EE 735 Assignment 4

Name-Rajdeep Sinha

Roll- 23M1133

Q1A) In this time independent problem, the diffusive transport of particles from point A to point B was found by solving the steady state equation as follows

$$D\frac{d^2n}{dx^2}=\frac{n}{\tau}$$

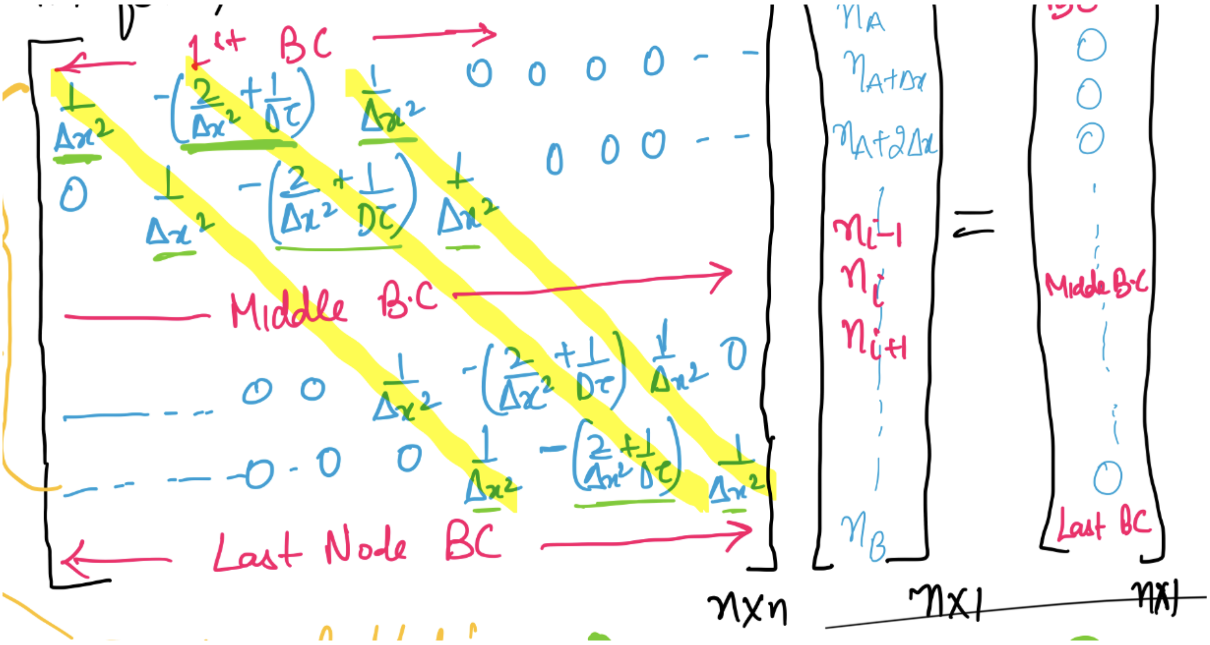
In doing so we have to discretize the X axis using a small enough step size so that the equation can be solved with minimal error. The exact number of grid points were found by using the round function and dividing the total length by the step size chosen. We can solve this differential equation numerically by using the Central difference formula for the ith node as follows

$$D * \left\{ \frac{n(i-1) - 2n(i) + n(i+1)}{\Delta x^2} \right\} = \frac{n(i)}{\tau}$$

We can rearrange the following equation to get the following for the 2nd to (n-1)th nodes. The 1st and nth can be found by applying the necessary boundary condition.

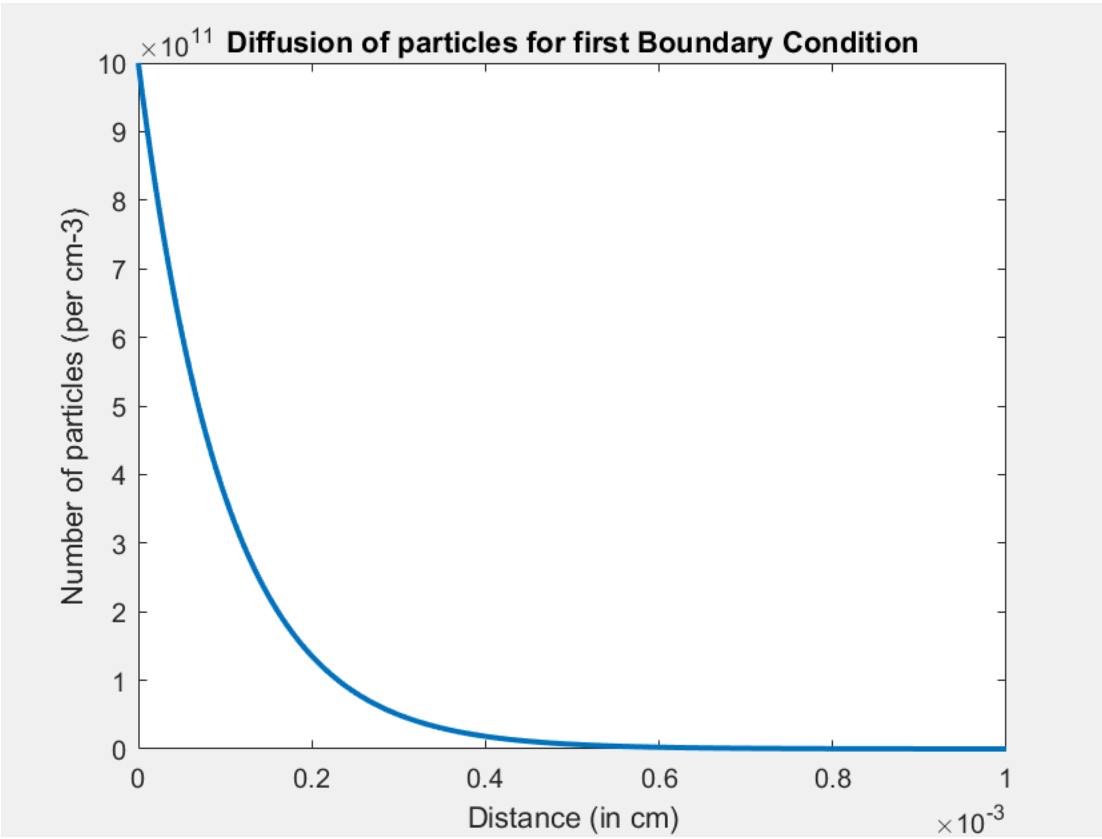
$$\left(\frac{1}{\Delta x^2}\right) n_{i-1} + \left(-\frac{2}{\Delta x^2} - \frac{1}{D\tau}\right) n_i + \left(\frac{1}{\Delta x^2}\right) n_{i+1} = 0$$

