

DOING PHYSICS WITH MATLAB

Ian Cooper

matlabvisualphysics@gmail.com

A COMPUTATIONAL APPROACH TO ELECTROMAGNETIC THEORY

ELECTRIC FIELD AND ELECTRIC POTENTIAL: POISSON'S and LAPLACE'S EQUATIONS

DOWNLOAD DIRECTORIES FOR MATLAB SCRIPTS

[Google drive](#)

[GitHub](#)

cemLaplace01.m

Solution of the [2D] Laplace's equation using a relaxation method. Can specify the number of iterations or set the tolerance by modifying the program. The Script can be used to investigate the convergence of the relaxation method. Graphical output of the potential and electric field.

cemLaplace02.m

Solution of the [2D] Laplace's equation using a relaxation method with the termination of the iterative process determined by the tolerance set in the INPUT section of the code. Graphical output of the potential and electric field. You may need to modify the code for the calculation and the graphical outputs for different boundary conditions and inputs.

cemLaplace03.m

Solution of the [1D] Laplace's equation using a relaxation method with the termination of the iterative process determined by the tolerance set in the INPUT section of the code. A comparison can be made between the numerical predictions for the variation in potential with the exact analytical results.

cemLaplace04.m

Solution of the [2D] Laplace's equation using a relaxation method for two infinite concentric squares

The equations of Poisson and Laplace for electrostatics

Maxwell's derivation of Maxwell's equations marked an incredible achievement where a set of equations can completely describe charges and electric current. Gauss's law is one of these equations and it describes electric fields in a vacuum with charge density ρ

$$(1) \quad \nabla \cdot \vec{E} = \frac{\rho}{\epsilon_0}$$

The electric field $\vec{E}(\vec{r})$ for a given a stationary charge distribution $\rho(\vec{r})$ can be calculated from

$$(2) \quad \vec{E}(\vec{r}) = \frac{1}{4\pi\epsilon} \iiint_V \left(\frac{\rho(\vec{r}') dV'}{|\vec{r} - \vec{r}'|^2} \right) \hat{r}$$

The electric field $\vec{E}(\vec{r})$ is a vector quantity, therefore, it is often easier to calculate the scalar quantity known as the electrostatic potential $\phi(\vec{r})$ from equation 3 rather than calculate the electric field using equation 2.

$$(3) \quad \phi(\vec{r}) = \frac{1}{4\pi\epsilon} \iiint_V \left(\frac{\rho(\vec{r}') dV'}{|\vec{r} - \vec{r}'|} \right)$$

The electric field is the gradient of the potential

$$(4) \quad \vec{E} = -\nabla\phi$$

The electrostatic potential ϕ can only be evaluated analytically for the simplest charge configurations. In addition, in many electrostatic problems, conductors are involved and the charge distribution $\rho(\vec{r})$ is not known in advance (only the total charge or potential on each conductor is known).

A better approach to determine the electrostatic potential ϕ is to start with **Poisson's equation**

$$(5) \quad \nabla^2\phi = -\frac{\rho}{\epsilon} \quad \nabla\cdot\vec{E} = \frac{\rho}{\epsilon_0} \quad \nabla\cdot\nabla\phi = \nabla^2\phi = -\frac{\rho}{\epsilon_0}$$

and together with the applied **boundary conditions** it is equivalent to equation 3.

Very often we only want to determine the potential in a region where $\rho = 0$. In this region Poisson's equation reduces to **Laplace's equation**

$$(6) \quad \nabla^2\phi = 0$$

Laplace's equation in Cartesian coordinates is written as

$$(7) \quad \nabla^2 \phi = \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0$$

This is a fundamental equation of electrostatics. It is also an important equation in many branches of physics such as magnetism, gravitation, thermal physics, fluids, and soap bubbles.

Numerical solutions of Poisson's equation and Laplace's equation

We will concentrate only on numerical solutions of Poisson's equation and Laplace's equation. As an introduction, we will only consider [1D] and [2D] cases. The methods discussed can easily be extended to [3D] situations.

