EE2703: Assignment 6

Sarthak Vora (EE19B140)

April 12, 2021

1 Abstract

In this report, we look at solving the simulation of a tublelight, in 1-Dimension. We use a 1-Dimensional model of the tubelight and make certain simplifying assumptions that allow us to simulate with ease.

2 Introduction

Consider electrons being emitted by the cathode with zero energy, and accelerate in this field. When they get beyond a threshold energy E_0, they can drive atoms to excited states. The relaxation of these atoms results in light emission. In our model, we will assume that the relaxation is immediate. The electron loses all its energy and the process starts again.

Those electrons reaching the anode are absorbed and lost. Each "time step", an average of N electrons are introduced at the cathode. The actual number of electrons is determined by finding the integer part of a random number that is "normally distributed" (Gaussian Distributed) with standard deviation of 2 and mean 10.

3 Description

We create a simulation universe. The tube is divided into **n** sections. In each time instant, **M** electrons are injected. We run the simulation for **nk** turns. The electrons are unable to excite the atoms till they have a velocity of **u0**. Beyond this velocity, there is a probability **p** in each turn that a collision will occur and an atom excited. The electron's velocity reduces to zero if collision takes place.

The default values taken for the above mentioned variables are as follows -

n = 100 : *Spatial grid size*
M = 5 : *Number of electrons injected per turn*
nk = 500 : *Number of turns to simulate*
u0 = 5.0 : *Threshold velocity*
p = 0.25 : *Probability that ionization will occur*

We keep track of the following parameters of the following parameters of the electrons -

```
Electron position xx
Electron velocity u
Displacement in current turn dx
```

We update the displacement as follows -

$$dx_i = u_i \Delta t + \frac{1}{2} a (\Delta t)^2 = u_i + 0.5 \tag{1}$$

When a particle reaches the anode, it's velocity and position is reset to zero. We find electrons which have crossed the threshold velocity and which ones emit photons, determined by a **Gaussian Distribution**. We then inject a number of electrons into the tubelight using a probability distribution.

The above logic is implemented in the code shown below.

4 Code

```
xx = np.zeros(n*M)  # electron position
u = np.zeros(n*M)   # electron velocity
dx = np.zeros(n*M)  # electron displacement
np.random.seed((int)(seedValue))   # A seed given so that uniform numbers results across runs for the rand

I = []  # This is used to store the photons generated at each location in each turn
X = []  # This stores the position of every electron after each turn
V = []  # Stores the velocity of the electron after each turn


def electronCollision(u,xx,dx,kl):    # This function does the collision update as per the question
    u[kl] = 0.0
    xx[kl] = xx[kl] - dx[kl]*np.random.rand(1)[0]


def electronCollisionModified(u,xx,dx,kl):   # This function does the collision update more accurately tak
    t = np.random.rand(1)[0]
    xx[kl] = xx[kl] - dx[kl]
    u[kl] = u[kl] - 1.0
    xx[kl] = xx[kl] + u[kl]*t + 0.5*t*t
    u[kl] = 0.0


ii = []
for k in range(1,nk):
    dx[ii] = u[ii] + 0.5
    xx[ii] = xx[ii] + dx[ii]
    u[ii] = u[ii] + 1.0

    jj = np.where(xx > n)[0]
    dx[jj] = 0.0
    xx[jj] = 0.0
    u[jj] = 0.0

    kk = np.where( u >= u0 )[0]
    ll = np.where(np.random.rand(len(kk)) <= p)
    kl = kk[ll]

    electronCollisionModified(u, xx, dx, kl)
    I.extend(xx[kl].tolist())

    m = np.random.randn()*Msig + M
    ll = np.where(xx == 0)[0]
    maxElec = min(len(ll),(int)(m))
    xx[ll[0:maxElec]] = 1.0
    u[ll[0:maxElec]] = 0.0

    ii = np.where(xx > 0)[0]
    X.extend(xx[ii].tolist())
    V.extend(u[ii].tolist())
```

5 Simulation

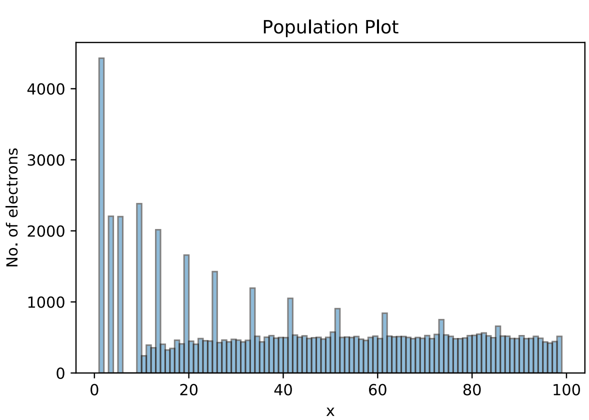
We plot the following plots for different values of the simulation parameters -