This can be finally put in a matric form of A*N=B which can be further solved to find N=B*(A^-1).The matrix form is as follows

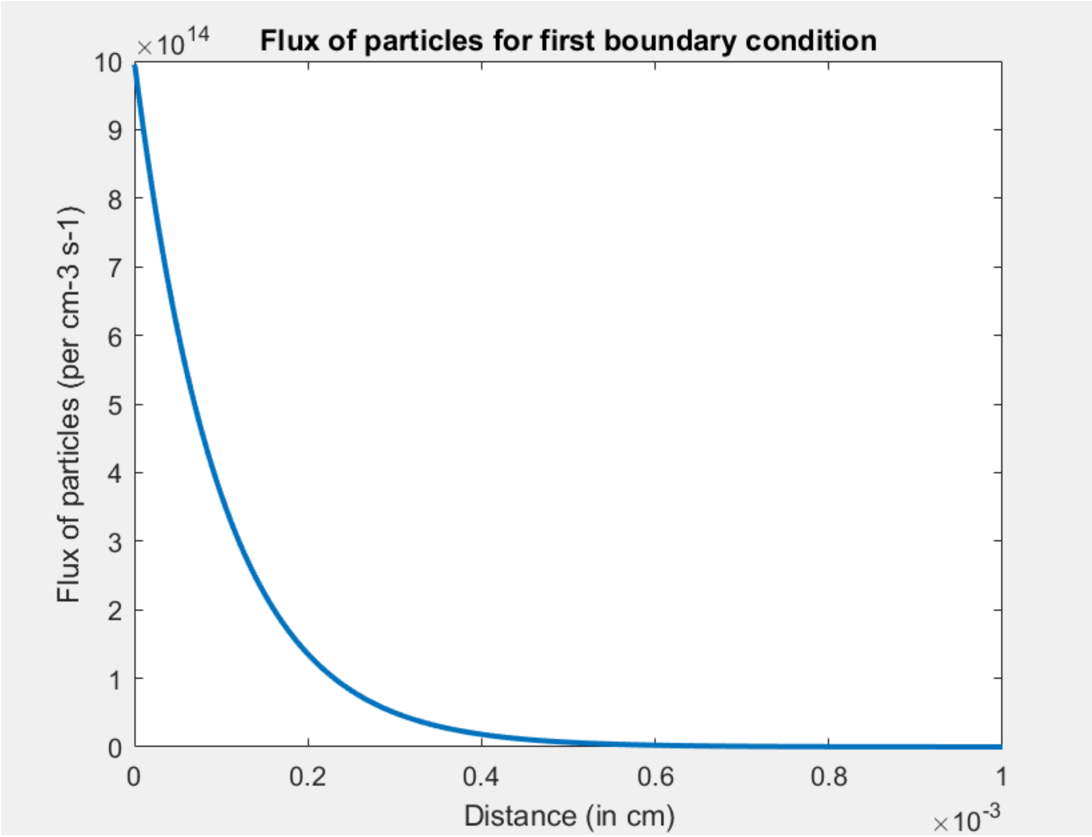


The first and last nodes can be used to apply boundary conditions by modifying A matrix and putting the (1,1) and (last,last) elements as 1.

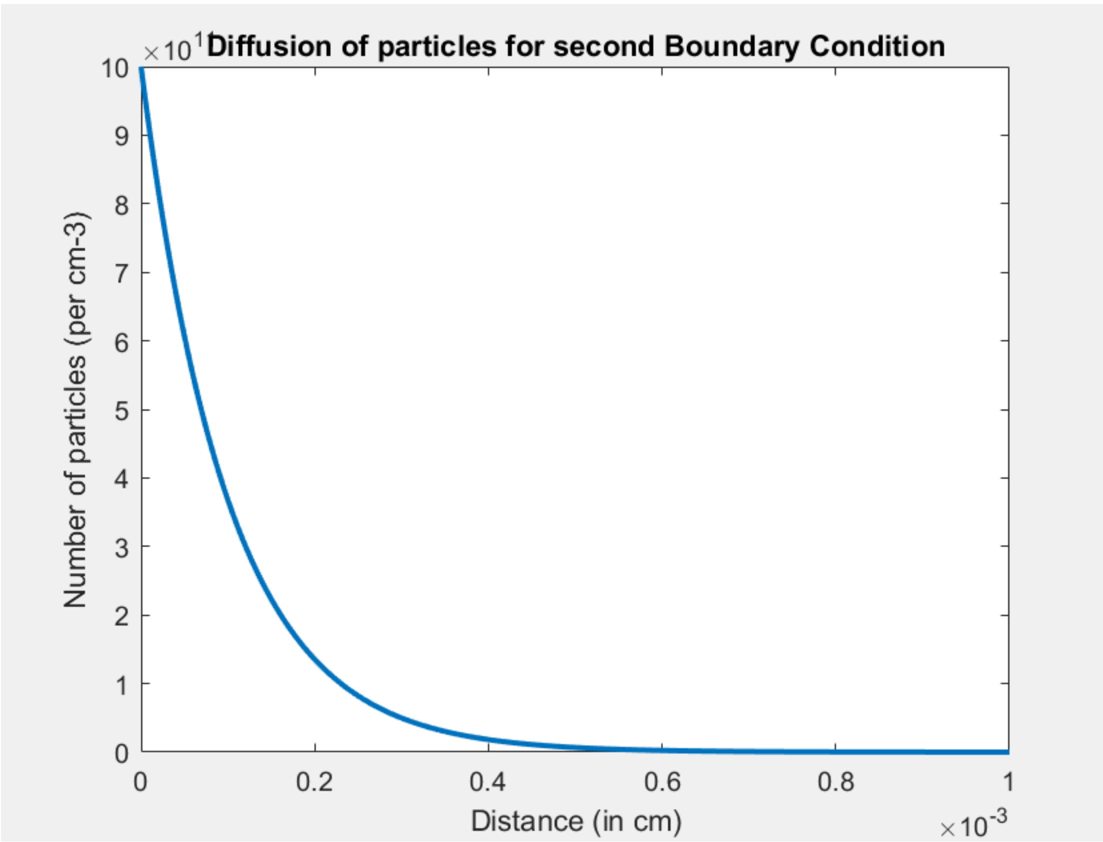
Solving this matrix we find the particle profile from A to B as follows