The central difference approximation to the second derivative of the function $y(x)$ is

$$(8) \quad \left. \frac{d^2 y(x)}{dx^2} \right|_x \approx \frac{y(x + \Delta x) - 2y(x) + y(x - \Delta x)}{\Delta x^2}$$

The [2D] space is divided into a grid of $N_x \times N_y$ points with grid spacing $h_x \equiv \Delta x$ in the X direction and $h_y \equiv \Delta y$ in the Y direction.

For computational purposes, it is best to express all quantities defined at the grid positions in terms of grid indices n_x and n_y where

$$n_x = 1, 2, 3, \dots, N_x \quad \text{and} \quad n_y = 1, 2, 3, \dots, N_y$$

Using the central difference approximation of equation 8, we can approximate the value of the potential $\phi(n_x, n_y)$ in terms of the potentials at the surrounding grid points from Poisson's equation

$$\frac{\phi(n_x + 1, n_y) - 2\phi(n_x, n_y) + \phi(n_x - 1, n_y)}{h_x^2} + \frac{\phi(n_x, n_y + 1) - 2\phi(n_x, n_y) + \phi(n_x, n_y - 1)}{h_y^2} = -\frac{\rho}{\epsilon_0}$$

Rearranging this expression, we get

(9)

$$\phi(n_x, n_y) = \left(\frac{h_x^2}{2(h_x^2 + h_y^2)} \right) (\phi(n_x, n_y + 1) + \phi(n_x, n_y - 1)) + \left(\frac{h_y^2}{2(h_x^2 + h_y^2)} \right) (\phi(n_x + 1, n_y) + \phi(n_x - 1, n_y)) + \frac{h_x^2 h_y^2}{2(h_x^2 + h_y^2)} \frac{\rho}{\epsilon_0}$$

The potential at each point is a weighted average of the values of the potential at the surrounding points. So, if we start with known fixed values of the potential on the boundaries, we can repeatedly compute the interior values until we get a convergence in their

values. This is known as a **relaxation method** to solve **boundary value problems**.

When using the relaxation method, we mostly specified the boundary conditions and set all values of the potential to zero (or best guess value) at each interior grid. We then repeatedly update all interior values of the potential by the average of the latest neighbouring grid point values. We also have to test the convergence for the values for the potential. There are many ways in which this can be done. In the Scripts for solving Poisson's equation or Laplace's equation, we will test the convergence by continuing the iterative process while the difference in the sums of the square of the potential at each grid point for the current and previous iterations is greater than specified tolerance.

Section of the Script for the [1D] Laplace's equation showing the iterative process

```
tol = 10;
% dSum Difference in sum of squares / n number of
iterations
dSum = 100; n = 0;
while dSum > tol
    sum1 = sum(sum(V.^2));
    for nx = 2: Nx-1
        V(nx) = 0.5*(V(nx+1) + V(nx-1));
    end
    sum2 = sum(sum(V.^2));
    dSum = abs(sum2 - sum1);
    n = n+1;
end
```

From now on the electrostatic potential is given by the letter V rather than the Greek letter ϕ .

Example 1 **cemLaplace03.m**

Solve the [1D] Laplace's equation for the potential $V(x)$ with the boundary conditions

$$x = 0 \text{ m} \quad V = 100 \text{ V} \quad \text{and} \quad x = 5.0 \text{ m} \quad V = 0 \text{ V}$$

and calculate the electric field in this region



Download the mscript **cemLaplace03.m**. Review the code so that you understand each line and how Laplace's equation was solved using the relaxation method.

