MATH 405/607E Numerical methods for differential equations

Project 2

1 Question Review

For this project, our task is to solve elliptic, parabolic and hyperbolic equations (over the same domain) numerically using finite element method (FEM). The domain is shown as below.

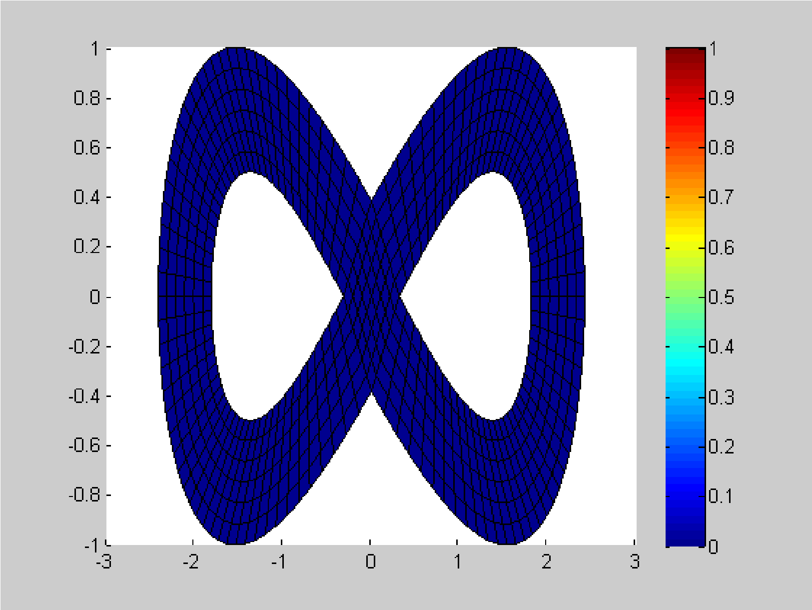


Figure 1: Domain

There are three equations to be solved:

a) Laplace equation:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

with boundary condition:

$$u_{left} = 1, \qquad u_{right} = 0.$$

b) Diffusion equation:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

with boundary condition: u((x,y) = left, t) = 1, u((x,y) \in Domain/left, t = 0) = 0.

c) Wave equation:

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 0$$

with boundary condition: u((x, y) = left, t) = 1, u((x, y) \in Domain/left, t = 0) = 0.

For the last two equations, transform them to ODE and then using backward Euler, Crank-Nicholson and RK4 algorithms and then analysis the stability. Get the requirement of stability on dt and dx.

2 Laplace Equation

After testing, I found that the script will run out of the memory when N > 12. Considering the time consuming in the next questions, I choose N =6 to construct the elements.

After deciding the elements, we need to construct matrix K and vector F as:

$$K^{(n)} = \int_{-1}^{1}\int_{-1}^{1} [J^{-1}\cdot B]^T \cdot [J^{-1}\cdot B]\,|\det(J)|\,d\xi d\eta,$$

$$F^{(n)} = \int_{-1}^{1}\int_{-1}^{1} f(x(\xi,\eta), y(\xi,\eta))\,N\,|\det(J)|\,d\xi d\eta.$$

$$K(GN,GN) = K(GN,GN) + K^{(n)}, \qquad F(GN) = F(GN) + F^{(n)},$$

Also we need to deal with the boundary condition as:

$$K(BC,:) = 0, \qquad K(BC,BC) = 1, \qquad F(BC) = \alpha,$$

Then solve the equations using solving K V = F and plot the solution as follows.

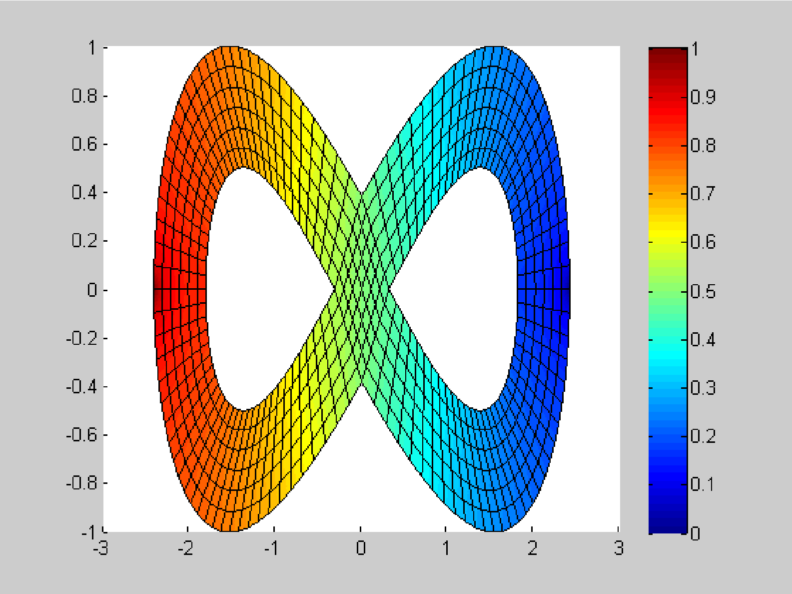


Figure 2: Solution of Laplace equation

2 Diffusion equation

First we use finite element method to construct K and M matrix. K matrix is the same as previous. M matrix is defined as follows:

$$M^{(n)} = \int_{-1}^{1}\int_{-1}^{1} N^T \cdot N\,|\det(J)|\,d\xi d\eta,$$

Also we need to deal with the boundary condition as:

$$M(BC,:) = 0, \qquad M(BC,BC) = 1, \qquad K(BC,:) = 0, \qquad F(BC) = 0.$$

Then the system reduces to the system of linear ODEs:

$$\dot{U} \;=\; -M^{-1} \cdot K \cdot U + M^{-1} \cdot F,$$

Then next step is to decide the steps of h. We try t = linspace(0, 5, N) with N ranging from {5000, 10000, 20000, 50000}and plot the eigenvalues of Kbar on the stable on top of the stability regions as follow for three methods respectively.