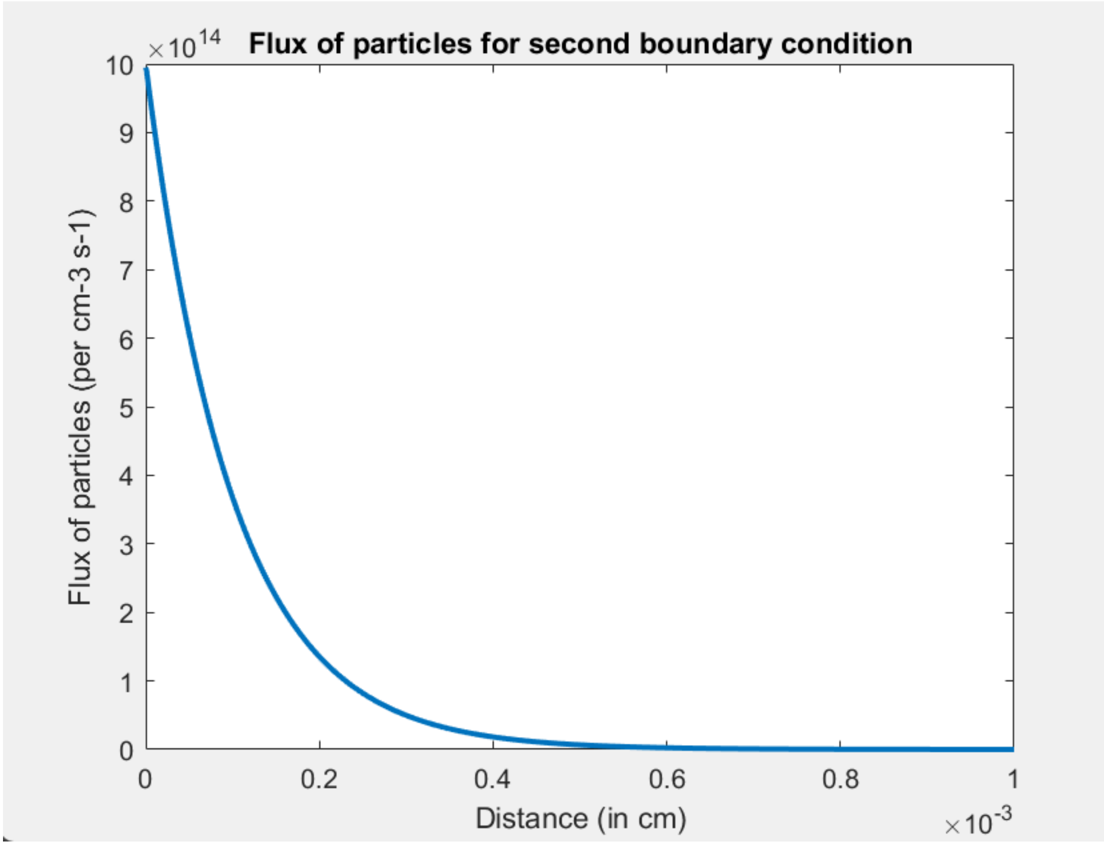


The particle flux was also found using Fick's first law J=-Ddn/dx. The plot is as follows