The potential is given by

$$V(n_x) = \frac{V(n_x + 1) + V(n_x - 1)}{2}$$

where $n_x = 2, 3, 4, \dots, N_x - 1$ $V(1) = 100$ $V(N_x) = 0$

The exact solution of Laplace's equation with the given boundary conditions is

$$V(x) = -20x + 100$$

The Script **cemLaplace03.m** is used to solve this problem. The exact solution can be compared with the solution using the relaxation method as shown in figure 1. Figure 1 clearly shows that the smaller

the value of the tolerance, the better the agreement between the exact solution and the solution using the relaxation method. For a tolerance of $\text{tol} = 1$, the difference between the exact solution and approximate solution is about 0.1%. You always need to be careful in using approximation methods to ensure that the computed values are accurate. You always should check that you have used a smaller enough grid spacing and/or used a smaller enough tolerance.

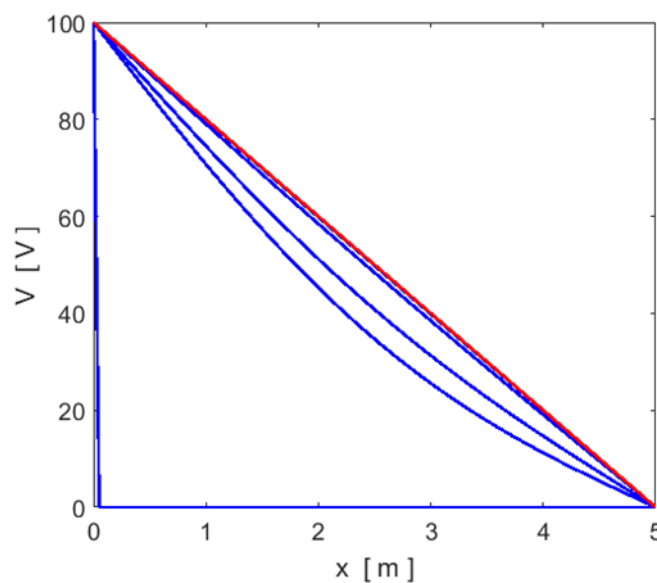


Fig. 1. Plot showing the exact solution (red) and solutions using the relaxation method (blue) with tolerances of 100, 75, 50, 10 and 1. The number of iterations for each tolerance are 0, 1415, 1934, 3692 and 6049 respectively. The initial values assigned to the potentials at each grid point is shown when $\text{tol} = 100$ (number of iterations = 0). There is very good agreement between the exact solution and the relaxation method solution when the tol is less than 1. Number of grid points is $N_x = 101$. The computation time is less than 1 second. **cemLaplace03.m**

The potential at any interior point is a weighted average of the potential at surrounding points. As a consequence, there can be no interior minima or maxima: the extreme values of the potential must occur only at the boundaries. In this [1D] example, the value at an interior grid point is the average of the two values of the potential at the two adjacent grid points, therefore, its value can't be greater than or less than the value at either adjacent grid point. This fact is true for the [2D] and [3D] solutions of Laplace's equation.

The electric field is given by $\vec{E} = -\nabla V$. The electric field is uniform and points in the +X direction and its exact value is 20 V.m^{-1} . The numerical estimate of the electric field is calculated from the potential using the function gradient

$$E = -\text{gradient}(V, hx);$$

Figure 2 shows the potential and electric field calculated by the relaxation method to solve Laplace's equation.

