoms if they have a velocity greater more than some given threshold velocity, leading to an emission of a photon,but this a probabilistic process.We plot various aspects of this simulated tubelight like electron densities and intensities.

2 Tubelight simulation

First we have to define certain parameters that should either be given as inputs by the user or should take up some default values.The parameters are:

- n which is the spatial grid size, that is the one dimensional length of the tubelight.Default value is 100.
- M which is the number of electrons injected per turn(timestep).Default value is 5.
- nk which is the number of turns to simulate.Default value is 500.
- u0 which is the threshold velocity above which ionization is possible.Default value is 5.
- p which is the probability that ionization will occur.Default value is 0.25 .
- Msig which is the variance in the actual number of electrons.Default value is 1.

Now we define a function Tubelight_sim() whose input parameters are all of the above parameters. Inside this function, our entire simulation is done.That is, we go through a loop with the simulation inside it for nk number of times.Before the loop we initialize variables xx(electron position),u(electron velocity),dx(displacement in current iteration/turn) to a list containing all zeros of length n*M.We also initialize I,X,V to empty lists which are just used to accumulate information of all the turns. Inside the loop , we execute the following:

- Displacement and Velocity are updated with time steps as: $dx_i = u_i\Delta t + \frac{a(\Delta t)}{2} = u_i + 0.5$ and $u_i = u_i + 1$
- Determine which electrons have hit the anode and set $x = 0, u = 0$ for these electrons
- Find out which electrons have velocity more than threshold velocity and along together with a probability of ionization, determine the number of photons emmited at a particular position.
- Assuming inelastic collision, set their velocities to zero, and random position.
- Inject a randomly distributed number of electrons, with the given mean M and variance M_sig.
- The injected electrons go to unused indices.

The code for this tubelight function is given below:

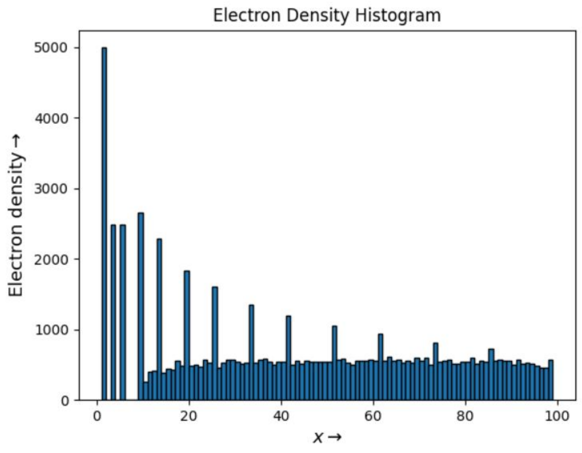
```
def Tubelight_sim(n, M, nk, u0, p, Msig):
    # Initializing the required lists to list of zeros
    xx = np.zeros(n*M)
    u = np.zeros(n*M)
    dx = np.zeros(n*M)

    """The following lists are kept empty initially as
    we do not know their actual lengths.As and when required,
    we append the values to these lists"""
    I = []
    X = []
    V = []

    # Looping over the turns
    for k in range(nk):

        """For the first iteration we check where xx>0.
        But obviously this is not required as locations are
        zero for first iteration.(So this can be removed,but
        I have kept it just to see the logic).For the remaining
        iterations,we dont do this because it has already been
        done at the ending of previous iteration."""
        if k == 0:
            ii = np.where(xx > 0)[0]
```