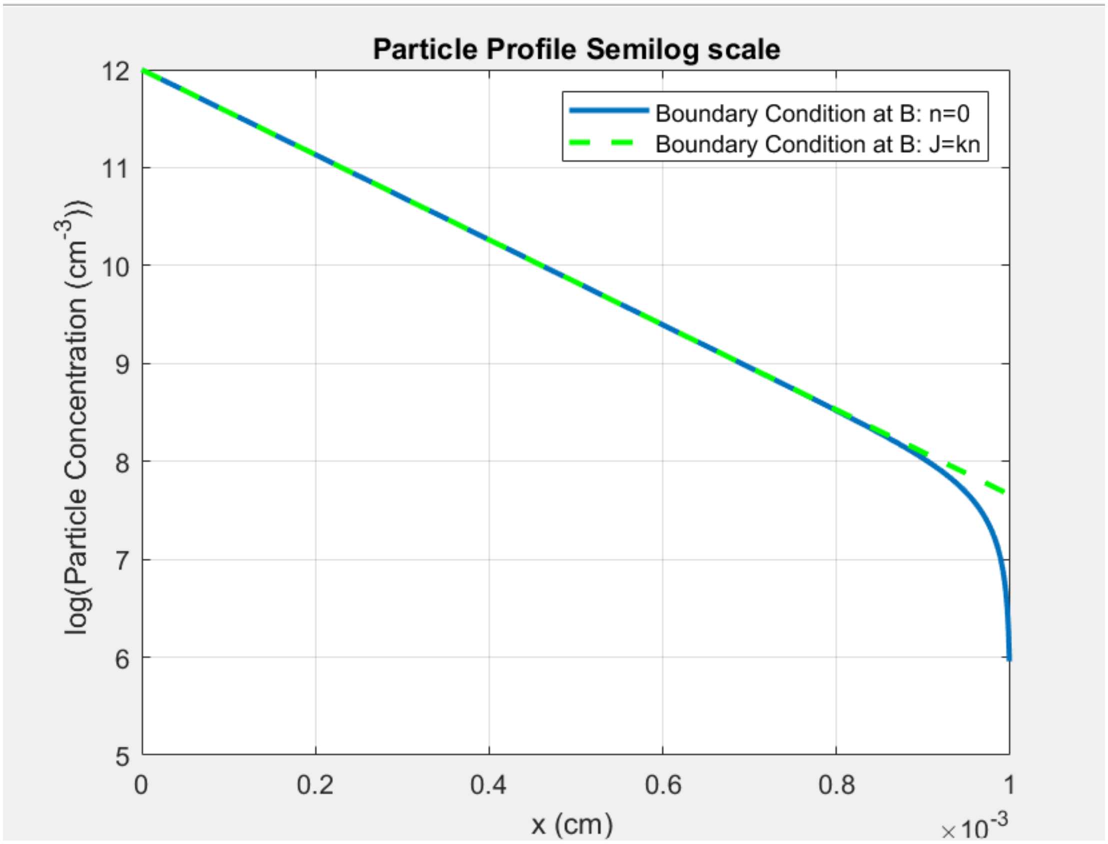


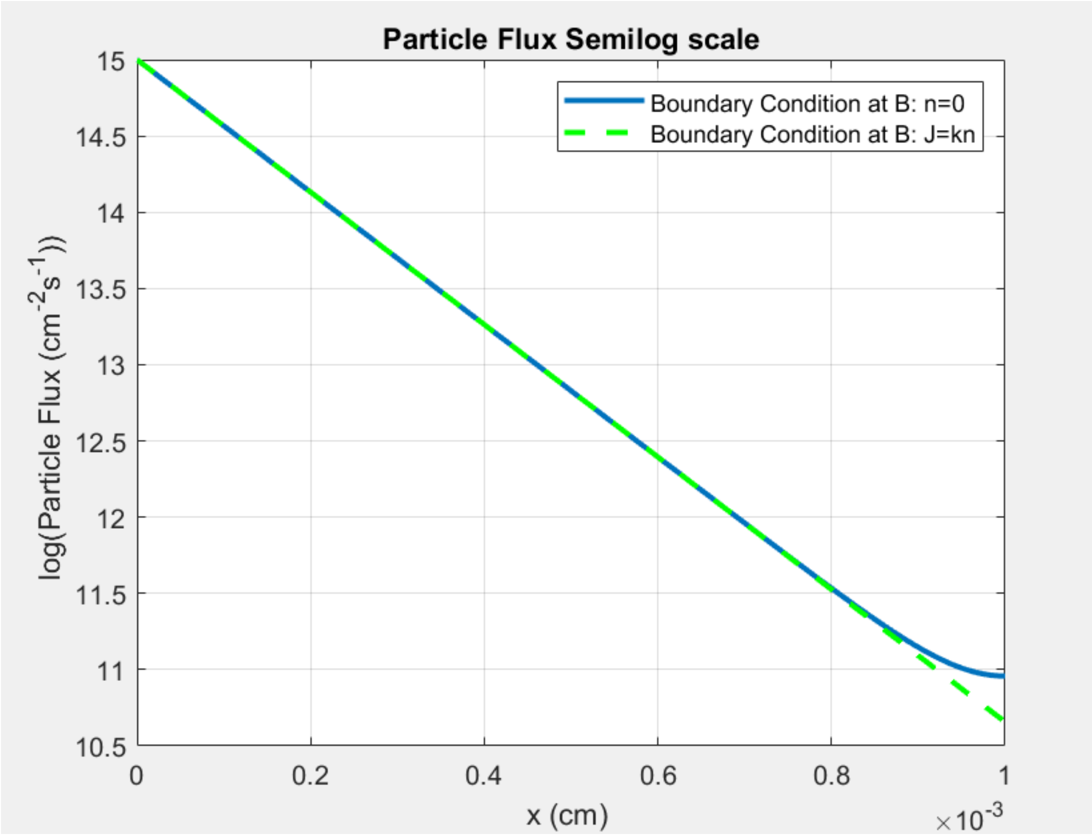
For the second boundary condition, the boundary condition at B was applied in B matrix and then N was calculated again. The plots came as follows.





The plots for particle profile and flux were further compared to find difference between the two boundary conditions in the semilog scale.





Q1B)

In this problem, the particle flux was introduced at 4.5um at a rate of 10^13 cm^2/sec. The particle densities at points A and B are held constant at n=0.The steady state equation was again solved using appropriate boundary conditions. The A matrix was modified at the flux point node as follows

```
A(fluxpt,fluxpt) = 1;
A(fluxpt,fluxpt-1) = -1;
A(fluxpt,fluxpt+1) = 0;
```

The B matrix was modified as follows

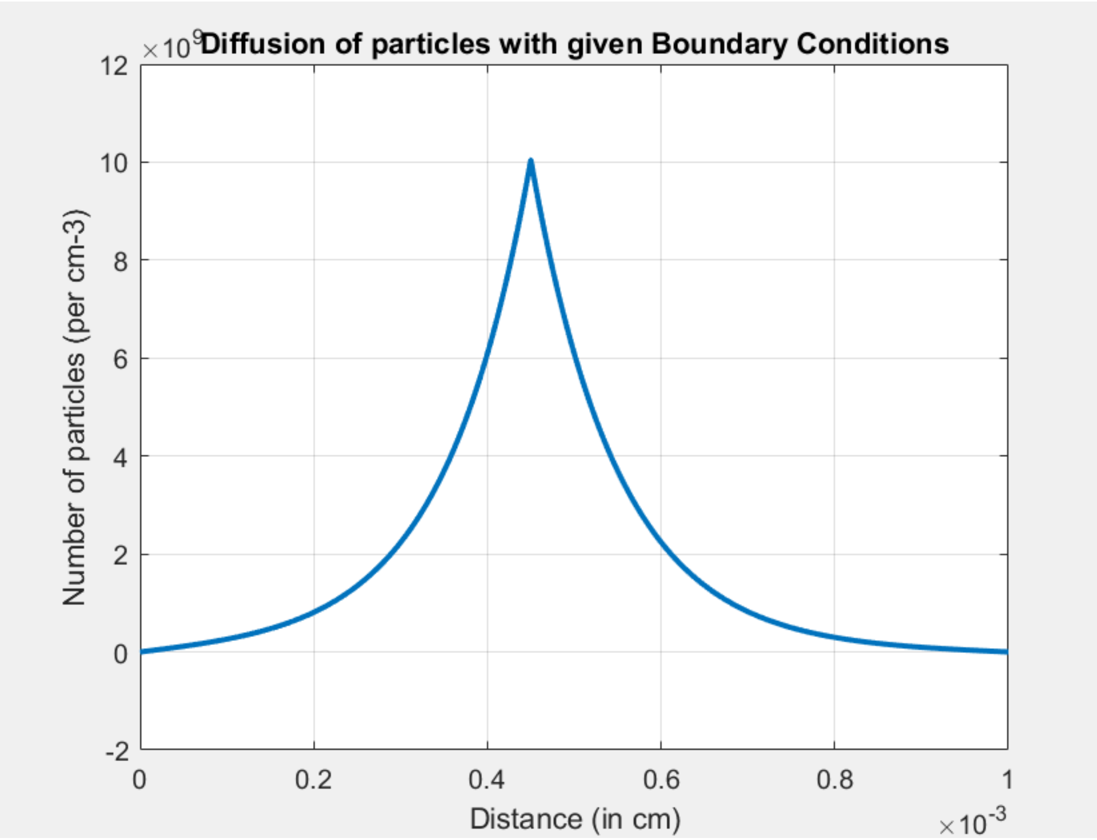
```
B(1,1) = 0;
B(gridno,1) = 0;
B(fluxpt,1) = (dx*J1)/D;
```