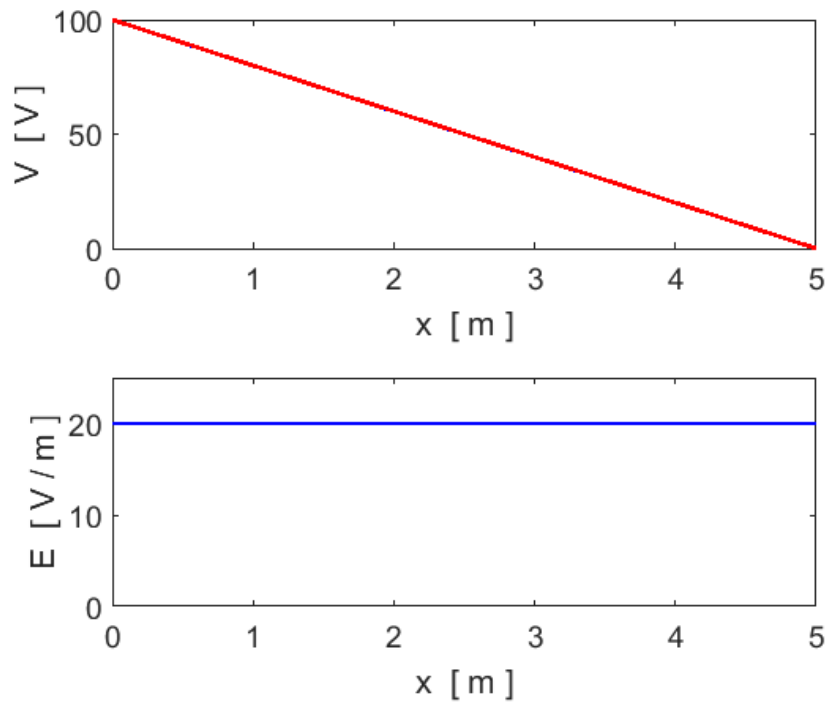


Fig.2. The potential and electric field calculated by solving Laplace's equation by the relaxation method.

cemLaplace03.m



The solution to Laplace's equation in a volume is uniquely determined if the potential V is specified on the boundary surface S of the volume.

Example 2 `cemLaplace02.m`

There are two parallel semi-infinite metal plates in the XZ plane. One plate is located at $y = -1$ m and extends from $x = 0$ m to $x = 4$ m and the other plate is located at $y = +1$ m and extends from $x = 0$ m to $x = 4$ m. At $x = 0$ m, an insulated strip connects the two semi-infinite plates and is held at a constant potential 100 V. Find the potential and the electric field in the region bounded by the plates.



The configuration is independent of z , so this reduces to a [2D] problem. We can find the potential in the interior region by solving Laplace's equation. The region of interest is divided into a rectangular grid of $N_x \times N_y$ grid points. The potential at all grid points is set to 0. Then the boundary condition, $V = 100$ V is applied to all grid points with $y = 0$. Then the iterative process updates the potential at all interior grid points until the values of the potential satisfy the tolerance condition. Download and inspect the mscript `cemLaplace02.m`. Make sure you understand each line of the code and how the Laplace's equation is solved using the relaxation method.

Figure 3 gives a plot of the potential and shows that the potential V has no local maximum or minima; all extremes are on the boundary. The solution of Laplace's equation is the most featureless function

possible, consistent with the boundary conditions: no hills, no valleys, just the smoothest surface. Figure 4 shows the variation in potential V as a function of x for different values of y . For each y value, the potential monotonically decreases from its maximum value at the boundary $x=0$ to zero at $x=4$ m. Figure 3 to 6 were produced using the mscript **cemLaplace02.m**