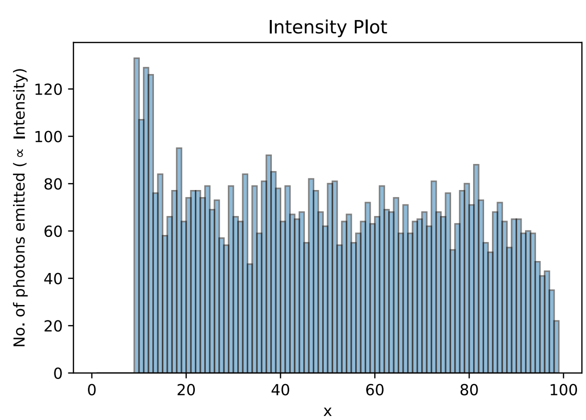
- Light Intensity Histogram
- Electron Density Histogram
- Electron Phase Space Plot

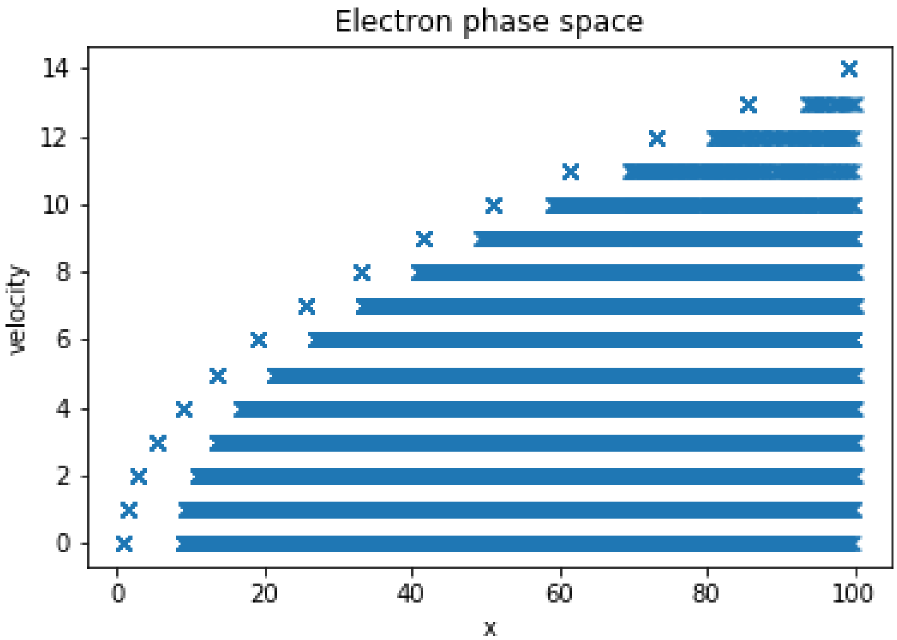
5.1 Code

```
fig, axes = plt.subplots(1, 2, figsize=(20, 7))

# Population plot for electron density
axes[0].hist(X,histtype='bar',range=(10,n), bins=np.arange(1,n,n/100),ec='black',alpha=0.5)
axes[0].set_title('Population Plot')
axes[0].set_xlabel('x')
axes[0].set_ylabel('No. of electrons')

# Population plot for intensity of emitted light
axes[1].hist(I,histtype='bar',range=(10,n), bins=np.arange(1,n,n/100),ec='black',alpha=0.5)
axes[1].set_title('Intensity Plot')
axes[1].set_xlabel('x')
axes[1].set_ylabel('No. of photons emitted ($\propto$ Intensity)')
plt.show()

plt.plot(X,V,'x')
plt.title('Electron phase space')
plt.xlabel('x')
plt.ylabel('velocity')
plt.show()
```

5.2 Plots







5.3 Observation

The above code also prints the Intensity values as a table. The **Intensity values** table is shown below as follows -