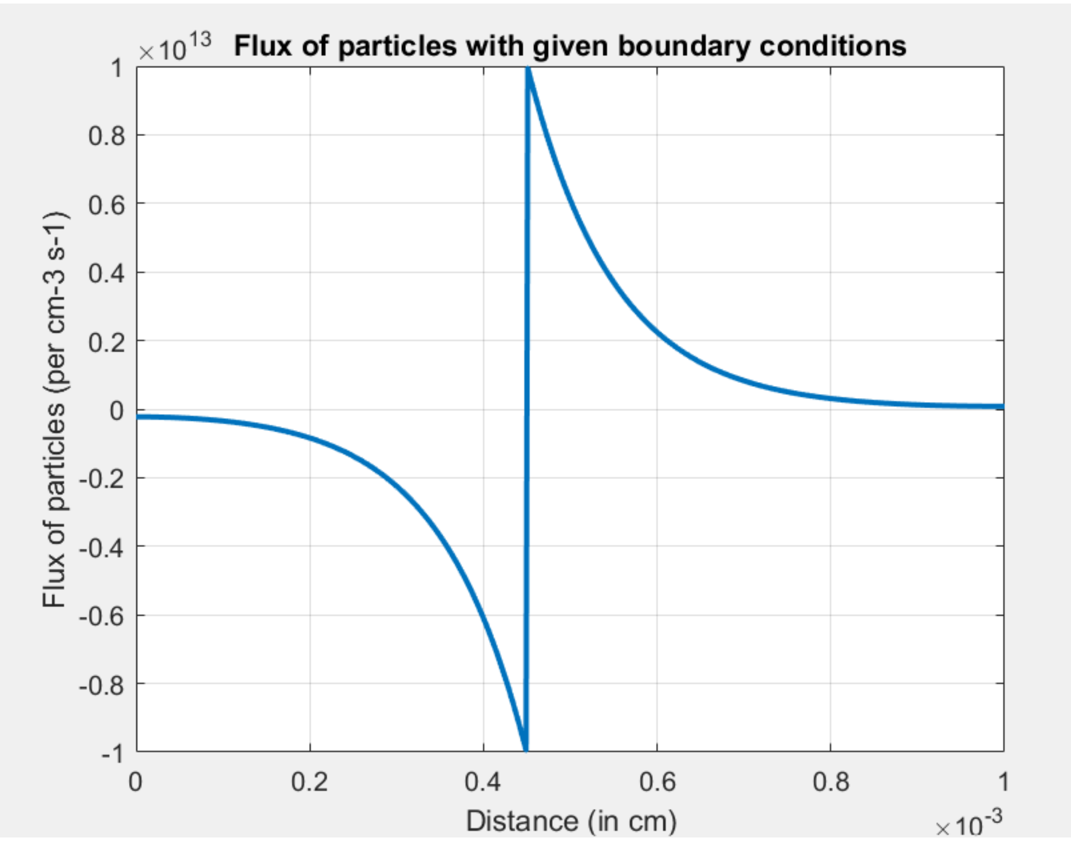
The flux at A and B were found from the code by considering the first and last node of J matrix. The results are as follows

The particle flux at A is -2.232728e+11 cm^-^2s^-^1

The particle flux at B is 8.212903e+10 cm^-^2s^-^1

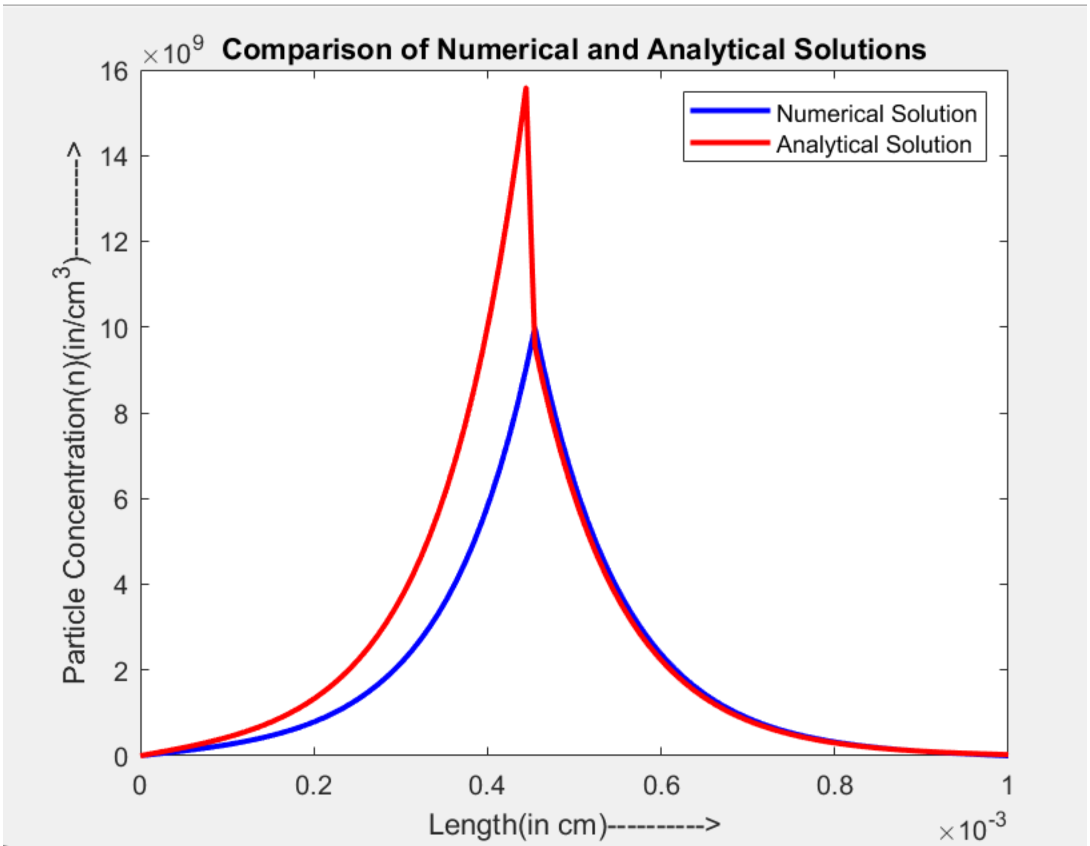
The plots obtained are given below





The flux comes negative due to the fact the flux decreases in the -ve X direction. Thus J=-Ddn/dx is negative in value.