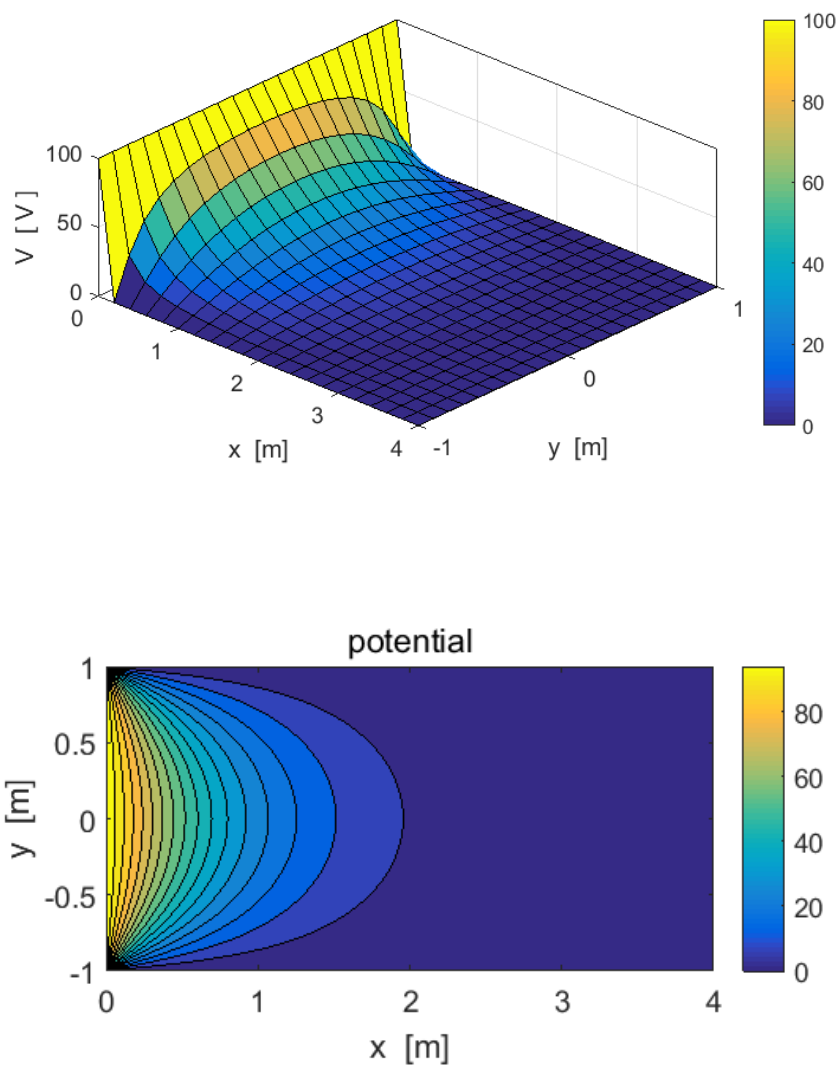


Fig. 3. The potential V in the region between the plates.

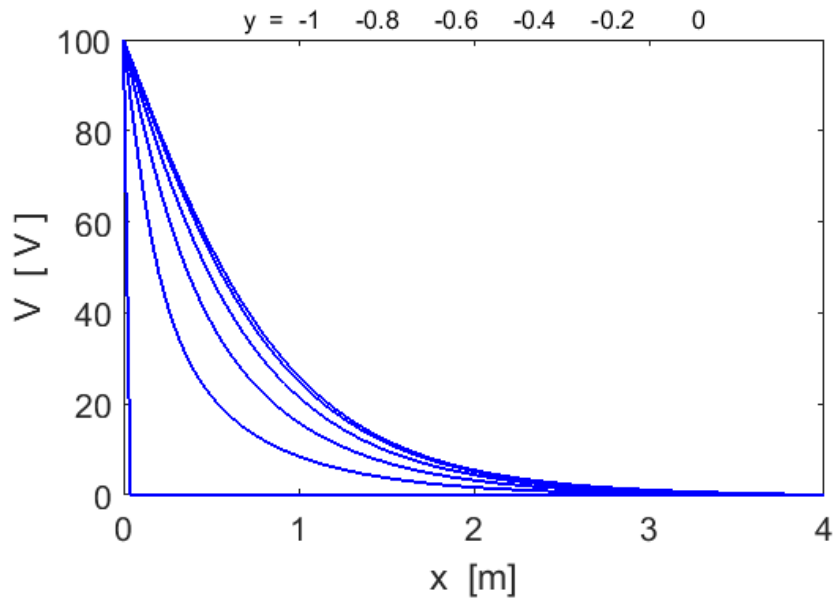


Fig. 4. The profiles of the potential V as functions of x for different y values.

The electric field \vec{E} is found by finding the gradient of the potential V

$$\vec{E} = -\nabla V$$

The Matlab code for the computing the components and magnitude of the electric field is

```
% electric field
[Exx, Eyy] = gradient(V, hx, hy);
Exx = -Exx; Eyy = -Eyy;
E = sqrt(Exx.^2 + Eyy.^2);
```

Figure 5 shows a [3D] plot of the magnitude of the electric field \vec{E} as a function of positions x and y . The electric field rapidly approaches zero from the two corner positions. The direction of the electric field is shown in the quiver plot in figure 6. Since the electric field drop to zero from its extreme values very rapidly away from the corner positions, it is necessary to use the zoom function in the Figure Window to clearly see the direction of the electric field.

