MAT 418/518 Project – Fall 2024

Introduction

This is a group project. You will work in groups of two. The end product will be several progress reports and a final report containing all of the results written in paragraph form with appropriate derivations, results, and plots.

All work should be typed with double-spacing and 12 pt font. You may use LaTeX. You will be expected to use correct English grammar and punctuation. This is a report and thus you will use proper sentence and paragraph formatting. You are not just listing completed steps like a homework set.

You will be graded on the evidence of work, mathematical detail and understanding, proper exposition and neatness. Your work should also be supported with properly labeled and embedded plots and numbered/referenced important equations. Any outside references used should be cited as well. These reports will count towards the project component of your grade.

Part I. 1D Heat Equation and 2D Model (Due Oct 17)

For the first part, you will get familiar with MATLAB by playing with the MATLAB code HeatEqn1d.m I will have demonstrated in class for the 1D heat equation. The instructions for how to do this are below. Then, you will write up your results and observations for your first report. In particular, review Chapter 3 and your notes to understand how one develops a finite difference scheme to solve the heat equation in one dimension. Then, download the file HeatEqn1d.m by right-clicking the link and saving to the working directory for your version of MATLAB. [Or, you might be able to click the link and copy and paste the code to the MATLAB editor. Then, save the file as indicated below.]

For those not familiar with MATLAB, you can look at Dr. Herman's Wiki called MATLAB a brief introduction for students or an online tutorial like https://web.eecs.umich.edu/~aey/eecs451/matlab.pdf. You can access MATLAB from the computer labs or through https://horizon.uncw.edu/. These codes should also work in GNU Octave, an open source MATLAB clone at https://www.gnu.org/software/octave/. Finally, students can download MATLAB to their personal machine and not work online. See https://uncw.teamdynamix.com/TDClient/1875/Portal/KB/ArticleDet?ID=142181 .

If you are opening MATLAB and need to load your file. Make sure you have opened the directory in which you have downloaded the file and can see it. Type **edit HeatEqn1d** in the Command Window. The editor should appear with the contents of the m-file. Alternatively, you can copy the text in the file into the editor and save as **HeatEqn1d.m**. (Sometimes single quotes do not copy correctly. If that is the case, retype the quotes.)

In the Command Window type Heat Eqn1D after the prompt (>>). This will run the script in the m-file. You will see a Figure show with a preprogrammed solution to the one dimensional heat equation.

If you type **alpha** in the Command Window, you should get the current value of $\alpha = \frac{k\Delta t}{\Delta x^2}$. a constant . In Chapter 3 we show that we need $\alpha \le \frac{1}{2}$ for the numerical scheme to be stable.

Play with the m-file, changing the following in the code:

- Increase **N** to make plot smoother, deleting figure beforehand, if needed. [Note: if you mess up the original file, just download it again from the class webpage.]
- Record what you see in the Figures.
- Increase **M** and repeat until plot looks good and $\alpha \le \frac{1}{2}$.
- Repeat for different **N** and **M** and verify that one needs $\alpha \le \frac{1}{2}$ to get nice plots.
- You can move the **plot** statement to before the last **end** statement and add **hold** before and after the plot command. This will allow you to plot solutions at each time step. To not plot every time step, change plot line to following:

  ```
  if mod(j,10)==0
     plot(x, u1);
  end
  ```

  What does this statement do? What if you change the 10?
- Settle on particular values for **N** and **M** and try different initial conditions: Change the initial condition to:

  ```
  u0(i)=sin(3*pi*x(i));
  u0(i)=x(i)*(1-x(i));
  ```

- You can replace the fixed boundary conditions with insulating conditions at both ends. Replace the lines

  ```
  u1(1)=0;
  u1(N+1)=0;
  ```

  with the lines

  ```
  u1(1)=u1(2);
  u1(N+1)=u1(N);
  ```

  What do these lines do?

- Now you will need to use different initial conditions. Try the following and record your results.

```
u0(i)=cos(pi*x(i));
u0(i)=cos(3*pi*x(i));
u0(i)=x(i)*(2-x(i));
```

Write up your results for your first report describing the behavior of the numerical solutions you have seen. What do you think the numerical scheme might look like for the two-dimensional heat equation? (Do not put into MATLAB at this time.)

Part II. Exact Solutions (Due Oct 31)

Consider a thin plate of length L and width W governed by the heat equation $u_t = k(u_{xx} + u_{yy})$ for $u = u(x, y, t)$. Each group will be given a value for *k*, a set of boundary conditions at $x = 0, L,\ y = 0, W,$ and an initial condition of the form $u(x, y, 0) = f(x, y)$. You can see all of the problems at the course web site. Each group will be assigned a different problem.

For your problem,

1. Find the general solution, $u(x,y,t) = \sum_{n,m} c_{n,m}\phi_{n,m}(x,y,t)$ for $\phi_{n,m}(x,y,t)$ a set of product solutions satisfying the boundary conditions.
2. For the given initial condition, find the particular solution of your problem.
3. For several values of *t*, plot the solution in Maple. [MATLAB code: See Problems]
4. Write up your results for second report describing the behavior of the solution you have found.

Part III. Numerical Solution to 2D Problem (Due Nov 12)

Study numerical solutions of the 2D heat equation using the file HeatEqn2D.m. This file can be also be found on the materials page. There are comments to aid in figuring out how to adapt the code to your problem. First, save and run the m-file in MATLAB. Note the parameters, initial condition, and boundary conditions used. Play with the grid size, time interval, etc. You will note that you can also produce a movie showing the solution as it evolves in time.

Now, change the m-file to provide a numerical solution of your original problem. You might want to save the m-file under a new name. You can change **L, W, T, k**, the initial condition, and the boundary conditions. How does this solution compare to what you found in Part II?

It should also be possible to plot your analytical solution (Part II, #2 in MATLAB just as you had in Maple. The file HeatEqn2Dexact.m is an example as to how you can plot a series solution. Namely, if you have the series $u(x,y,t)=\sum_{n,m} c_{n,m}\phi_{n,m}(x,y,t),$ then you can “loop” though the indices summing one term at a time: for each **n** and for each **m**, **u = u + c(n,m) phi(n,m);** This means add **c(n,m) phi(n,m)** to whatever **u** is to get a sum starting with **u = 0**. See the m-file. You can change **L, W, T, k**, and in lines 30, 53, and 54 the form of your series.

Write up your results in a final comprehensive report. The final report should be a combination of the 2D solution and plots for Part II, the solutions for Part III, and a discussion of what you learned about the analytical and numerical methods and how they seem to compare.