PHYS338:Computational Physics
HW9

Student Name: Hamza AlHasan-1181636

Submission Date: Dec 22, 2020

Consider this physical partial differential equation with boundary conditions:

$$\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} - k^2 V = 0 \tag{1}$$

$$V(x,0) = 1 \tag{2}$$

$$V(x,10) = 1 \tag{3}$$

$$V(0,y) = 1 \tag{4}$$

$$V(10,y) = 0 \tag{5}$$

Where $V(x,y)$ is 2D-voltage function, and k is inverse screening length. $k = \frac{1}{\lambda}$ where λ is screening length.

We want to solve this partial equation for different values of k, for $k = \{0, 0.1, 0.01, 0.001\}$.

A Wolfram Mathematica code can be used to solve such a problem. In Mathematica, there is function called: **NDSolve**[ ] which can be used to solve partial equations numerically. This code is shown in figure(1).

```
In[81]:= k = 0.0

Out[81]= 0.

In[85]:= res = NDSolve[{Laplacian[V[x, y], {x, y}] - k^2 V[x, y] == 0,
                V[x, 0] == 1, V[x, 10] == 1, V[0, y] == 1, V[10, y] == 0},
              V, {x, 0, 10}, {y, 0, 10}]

Out[85]= {{V → InterpolatingFunction[ Domain: {{0., 10.}, {0., 10.}}
                                      Output: scalar ]}}
```

Figure 1: Wolfram Mathematica Code

Result:

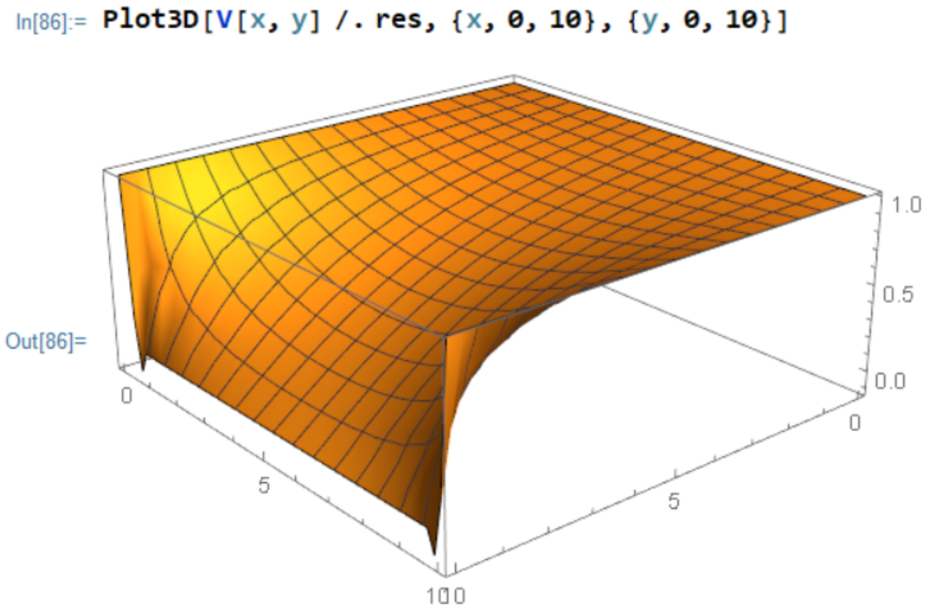

Figure 2: 3D Plot for partial equation(1)

In fact, there is no noticeable difference in plotting the solution for the four different values for k. Figure(2) and figure(3) are actually represent the solution for the four different values of k.

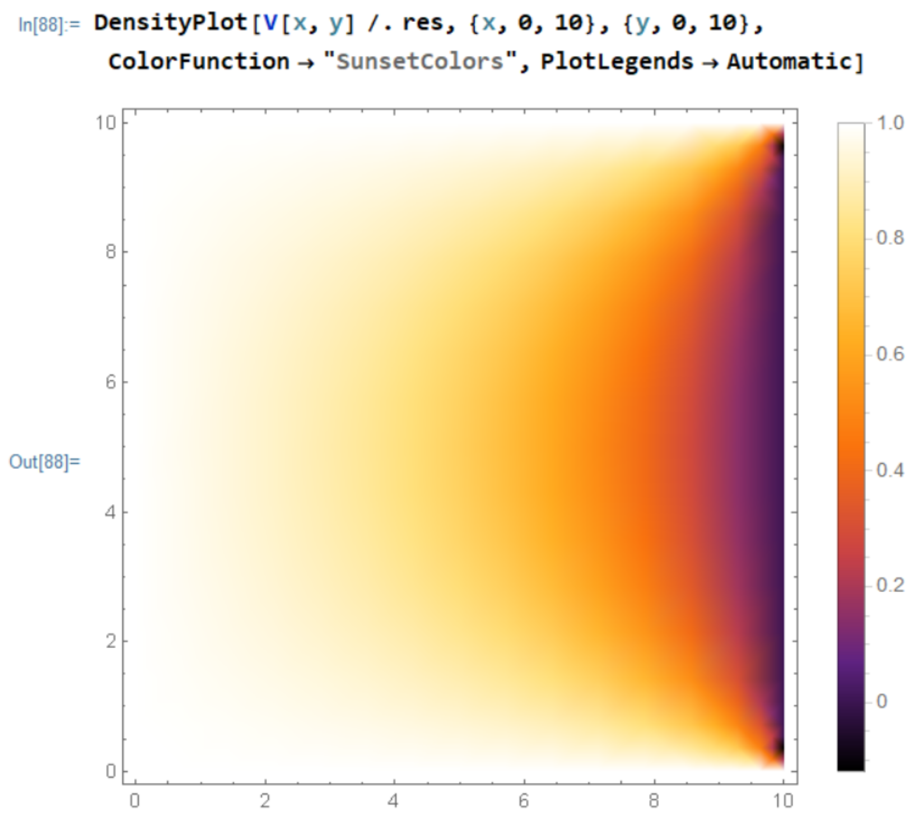

Figure 3: Density Plot Code for equation(1)