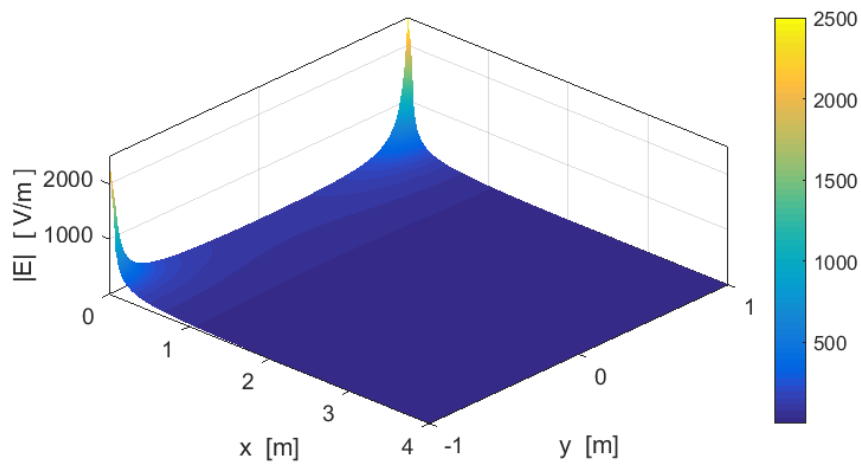
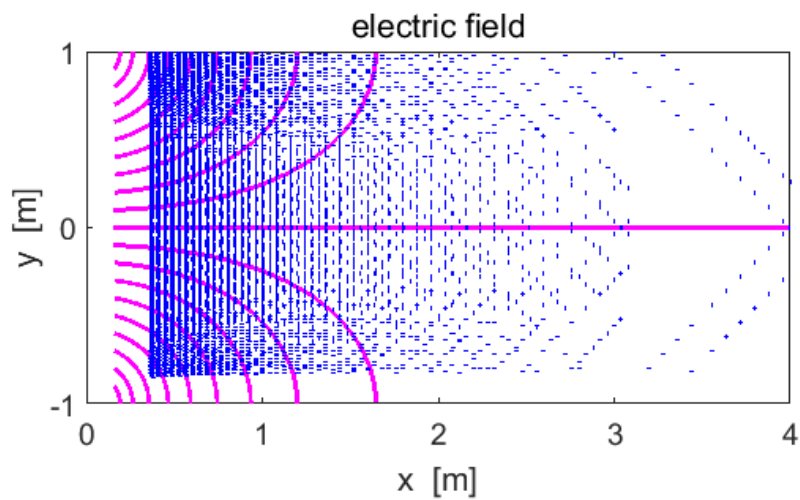


Fig. 5. Magnitude of the electric field.



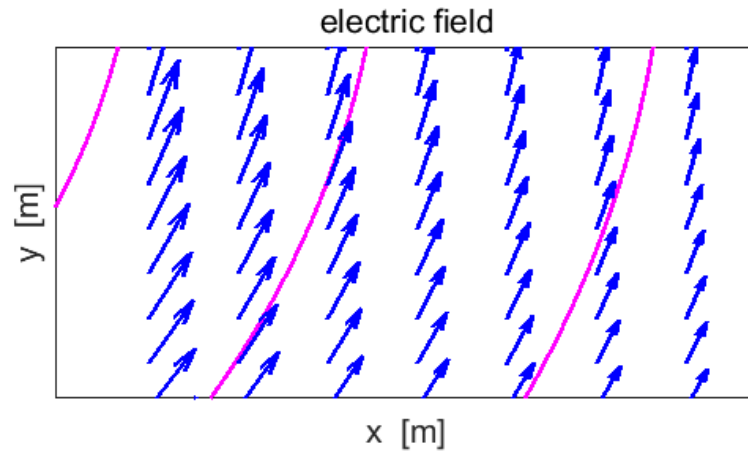


Fig. 6. The `quiver` command and `streamline` command are used to display the electric field. The zoom tool is used to show the direction of the electric field in a small region centred upon $x = 0.50$ m and $y = 0.85$ m).