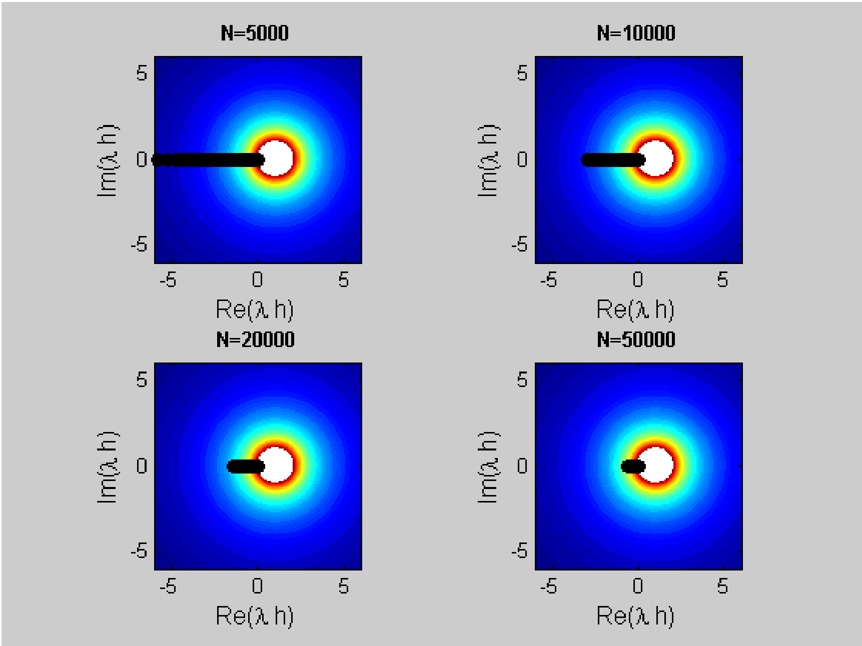


Figure 3: Backward Euler

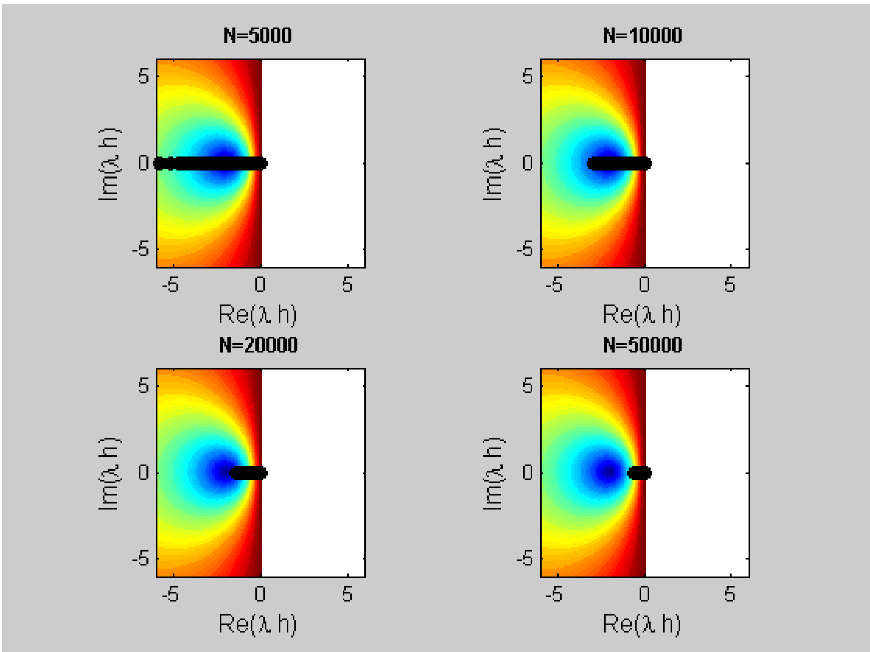


Figure 4: C-N

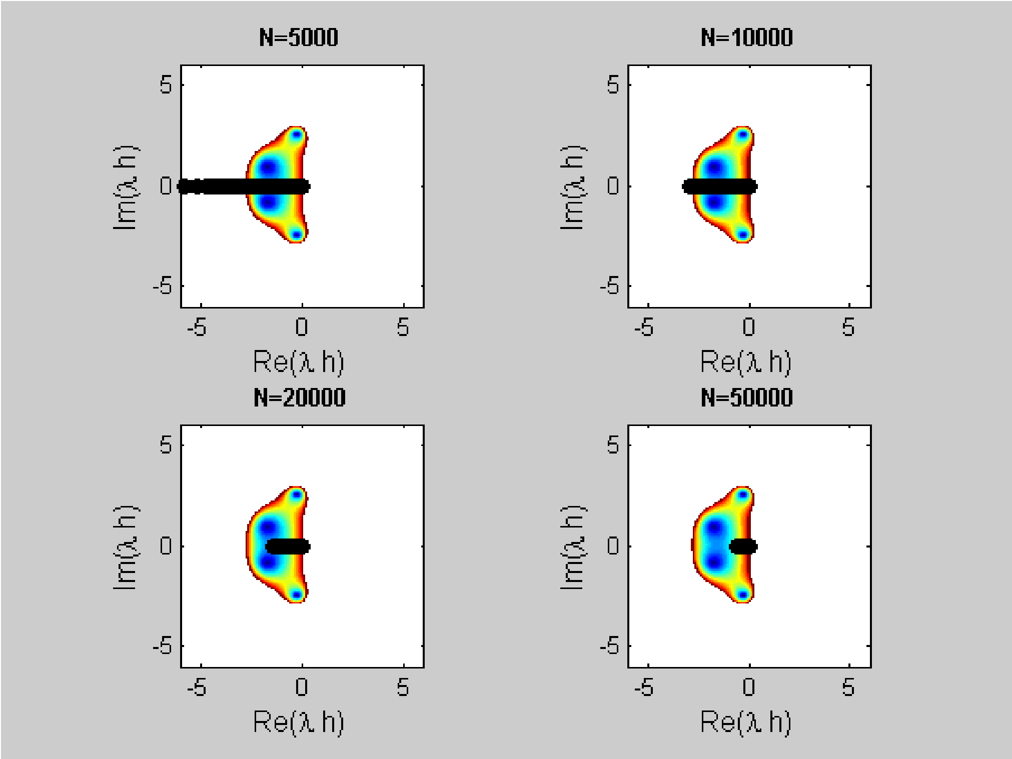


Figure 5: RK 4

We find that it is stable for all N for Backward Euler and C-N, but for RK 4, it may be unstable when N is small. Based on figure 5, we choose N = 2e4.

Then solve the equation by three methods, and plot the solutions as follows:

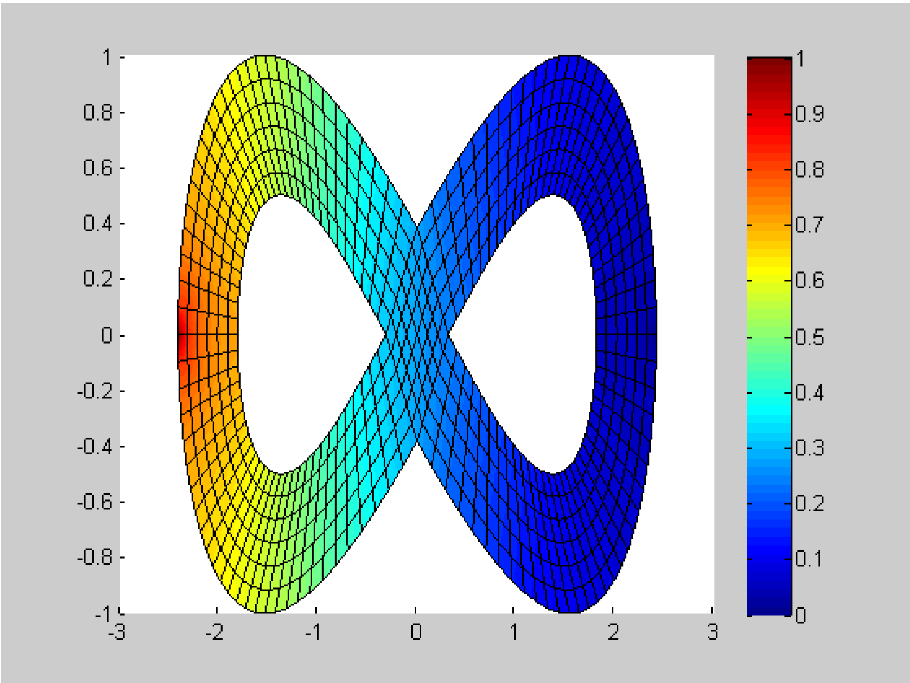


Figure 6: Solution of diffusion equation by Backward Euler

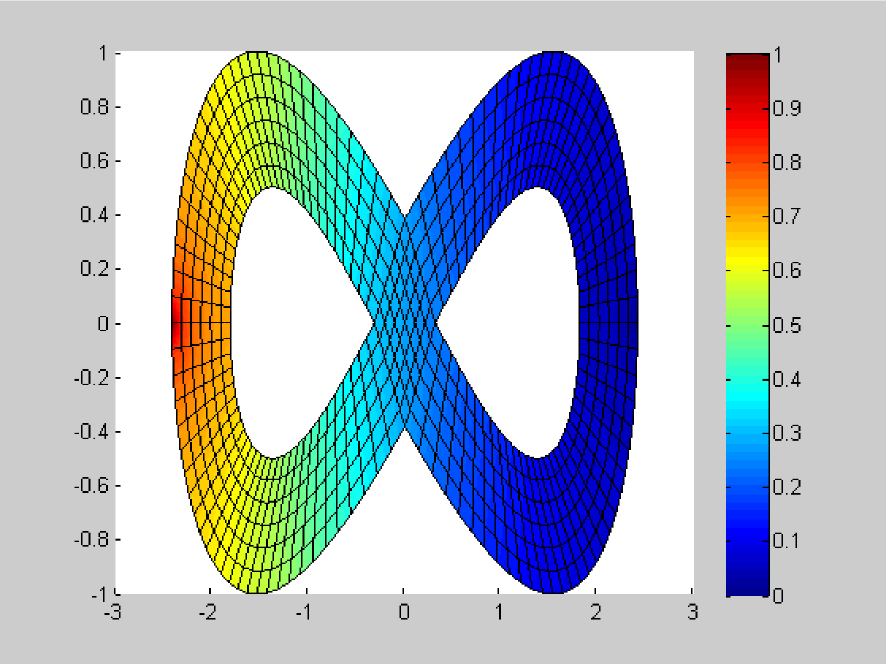


Figure 7: Solution of diffusion equation by C-N

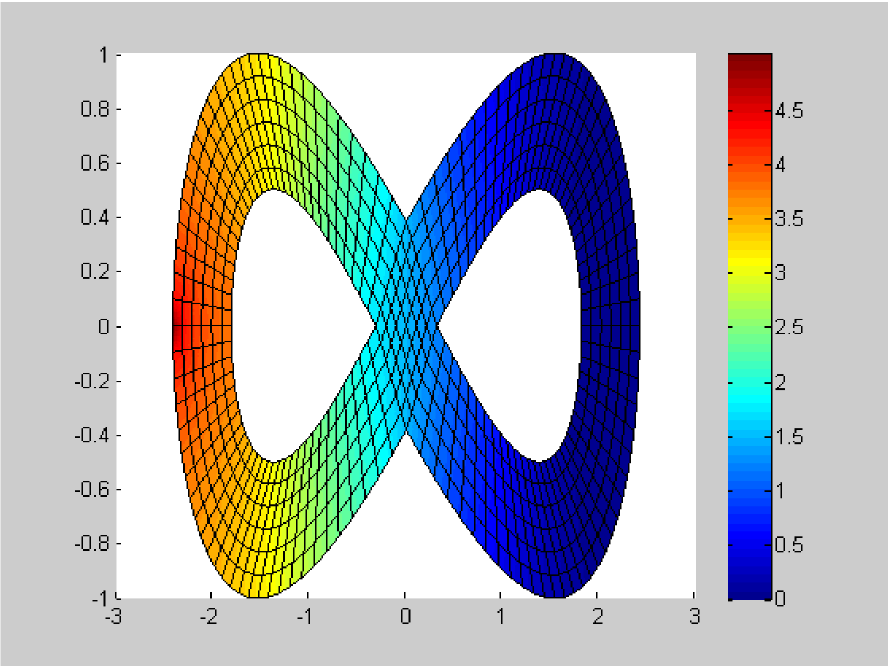


Figure 8: Solution of diffusion equation by RK 4

To compare this three methods, we find for the stable region, Backward Euler > C-N > RK 4 (See figure 3 - 5), so for stability, Backward Euler > C-N > RK 4. However, through the practical experience, we find for the speed of each method, RK 4 > Backward Euler > C-N. I believe this is because the method of C-N and Backward Euler involved in calculating the inverse of the matrix.

