

MECH5510: Computational and Experimental Methods: Computational Assignment 2

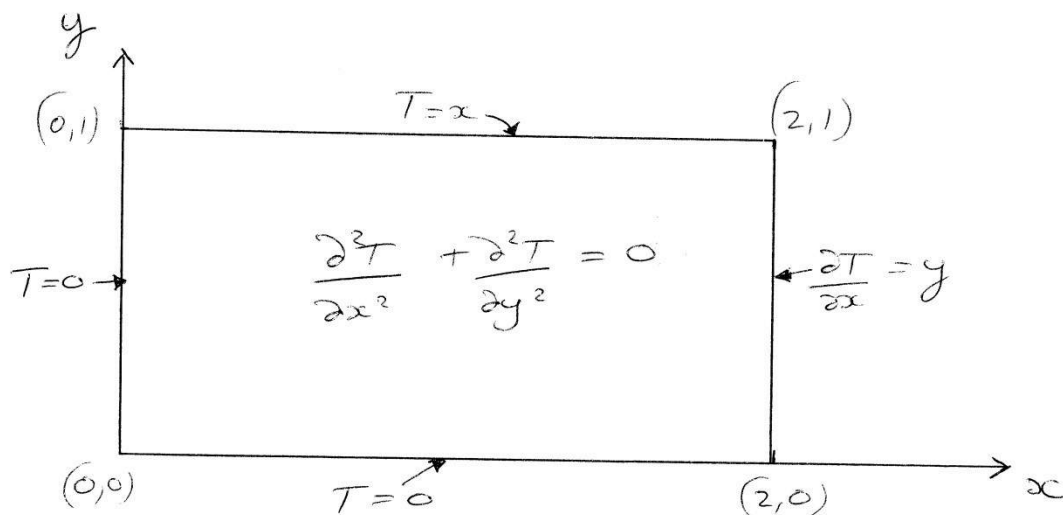
This document contains a description of the second computational assignment which is worth 25% of the overall mark for the module. The assignment is marked out of 100. Your assignment should be submitted on the VLE by **12 noon, Monday 12th January 2015**.

Please keep your submissions as brief as possible by simply supplying the information, programs and results specified below. You will not be awarded extra marks for irrelevant information.

You are free to use and modify the sample programs you have been given during the course.

a) The Steady-State Heat Diffusion Equation (40 marks)

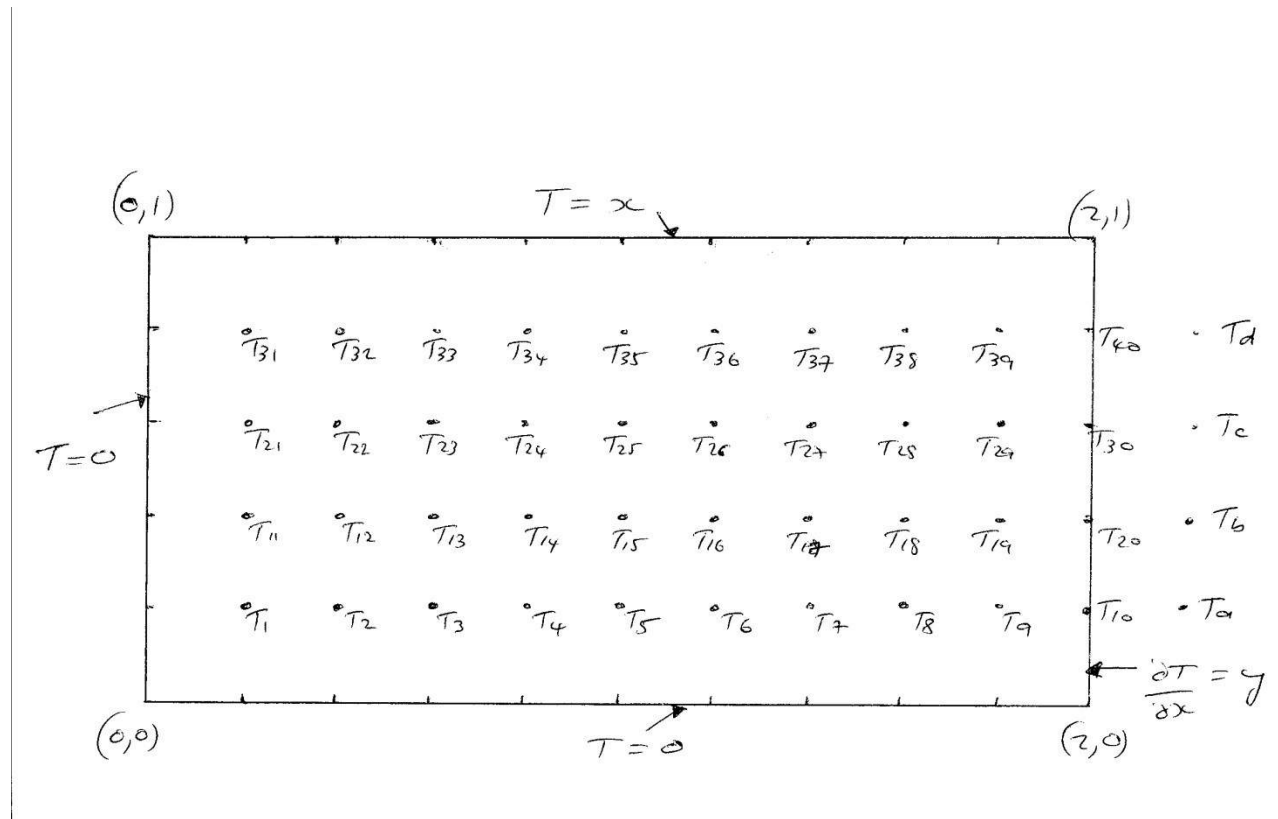
Consider steady state heat diffusion in the following domain:



The boundary conditions for this problem are:

- $T=0$ for $x=0$, $0 \leq y \leq 1$
- $T=0$ for $y=0$, $0 \leq x \leq 2$
- $T=x$ for $y=1$, $0 \leq x \leq 2$
- $\frac{\partial T}{\partial x} = y$ for $x=2$, $0 \leq y \leq 1$

Consider the following Finite Difference mesh with $N=4$ internal nodes in the vertical direction and a total of 10 nodes (including at $x=2$) in the horizontal direction. In all cases you will consider here, you will use the same uniform grid spacing in the x and y directions. This means that if N is the number of internal nodes in the vertical direction then the number of nodes in the horizontal direction is $2(N+1)$. In the example shown below this leads to a grid spacing $\Delta = \Delta x = \Delta y = 0.2$



The node numbering scheme to be used is shown above where T_i indicates the unknown temperature at node i . As in the relevant section of the notes, you will need to introduce “ghost” nodes outside the domain in order to apply the Finite Difference form of the heat diffusion equation

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

at the boundary nodes T_{10} , T_{20} , T_{30} and T_{40} and a second order, central difference approximation to the boundary condition $\frac{\partial T}{\partial x} = y$ at $x=2$ in order to eliminate the ghost nodes from the Finite Difference equations.

Write a Matlab M-code program which solves the Finite Difference equations for the unknown temperatures T_i where the value of N , the number of internal nodes in the vertical direction, is specified as a parameter at the beginning of the program. Your program should also do the following.

- Write the numerical values of temperature and the corresponding value of the analytical solution to this problem, $T = xy$, at each of your grid points, to a text file.
- Calculate the maximum percentage difference between your numerical solution and the analytical solution in the form $100 \frac{(\text{numerical} - \text{analytical})}{\text{analytical}}$

over all the grid points and write this value to the same text file to which you have written the numerical and analytical results.

You are asked to hand in the following:

(i) A copy of your Matlab M-code program, which is documented similarly to the programs you have been given during the lectures, together with your results file for the case with $N=4$, which show the numerical and analytical values at each grid point and which contains the maximum percentage difference between the analytical and numerical solution over the entire grid of 40 points.

(ii) A sheet on which you have derived (i.e. do not simply write them down) the Finite Difference equations for the above grid with $N=4$ (and a total of 40 unknowns) at nodes 1 and 40.

(iii) Generate a 3-D plot of the temperature against x and y including the boundary values, e.g. using the Matlab function *surf*.