Activity cemLaplace01.m

Use the mscript **cemLaplace01.m** to investigate the relaxation method to solve Laplace's equation in more detail. Determine the potential $V(x, y)$ inside a rectangle of dimensions 2.0 m x 1.0 m. Set all the boundaries to 10.0 V and all interior grid points to 0 V. Before you do the computation, guess the exact forms of the potential and the electric field. Use the values $N_x = 7$ and $N_y = 7$ for the number of grid points. You can view the number of iterations and the potential in the Command Window after running the program. How many iterations are required to achieve an accuracy of better than 1%? How does the number of iterations change if the number of grid points N_x and N_y are increased?

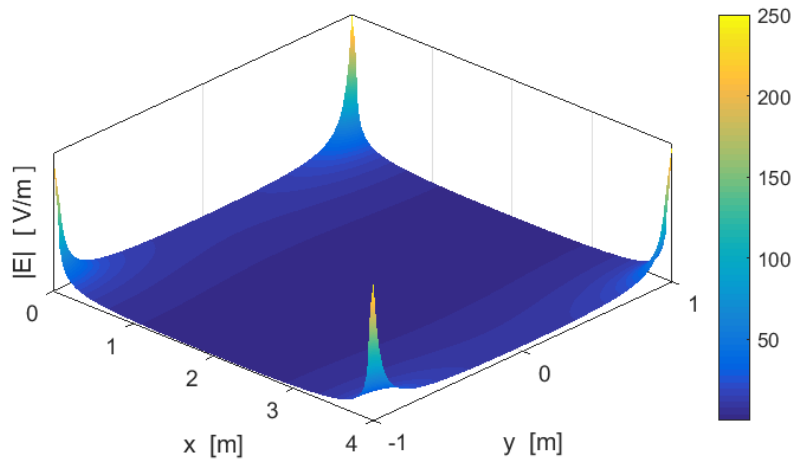
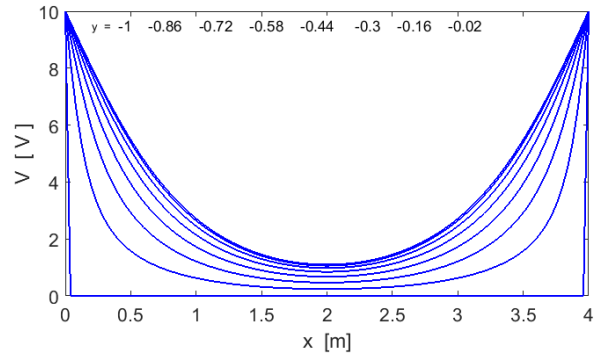
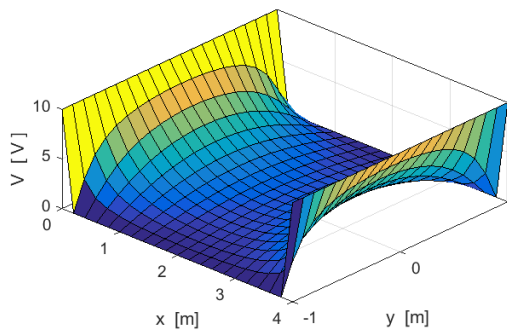
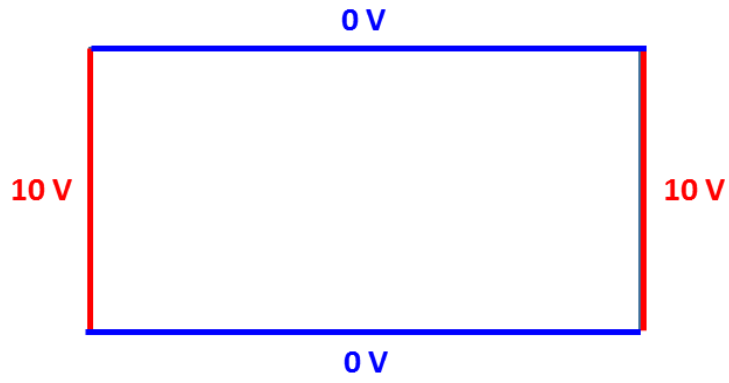