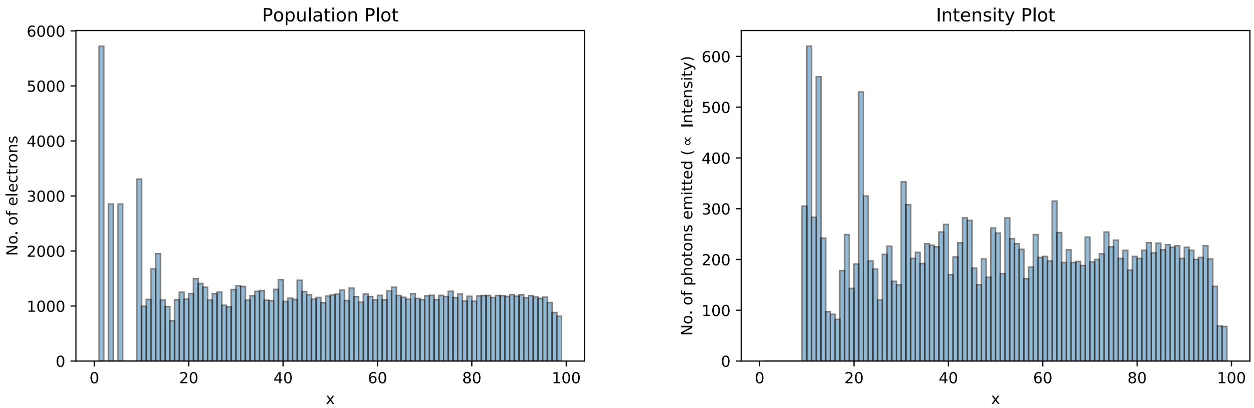
	Position	**Count**
0	1.5	0.0
1	2.5	0.0
2	3.5	0.0
3	4.5	0.0
4	5.5	0.0
..	...	...
93	94.5	47.0
94	95.5	41.0
95	96.5	43.0
96	97.5	35.0
97	98.5	22.0

- From the above **plots** and **Intensity table**, it can be noticed that the intensity is zero for positions less than around 15.
- The above phenomenon is because the electrons are building up their energy till **position 15**, leading to an almost **zero** value of intensity till around 15.
- There is a **peak** in the beginning, however the intensity decays because of the other peaks, which are less prominent.

6 Simulation (varying the parameters)

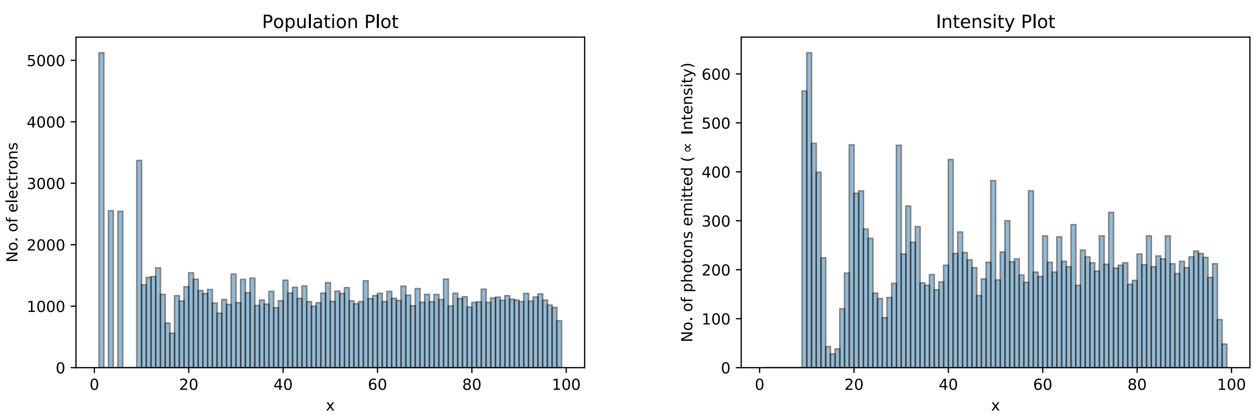
Here, we plot the same plots as shown above, but with varied parameters. This will give a better insight on the distribution of **electron density** and **intensity**. We simulate using the following parameters (varying **p** and keeping other parameters constant at default) -

```
n = 100
M = 5
nk = 500
u0 = 5
p = 0.7
```



Now, we simulate using the following parameters (varying **p** and keeping other parameters constant at default) -

```
n = 100
M = 5
nk = 500
u0 = 5
p = 0.9
```