Next task is to discover the relation between dt and dx to achieve the stability. By noticing that dx * N is a constant where N is the number of layers of the elements, we use 1/N to represent dx. To achieve the stability, we need dt * eig(Kbar) all lays in the stable region, we use 1/max(abs(eig(Kbar))) to represent dt. We vary N from 1 to 10, and plot dx vs dt as follows:

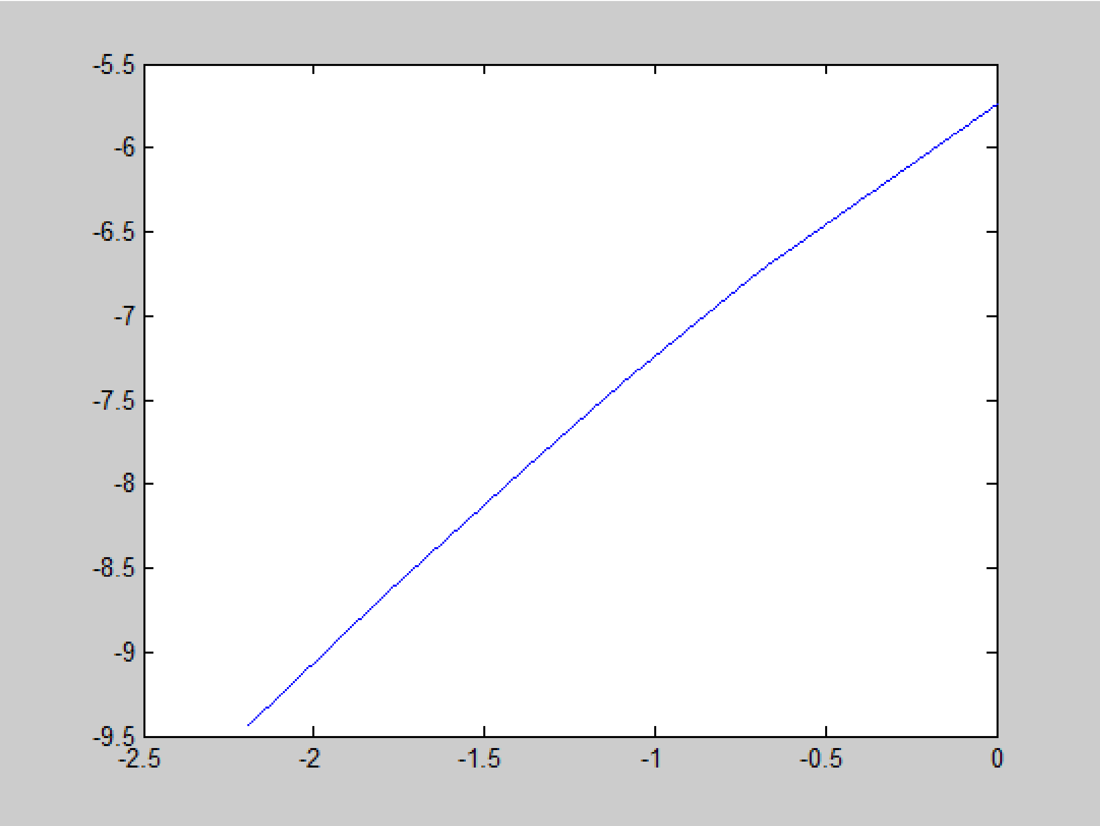


Figure 9: dx vs dt

Use the leastsquare method to fit the equation, we find the slope is 1.7063, which is near 2.