Further the differential equation was solved analytically and compared with the numerical results. The plot came as follows.



Q1C)

The non steady state diffusion equation has to be solved to find the solution for this problem.

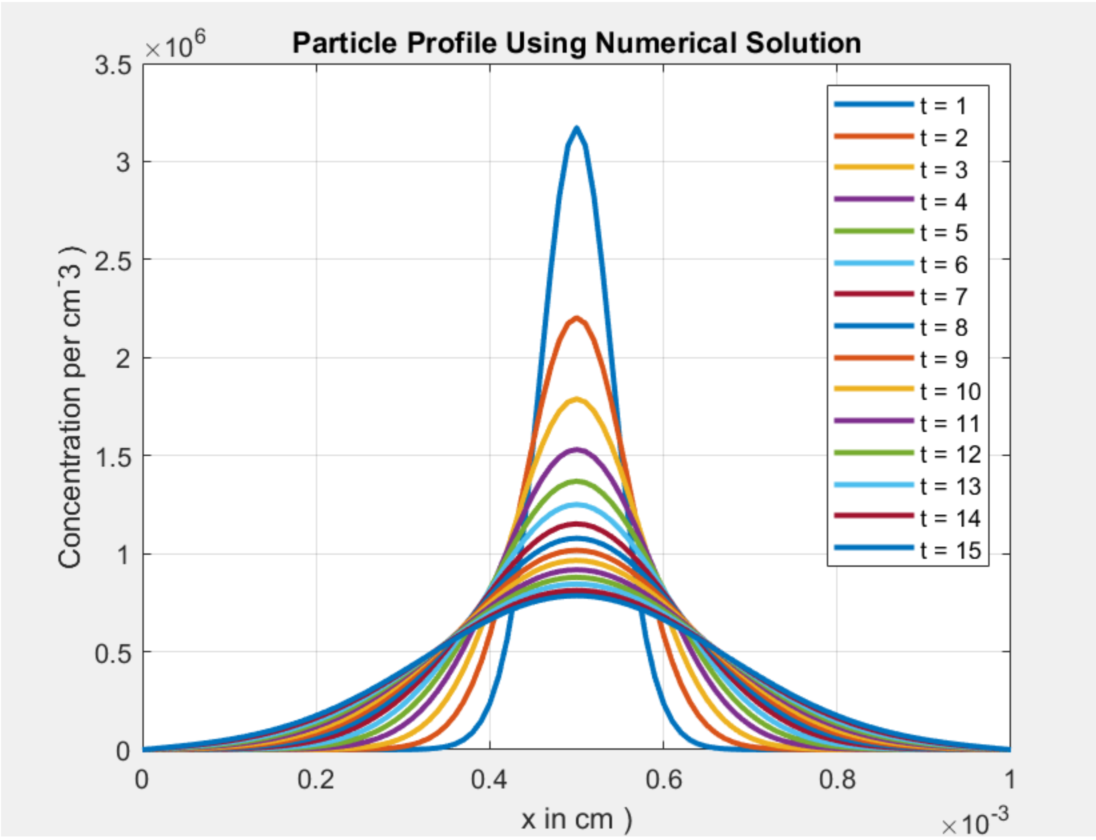
$$\frac{d^2n}{dx^2} = \frac{1}{D}\frac{dn}{dt}$$

Using central difference formula and using back substitution for time and rearranging, we get the following expression

$$\boxed{n_i^j = \left[c\left(n_{i-1}^j + n_{i+1}^j\right) + n_i^{j-1}\right] / (2c+1)}$$

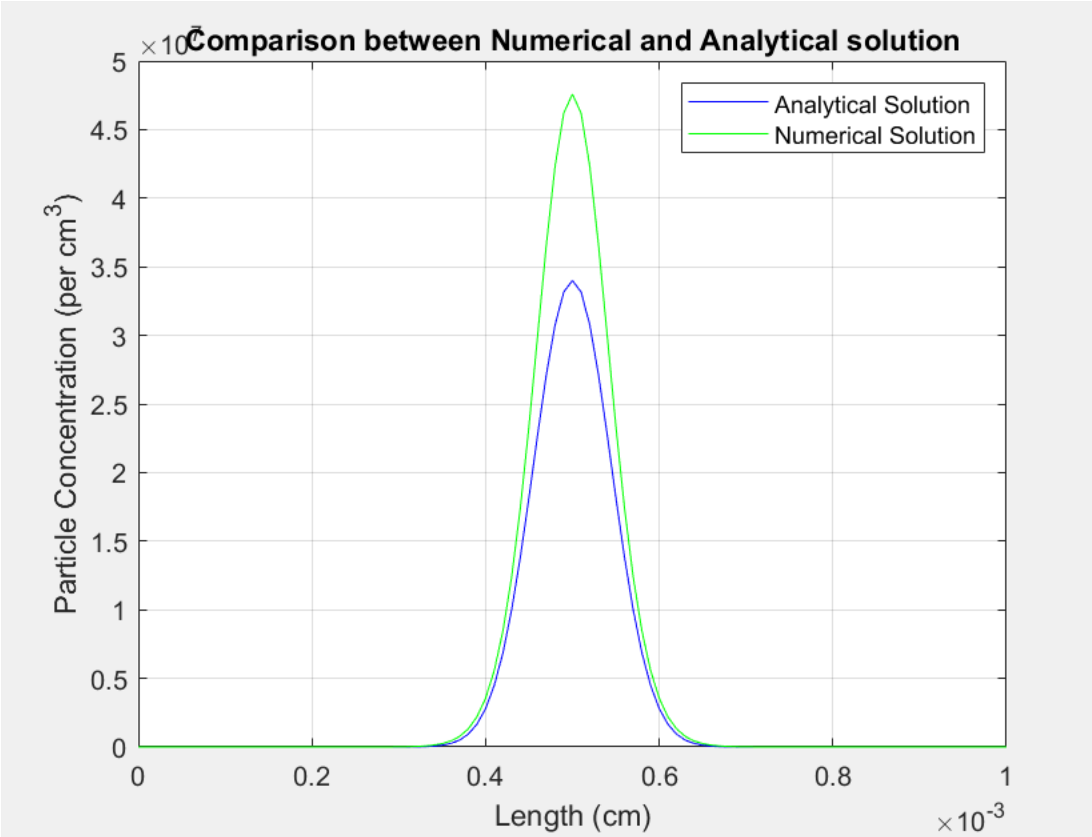
Where i is space node and j is time node.The value of C was taken to be 0.7 for the implicit case and 0.3 for the explicit case. The values of N matrix was taken for different time instants in different columns of the matrix.