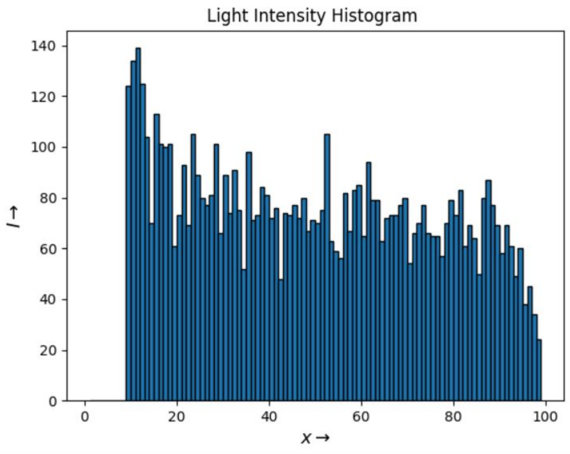
```
# Moving the electrons that exist by calculating displacement and adding to current
dx[ii] = u[ii]+0.5
xx[ii] += dx[ii]

# Increasing the velocity of the electrons that exist.
u[ii] += 1

# Determining which particles have hit the anode
anode_hit = np.where(xx >= n)
# The positions displacements and velocities of these are set to 0
xx[anode_hit] = 0
u[anode_hit] = 0
dx[anode_hit] = 0

# Determining electrons with velocity more than threshold velocity
kk = np.where(u >= u0)[0]
# Of these, which electrons are ionized (Probabilistic process)
ll = np.where(np.random.rand(len(kk)) <= p)[0]
kl = kk[ll]

# Reset the velocities of these electrons(ionized) to zero
u[kl] = 0
# The collision could have occurred at any point between the previous xi and the cu
xx[kl] -= dx[kl]*np.random.rand()

# Excited atoms at this location resulted in emission from that point.
I.extend(xx[kl].tolist())

# Inject M new electrons
m = int(np.random.rand()*Msig+M)

# Add them to unused slots.
Slots_add = np.where(xx == 0)[0]

"""We dont need to check if m>len(Slots_add) because the
below code will automatically add a maximum of only len(Slots_add)
number of electrons even if m>len(Slots_add)"""
xx[Slots_add[0:m]] = 1
u[Slots_add[0:m]] = 0

# Finding all the existing electrons.We add their positions and velocities to the X
existing_electrons = np.where(xx > 0)[0]
X.extend(xx[existing_electrons].tolist())
V.extend(u[existing_electrons].tolist())
# We set ii directly equal to existing electrons here so that we dont have to do it
```

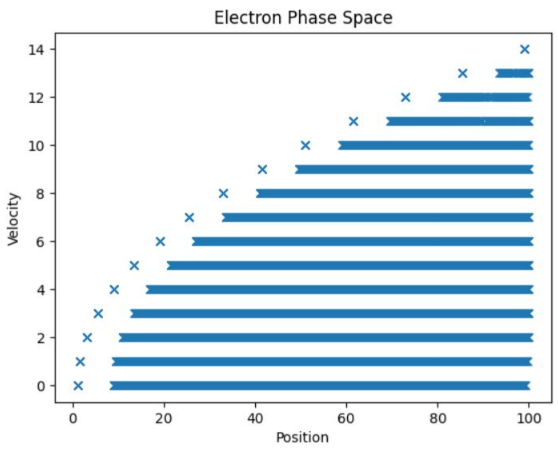
```
        # This ii will serve as the ii for next iteration
        ii = existing_electrons

    return X, V, I  # X,V,I are returned after all iterations are complete
```

3 Plots

Using the function above, The X,V,I lists are obtained after nk iterations. We plot the electron density histogram,light intensity histogram and the phase space of the electrons.The plots obtained are shown below:







The above plots are all with the default values for all parameters.

4 Table of Intesity data

We use the hist function to obtain the population counts and bin values which are used to output the table of intensity data.The code and the table are as follows:

```
ints, bins, trash = pylab.hist(I, bins=n)
xpos = 0.5*(bins[0:-1]+bins[1:])
print("Intensity Data")
print("xpos \t count")
for i in range(len(ints)):
    print(format(xpos[i], ".3f"), '\t', ints[i])
```

```
Intensity Data
xpos    count
9.453   107.0
10.356   117.0
11.258   126.0
12.160   137.0
13.063   107.0
13.965   60.0
14.867   76.0
15.770   96.0
16.672   100.0
17.574   88.0
18.477   90.0
19.379   51.0
20.281   66.0
```

....	..
85.248	56.0
86.151	66.0
87.053	72.0
87.955	71.0
88.858	67.0
89.760	70.0
90.662	46.0
91.564	65.0
92.467	55.0
93.369	46.0
94.271	46.0
95.174	44.0
96.076	34.0
96.978	44.0
97.881	27.0
98.783	16.0