3 Wave equation

First we use finite element method to construct K and M matrix as before. Then the PDE system transforms into the system of ODEs:

$$\ddot{U} = -M^{-1} \cdot K \cdot U + M^{-1} \cdot F,$$

Also this ODEs system can be rewritten as following first order system

$$\begin{pmatrix} \dot{U} \\ \dot{W} \end{pmatrix} = \begin{pmatrix} 0 & I \\ -M^{-1} \cdot K & 0 \end{pmatrix} \begin{pmatrix} U \\ W \end{pmatrix} + \begin{pmatrix} 0 \\ M^{-1} \cdot F \end{pmatrix}$$

Then next step is to decide the steps of h. We try t = linspace(0, 5, N) with N range from {100, 200, 400, 1000} and plot the eigenvalues of Kbar on the stable on top of the stability regions as follow for three methods respectively.

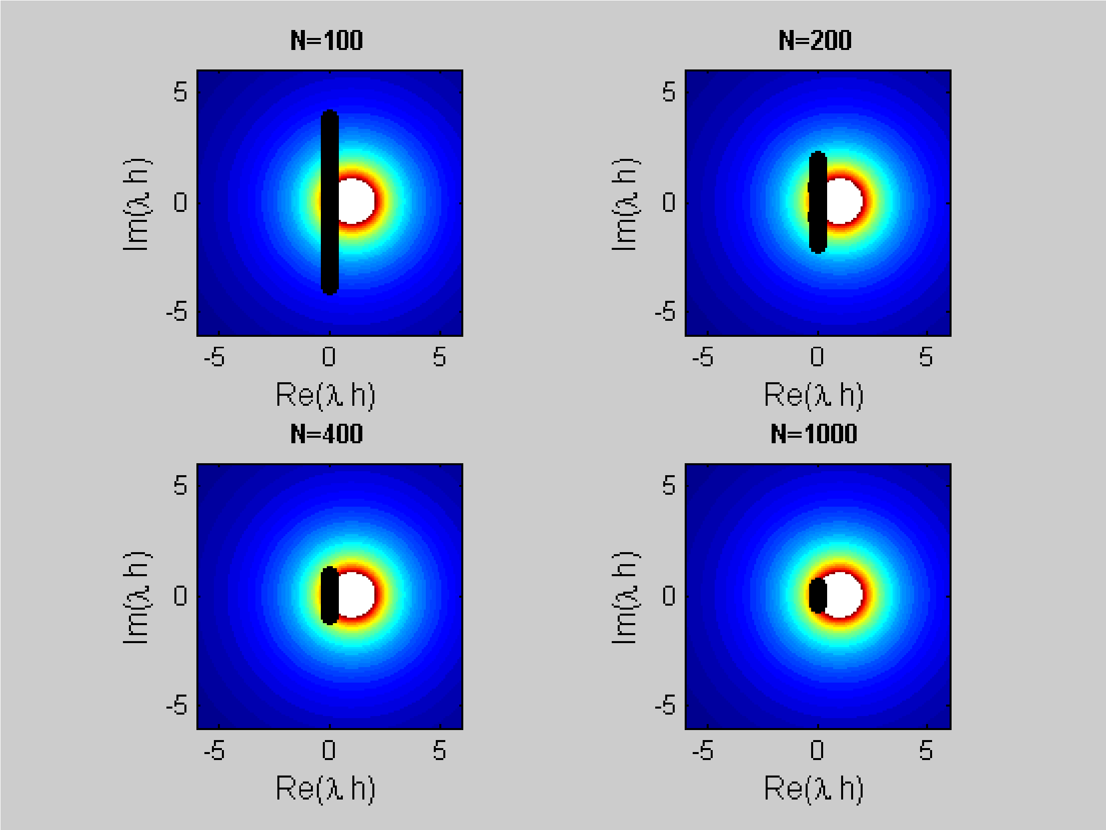


Figure 10: Backward Euler

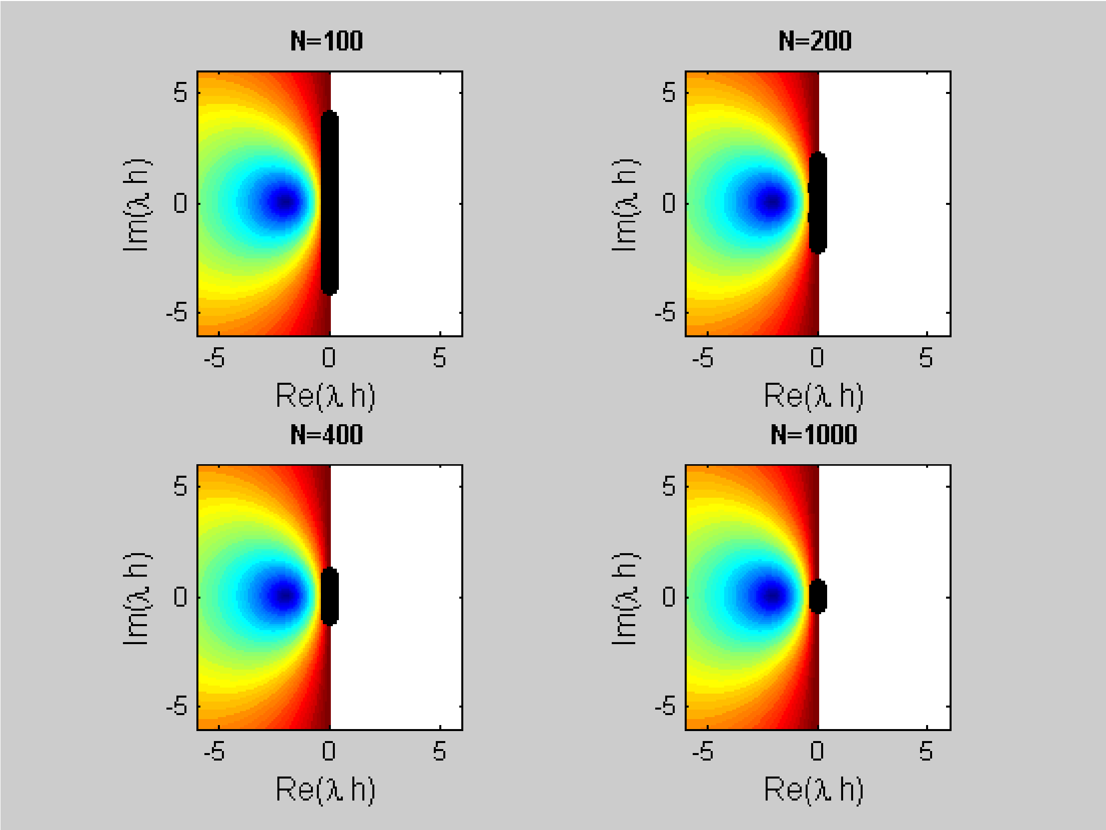


Figure 11: C-N

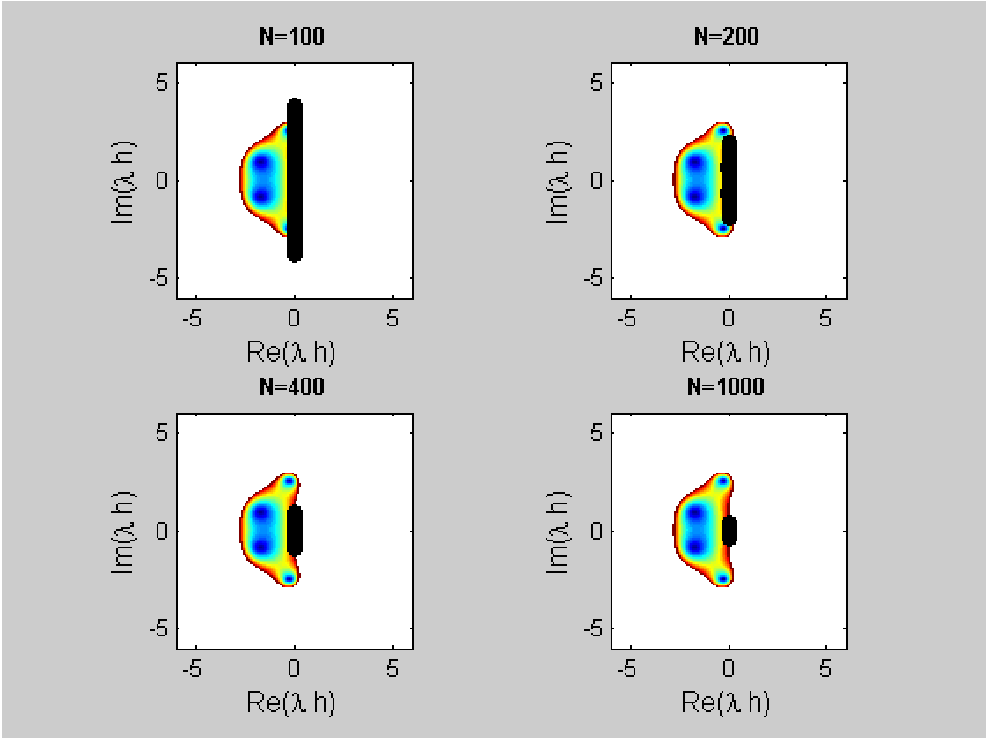


Figure 12: RK 4

Same as the previous question, we find that it is stable for all N for Backward Euler and C-N, but for RK 4, it may be unstable when N is small. Based on figure 12 and the consideration of precise, we choose N = 1e3.

Then solve the equation by three methods, and plot the solutions as follows:

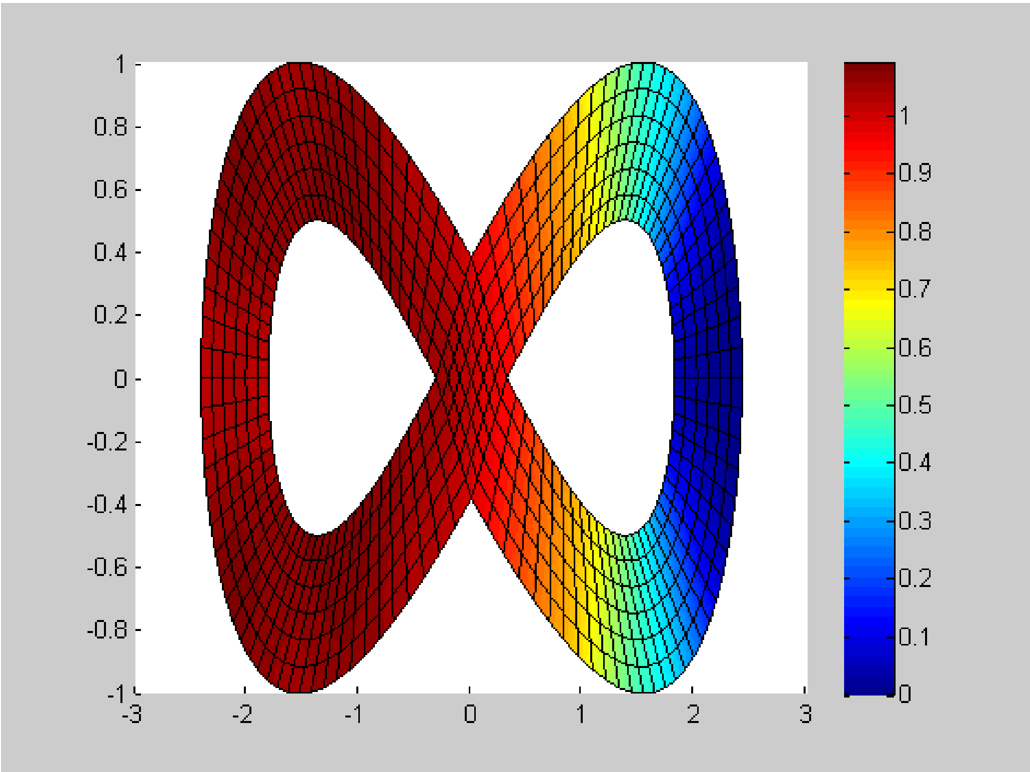


Figure 13: Solution of wave equation by backward Euler

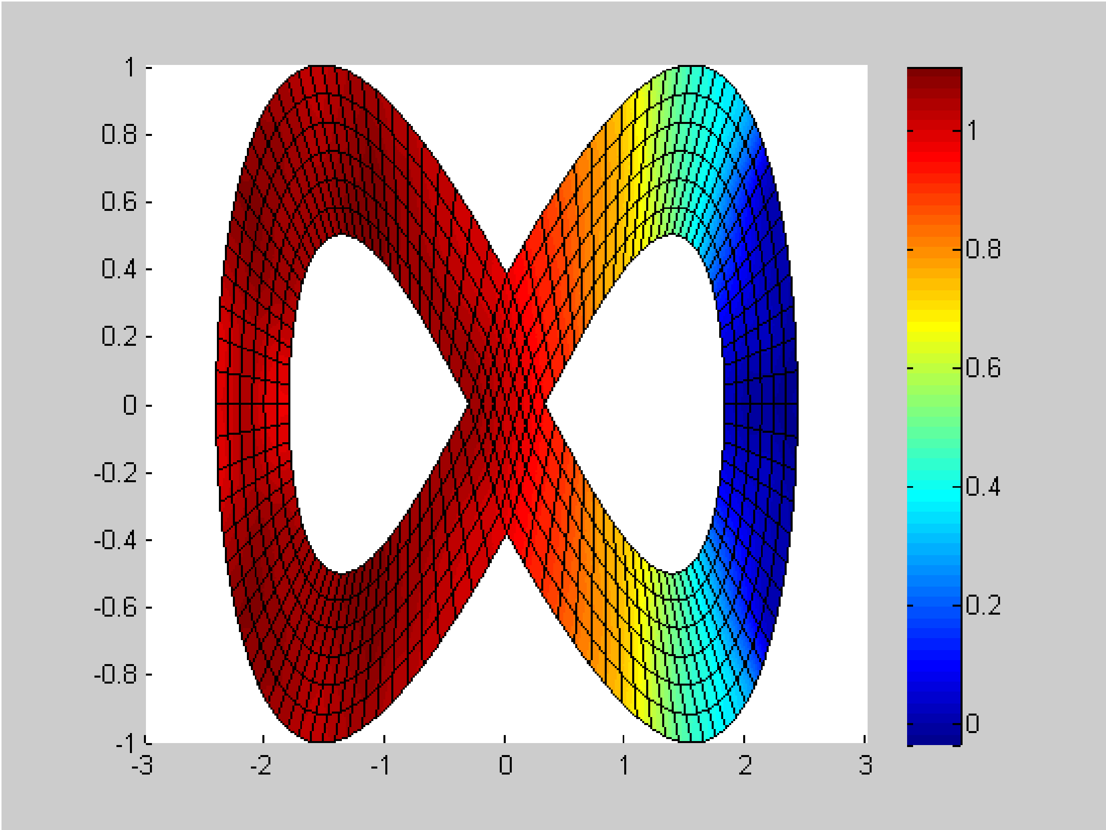


Figure 15: Solution of wave equation by C-N

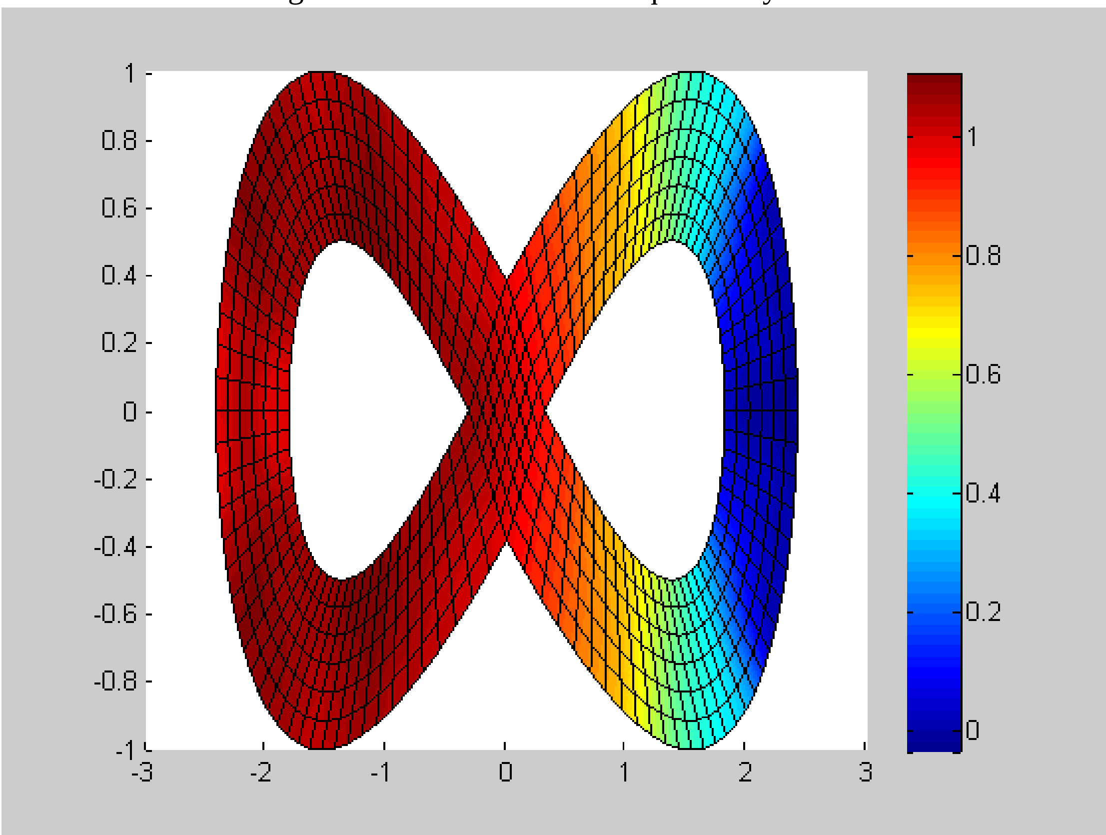
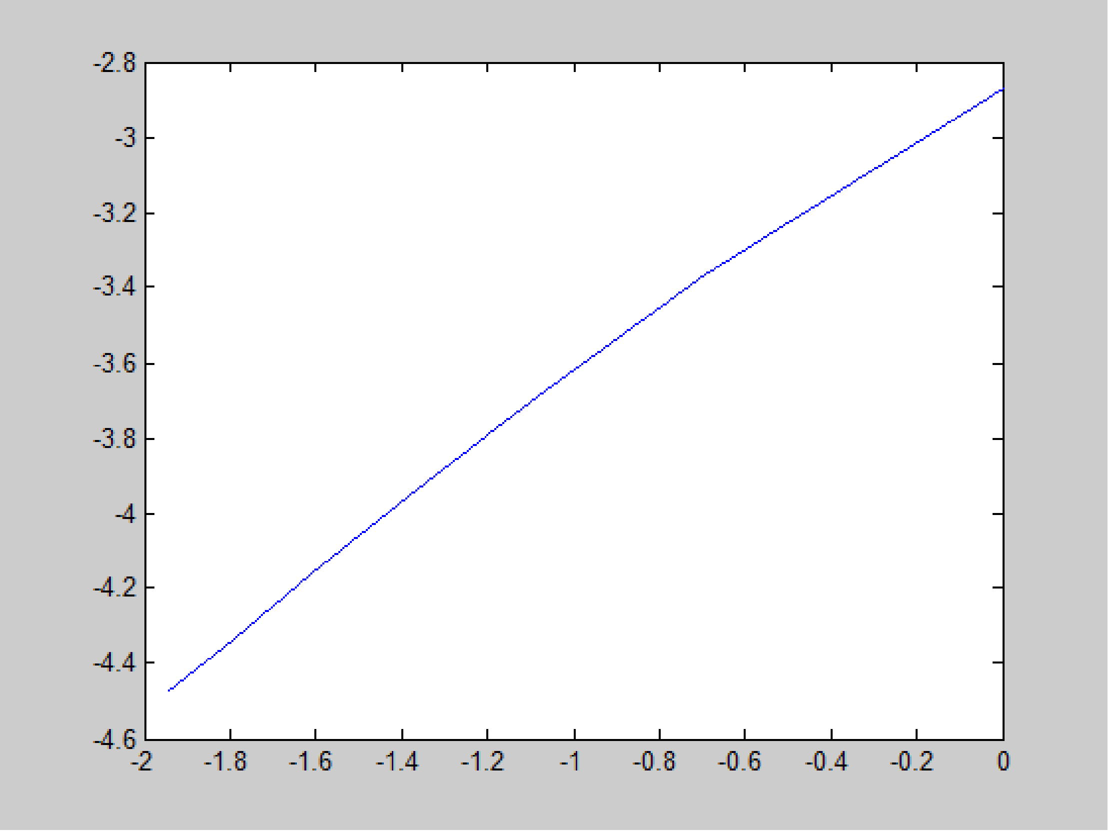


Figure 15: Solution of wave equation by RK 4

For the aspect of stability, it is quite similar with the case of heat equation. From figure 10 - 12, we find for the stable region, Backward Euler > C-N > RK 4, so for stability, Backward Euler > C-N > RK 4, that means, Backward Euler and C-N are more compariable . However, through the practical experience, we find for the speed of each method, RK 4 > Backward Euler > C-N. I believe this is because the method of C-N and Backward Euler involved in calculating the inverse of the matrix.

Next task is to discover the relation between dt and dx to achieve the stability. By noticing that dx * N is a constant where N is the number of layers of the elements, we use 1/N to represent dx. To achieve the stability, we need dt * eig(Kbar) all lays in the stable region, we use 1/max(abs(eig(Kbar))) to represent dt. We vary N from 1 to 7, and plot dx vs dt as follows:



Use the leastsquare method to fit the equation, we find the slope is 0.8315, which is near 1.