The plot for particle profile from 10us to 150us was found as follows



The plot for the explicit profile was also found to be similar to implicit case.

Finally the analytical equations were solved and plotted to compare with the numerical solutions. The results are as given below



Significance of sqroot(Dt) term-

Diffusion often follows a Gaussian or normal distribution, where the concentration of the diffusing substance is highest at the point of release and decreases as you move away from it. The root Dt term is related to the width of this distribution. A larger root Dt value indicates a wider distribution, meaning the particles have spread out more over time.The square root of Dt provides a measure of the average distance a particle or substance will have diffused over a given time period. It represents the standard deviation of the distribution of particles as they spread out in space.

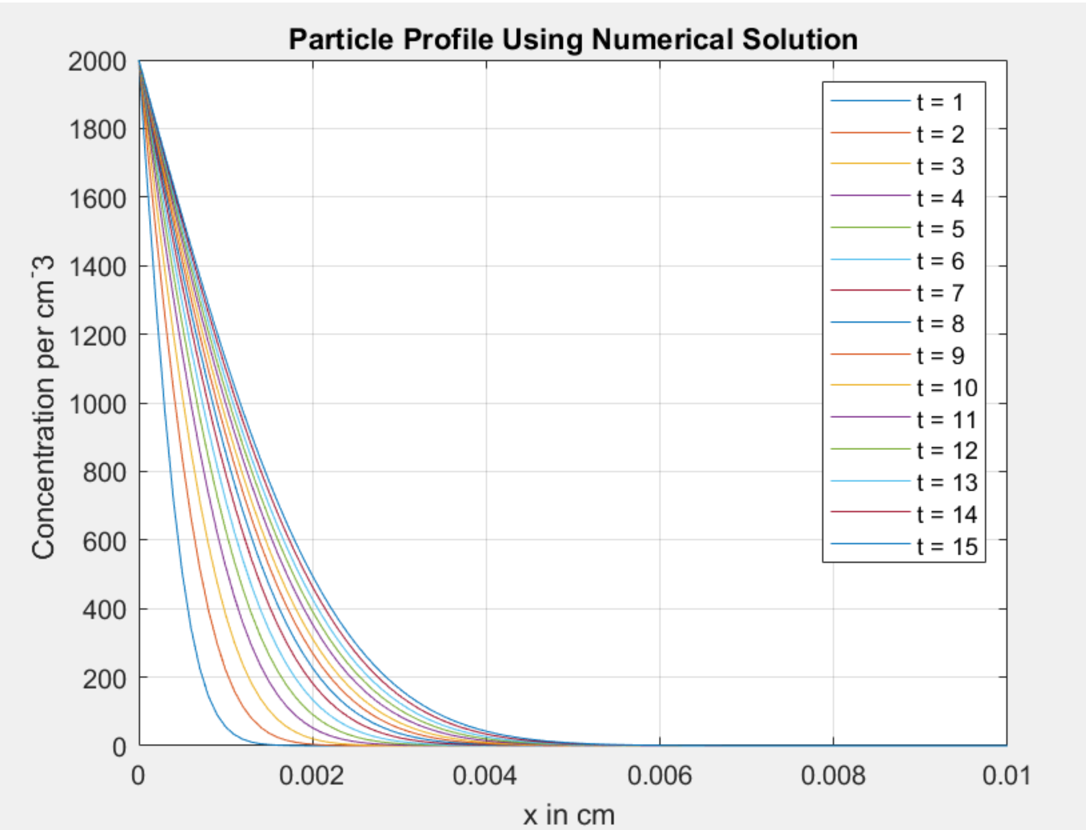
Q2) This problem is also similar to 1c. We have to plot the spatial and temporal evolution of the particle density profile under the boundary condition that n(x=0, t) =2000.The non steady state differential equation has to be solved again as before and the N values for every time instant has to be kept in different columns of the N matrix. The time values are taken from 1us to 15us at an interval of 1us.We create vectors ‘dist’ representing spatial positions and ‘T’ representing time intervals.Matrix N is initialized to store particle densities at various spatial positions and time points. The matrix dimensions are determined by the lengths of dist and T. The first row of N is set to the initial density N0. The code uses nested loops to update the particle densities at each spatial position and time point based on the FTCS scheme. It iterates Ni times, updating the values of N at each iteration.

The outer loop (for z = 1:Ni) controls the number of iterations.