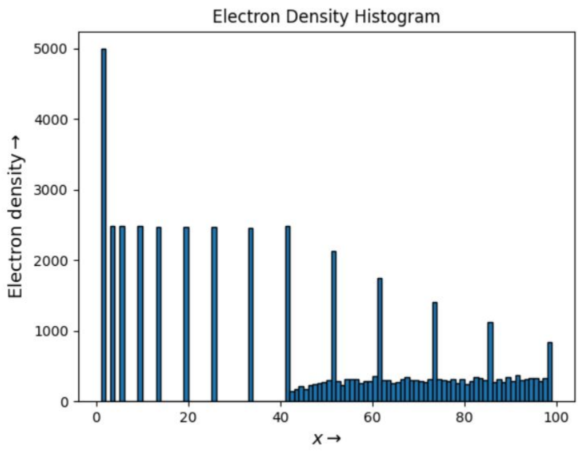
5 Simulations with different parameters

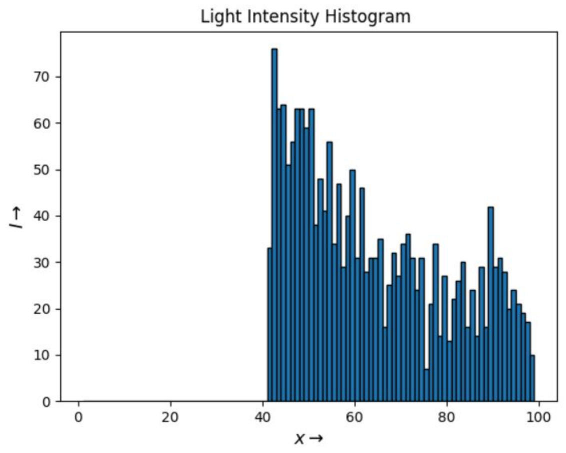
Let us try the following different parameter sets.

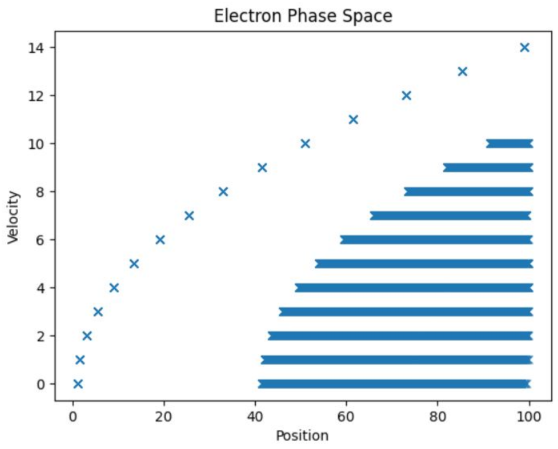
1. n=100, M = 5, nk =500, u0=10, p=0.25,Msig=1 (Higher threshold velocity).
2. n=100, M = 5, nk =500, u0=5, p=0.7 ,Msig=1 (Higher probabilitiy of collision).
3. n=100, M = 5, nk =500, u0=5, p=0.25 ,Msig=3 (Higher variation of randomness).

The plots for each of these cases are as follows.

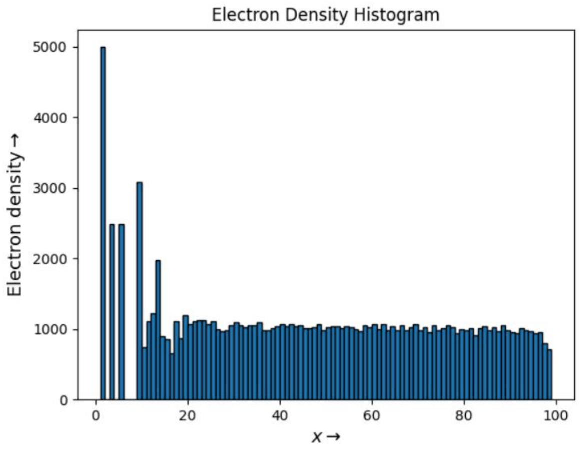
5.1 Case 1 (n=100, M = 5, nk =500, u0=10, p=0.25,Msig=1)