Table 1. The potential after 5 iterations

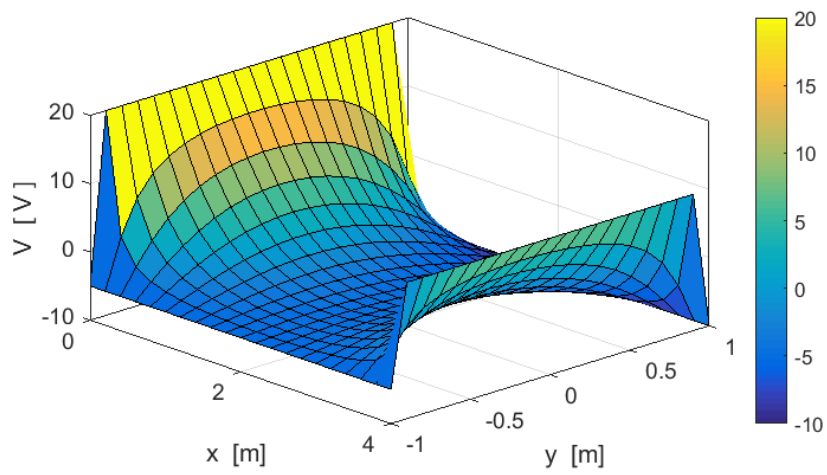
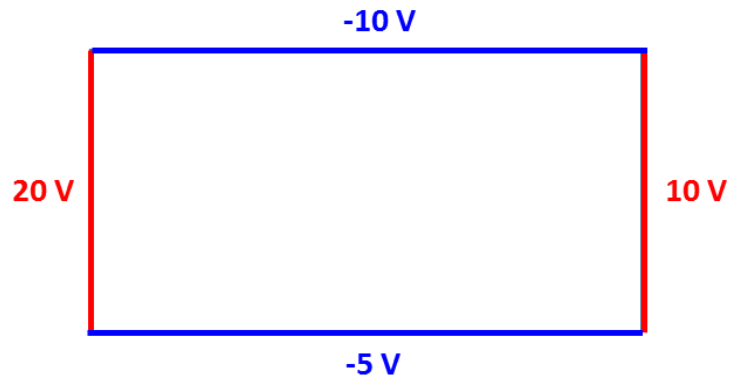
10.0	10.0	10.0	10.0	10.0	10.0	10.0
10.0	8.5	7.8	7.6	7.8	8.7	10.0
10.0	7.6	6.5	6.2	6.6	8.0	10.0
10.0	7.6	6.4	6.0	6.5	8.1	10.0
10.0	8.2	7.2	7.0	7.4	8.6	10.0
10.0	9.1	8.6	8.5	8.7	9.3	10.0
10.0	10.0	10.0	10.0	10.0	10.0	10.0



Example 3 cemLaplace02.m



Example 4 cemLaplace02.m



Example 5 cemLaplace02.m

Consider a [2D] charge free region of space with linearly increasing potentials on its boundaries



You need to modify the mscript for these boundary conditions and uncomment the code in the SETUP section which defines the linearly increasing potentials

```
% linearly increasing potentials on boundaries
%       m1 = 20/maxY; m2 = 20/maxX; m3 =
30/maxY; m4 = 10/maxX;
%       b1 = 0;       b2 = 20;       b3 = 10;
b4 = 0;
%
%       V(:,1)   = m1 .* y + b1;
%       V(:,end) = m3 .* y + b3;
%       V(end,:) = m2 .* x + b2;
%       V(1,:)   = m4 .* x + b4;
```

⇒