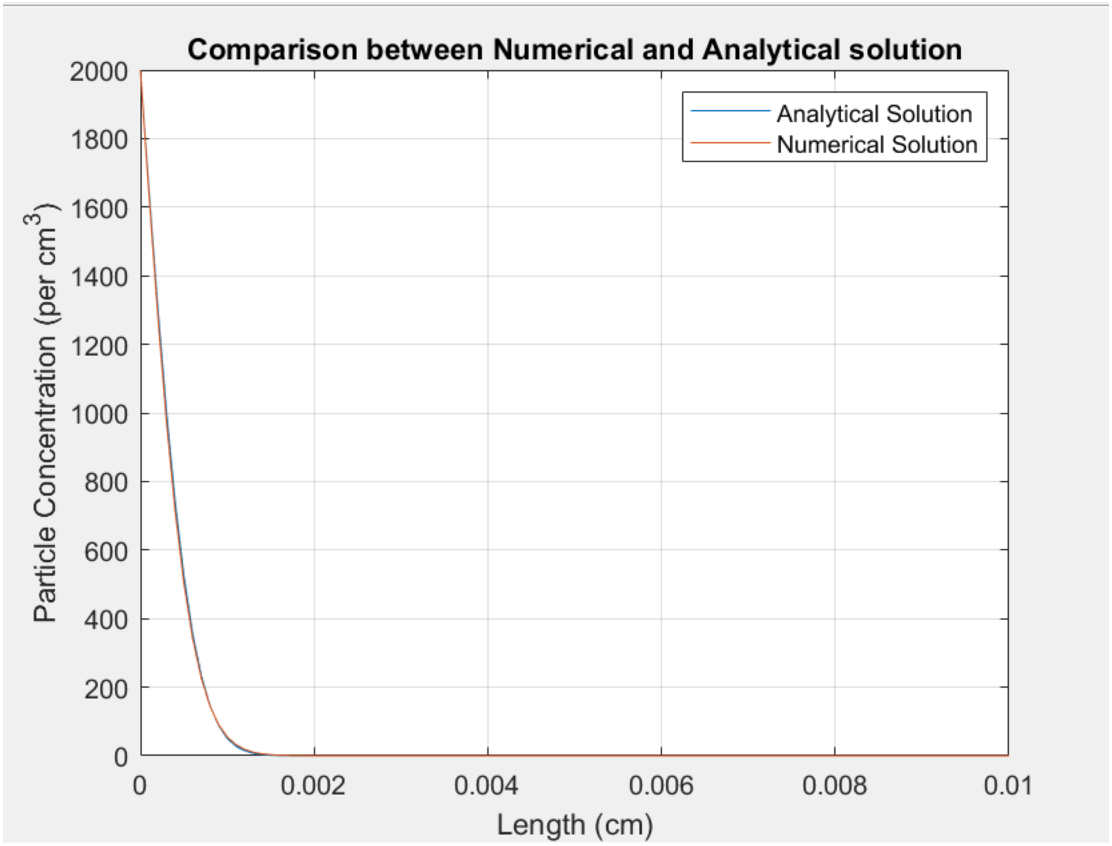
The middle loop (for t = 2:T1) iterates over time points.

The innermost loop (for i = 2:dist1-1) iterates over spatial positions, excluding the boundaries.

The update equation inside the innermost loop calculates the new particle density at a given spatial position and time point based on neighboring values and the previous time point's value. The plot for particle density was obtained as follows



The analytical problem was solved using appropriate functions and was plotted as follows.



Q3)

In this problem we have to understand how the concentration of minority carriers (holes) on the N-side changes with time and position within the semiconductor when we switch from forward bias to reverse bias. The diffusion of minority carriers (holes) is governed by a partial differential equation represented as p(x,t) = p0 + Δp(x,t). This equation involves the concentration of holes (p(x,t)), excess concentration of holes (Δp(x,t)), and several parameters.Matrices for hole concentration (p1), delta hole concentration (delp1), and delta charge (delQ) are initialized. The equilibrium hole concentration is set in p1.A boundary condition is applied at the depletion edge, ensuring that the slope of p(x,t) at x=wn is constant.The code iteratively solves a finite difference equation for Δp(x,t) using time and spatial discretization. It calculates how the hole concentration changes over time and position within the diode. The excess charge density is calculated at each time interval, providing insight into the distribution of holes within the diode. The storage time (tstorage) at which the excess charge density becomes zero, is calculated accordingly. Finally we generate plots of hole concentration over time, showing how it evolves within the diode.

We initialize various constants and parameters in the code initially. These parameters include physical constants like charge (q), donor and acceptor concentrations (N_D and N_A), diode length (L), hole mobility (up), hole lifetime (taup), intrinsic carrier concentration (Ni), current density (Jr), grid size (gridsize), and the number of time instants (instants).Wd is calculated as half of the diode length (L / 2).Po represents the equilibrium hole concentration without applied voltage and is calculated based on Ni, N_A, and N_D. esi is the product of the dielectric constant (er) and the permittivity of free space (eo). gridpoints defines a grid of positions in the N-side of the diode from 0 to 1e-3 cm. The built-in voltage (Vbi) based on the thermal voltage (Vt) and the donor and acceptor concentrations is calculated. The depletion width (Wdep) in the N-side of the diode is also calculated.We calculate the hole diffusion coefficient (Dp) using the hole mobility (up) and thermal voltage (Vt). We also calculate the hole diffusion length (Ldiff) based on the product of Dp and taup. gridstep defines the grid step size and xpositions is a vector of positions along the diode's N-side.

Then the code computes the analytical storage time (ts) based on hole lifetime (taup) and current density (Jr). The time step (delt) is calculated for the simulation. We initialize matrices (p1, delp1, and delQ) to store hole concentration, delta hole concentration, and delta charge. We also initialize the hole concentration matrix p1 with the equilibrium hole concentration Po.Boundary conditions are applied at the depletion edge (delp1(2, 1)) based on the grid step, current density, and physical constants.Delta voltage is initialized in the loop, considering exponential decay with distance. K1 is initialized as a constant value, which will be used in the finite difference equation to update the hole concentration.K2 is another constant that is used to apply a boundary condition at the depletion edge.Then we perform time iterations to simulate the evolution of hole concentration within the semiconductor diode's n-region. The outer loop (for j = 2:instants) iterates over time, starting from the second time instant (time step j = 2), and continues until the specified number of time instants (instants).Within each time instant, there's an inner loop (for iter = 1:100). This inner loop is repeated multiple times (up to 100 iterations) to ensure numerical stability and accuracy in solving the finite difference equation.The next nested loop (for i = 2:gridsize-1) iterates over spatial grid points within the diode's n-region, excluding the boundary points. Inside the innermost loop, the code calculates the new value of delp1 (the excess concentration of holes) at each spatial grid point and time instant based on the finite difference equation. The new value of delp1 is determined based on the values of delp1 at neighboring grid points and the previous time instant (j-1). The result is stored in the matrix delp1(i, j). After updating the values of delp1 for all spatial grid points, a boundary condition is applied at the depletion edge (x = 0). This condition ensures that the slope of the hole concentration (delp1) at this boundary remains constant.The boundary condition is implemented by setting delp1(1, j) to be equal to delp1(2, j) - K2. This step-by-step process effectively updates the hole concentration over time and space, taking into account diffusion and boundary conditions, allowing for the simulation of hole behavior within the semiconductor diode's n-region.Finally we plot the Evolution of hole concentrations on the N-side of diode.