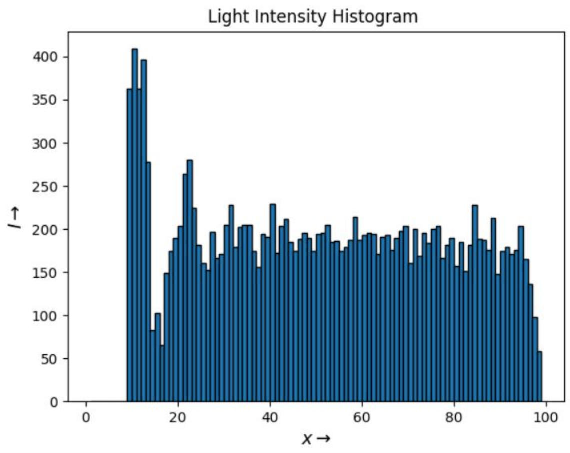


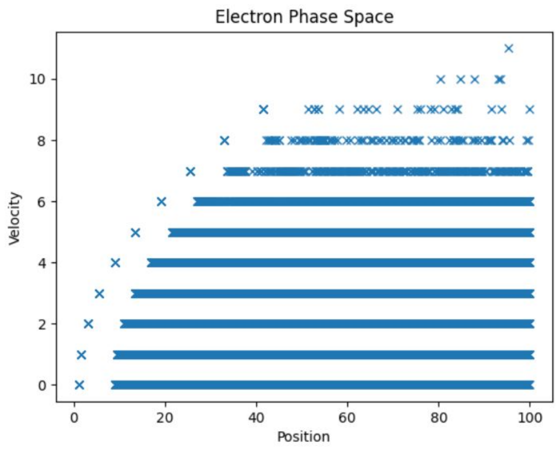




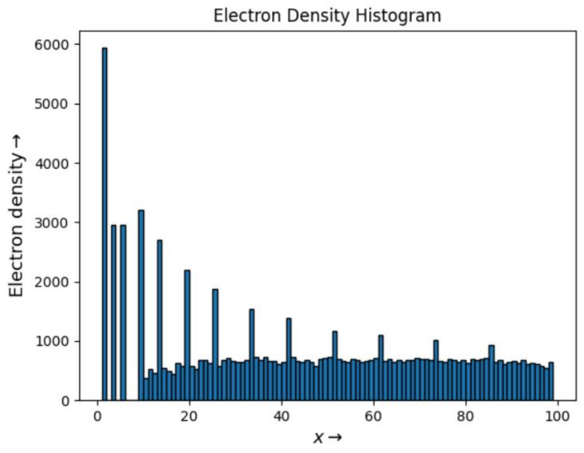
5.2 Case 2 (n=100, M = 5, nk =500, u0=5, p=0.7 ,Msig=1)

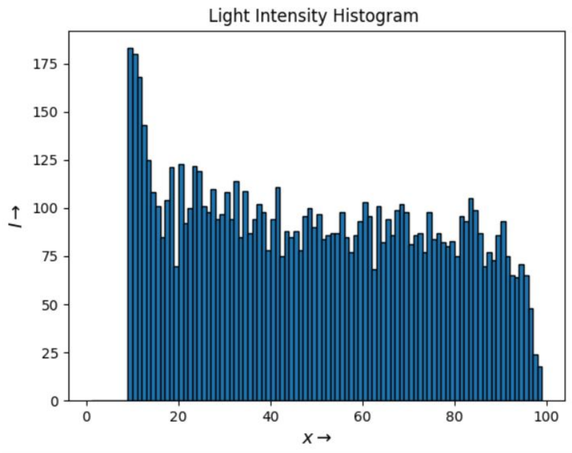


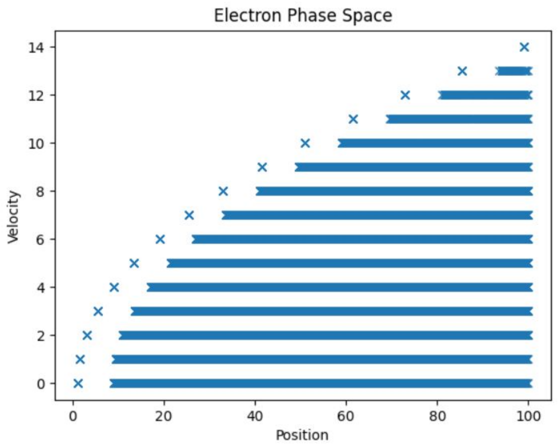

Light Intensity Histogram
I →
x →
0
50
100
150
200
250
300
350
400
0
20
40
60
80
100


Electron Phase Space
Velocity
Position
0
2
4
6
8
10
0
20
40
60
80
100

5.3 Case 3 (n=100, M = 5, nk =500, u0=5, p=0.25 ,Msig=3)







6 Conclusions

From the plots of the simulation with default parameters, we see that that there is no emission till after some length from the beginning of the tubelight.This is clearly because the electrons need to cross the threshold velocity before which they cannot cause any emission.We also observe that we have a peak at the first spot were intensity in non-zero.This is because lot of electrons would have crossed the threshold velocity here and cause a lot of emission of light.The electron phase plots show the constant acceleration all electrons initially undergo.
Now,from the 3 alternate parameter sets, we observe the following:

1. The first peak occurs after a larger distance as the electrons need more distance to cross the threshold velocity.

2. The intensity of the first peak has increased as more electrons cause emission. Now,there are less local minimas in the intensity plot

3. There is an increase in the overall intensity of light which is happening because there will be more electrons injected and this will increase electron density values too.