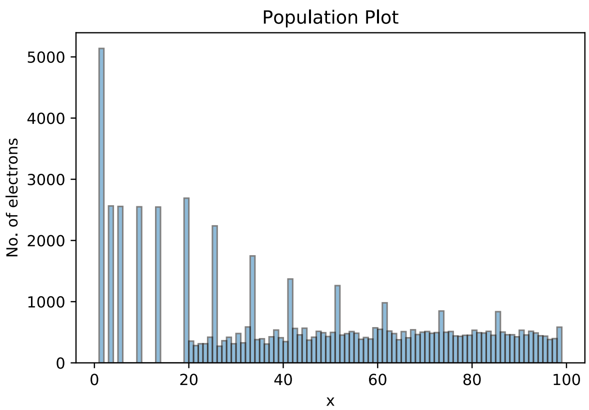


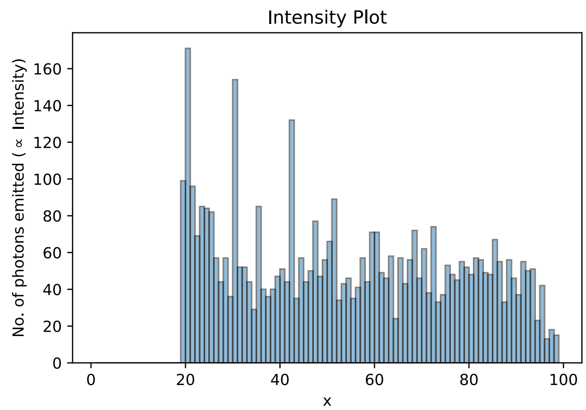
6.1 Observation

- We see that as we **increase p** keeping u0 constant, the **bright regions** become more clearly **separated** by dark intervals.
- Electron **density** does not change significantly upon **changing** p.
- However, as p **increases**, the chances of more number of electrons **colliding** also increases, hence the electron **density** also increases, but by a small margin.

Now, we simulate using the following parameters (varying **u0** and keeping other parameters constant at default) -

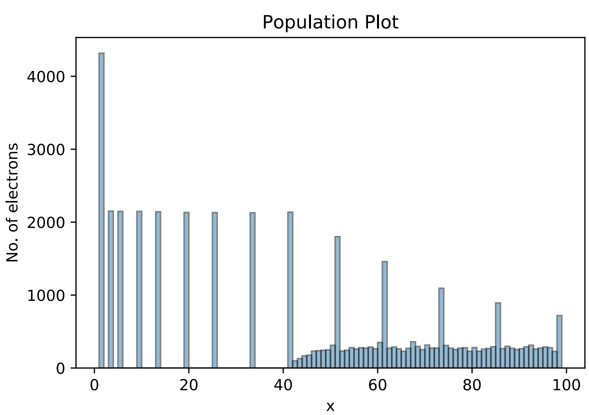
```
n = 100
M = 5
nk = 500
u0 = 7
p = 0.25
```

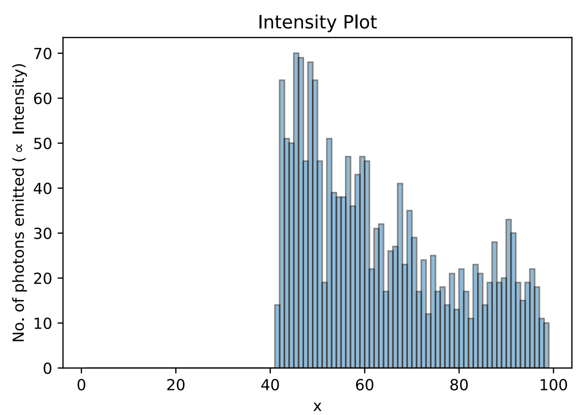




Now, we simulate using the following parameters (varying **u0** and keeping other parameters constant at default) -

```
n = 100
M = 5
nk = 500
u0 = 10
p = 0.25
```





6.2 Observation

- We see that as we **increase** u0 keeping p constant, the intensity decreases, and the **point** of maximum brightness shifts to the **right**.
- As u0 increases, the **threshold velocity** also increases. As a result, less number of electrons suffer a **collision** and produce light, shifting the curve to the right.

7 Conclusion

- As we **increase p**, keeping u0 constant, the bright regions become more clearly separated by **dark intervals**.
- This probably happens because the **bright regions** occur as a result of **collision** of electrons with atoms. As the **probability** of collision increases by a small amount, lesser electrons are able to make it past the point of **high brightness**.
- As we **increase u0**, keeping p constant, the intensity decreases and the **point** of maximum brightness also shifts to the **right**.
- This is because **higher** the threshold, more is the **velocity** required to be able to collide with an atom. Hence, electrons have to travel a **higher distance** each time u0 is increased.
- The **gaps** in electron density increases as we **increase** u0. This happens because the **point** of collision is shifting towards the **right** as we increase u0 and electrons move to higher distances.
- The phase plot obtained is **parabolic** in nature.
- Thus, we were able to conclude the various **properties** and **nature** of the intensity of light by varying the parameters.