When writing your program you may find it useful to populate your temperature matrix by considering the equations at the following nodes separately:

- i. All nodes not influenced by the boundary conditions, i.e. those not adjacent to the boundaries.*
- ii. Each of the four corner nodes separately*
- iii. Nodes adjacent to the bottom ($y=0$) boundary but not at the corners*
- iv. Nodes adjacent to the left boundary ($x=0$) but not at the corners*
- v. Nodes adjacent to the upper boundary ($y=1$) but not at the corners.*
- vi. Nodes on the right hand boundary ($x=2$) but not at the corners.*

Note that for the grid given above these different sections would be

- i. Nodes 12-19, 22-29.*
- ii. Nodes 1, 10, 31, 40*
- iii. Nodes 2-9*
- iv. Nodes 11, 21*
- v. Nodes 32-39.*
- vi. Nodes 20, 30.*

b) A Parabolic Partial Differential Equation (30 marks)

In this question you will solve the time-dependent heat diffusion equation

$$\pi^2 \frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2} \text{ over the region } 0 \leq x \leq 1, 0 \leq t \leq 10 \text{ subject to the boundary conditions } T(x,0) = \sin \pi x \text{ and } T(0,t) = T(1,t) = 0.$$

(i) By modifying the program tdepheat_theta.m for the general θ method, write a Matlab program that solves this equation numerically using the θ method with N evenly spaced nodes in the x direction (node 1 is at $x=0$ and node N is at $x=1$) and Nt evenly spaced time points in the time direction (time node 1 corresponds to $t=0$ and node Nt is at $t=10$). In this case the grid spacing in the x direction is $1/(N-1)$ and the time step is $10/(Nt-1)$. Your program should also plot out your numerical solution at $t=10$ for $0 \leq x \leq 1$. Hand in a copy of your program, highlighting the modifications to tdepheat_theta.m that you have made to solve this problem. (15 marks).

(ii) Use your program to plot out your numerical solution at $t=10$ for $0 \leq x \leq 1$ using the following parameters:

(a) $\theta=0.5$ (Crank-Nicolson) with $N=11$ and $Nt=181$.

(b) $\theta=1.0$ (Backward Euler) with $N=11$ and $Nt=181$.

(c) $\theta=0.0$ (Forward-Euler) with $N=11$ and determine the approximate value for Nt at which this approach becomes stable.

For each set of parameters, hand in hard copies of your graphs. Hence or otherwise discuss the stability of the solutions with the 3 methods (15 marks).

c) A Hyperbolic Partial Differential Equation (30 marks)

In this question you will solve the simple wave equation $4 \frac{\partial^2 y}{\partial t^2} = \frac{\partial^2 y}{\partial x^2}$ in the region

$0 \leq x \leq 1, t \geq 0$ subject to the initial conditions $y(x,0) = f(x) = 0$ and

$\frac{\partial y}{\partial t}(x,0) = g(x) = \sin(2\pi x)$ at $t=0$. You are also given that $y(0,t) = y(1,t) = 0$ for all t .

Write a Matlab program which solves this equation using finite differences. The boundary conditions for this problem represent a higher order vibration mode in comparison to the exercise covered in the notes, as defined by the initial condition for the velocity.

Your program should use a definition for Nx (the number of uniformly spaced nodes in the x direction including the two boundary nodes at $x=0$ and $x=1$) and calculate the time step to apply on the basis that $\Delta t = \frac{\Delta x}{c}$. Nt (the number of time steps) should be setup to show at least 1.5 full oscillations of the wave. You should generate solutions based on equation 8 and equation 15 (in the notes) to define the initial conditions.

You should also generate an analytical solution based on the application of D'Alembert's method, equation 12. The output should include a 3-D figure (e.g. using *surf*) of the solution obtained using finite differences and equation 15. You should also generate a plot which shows the accuracy of the two methods in comparison to the analytical solution, e.g. by plotting $y_1(\text{round}(Nx/4),:)$.

You are asked to hand in the following:

- a) A copy of your program with brief explanatory comments
- b) A graph of the results and comparison of D'Alembert and the two finite difference approaches with $Nx = 21$, or equivalently $\Delta x = 0.05$.

You should also submit a copy of your Matlab programs as an appendix to your submission.