MONASH University

School of Mathematical Sciences
MTH3011
Computer laboratory sheet 3

MTH3011 PARTIAL DIFFERENTIAL EQUATIONS

Computer laboratory sheet 3 – MATLAB version

This final computer laboratory session involves comparing several explicit numerical methods for obtaining an approximate solution of the advection equation. It is assumed that you are familiar with the material introduced in Computer laboratory sheets 1 and 2 as well as the relevant theory on the numerical solution of the advection equation, as covered in lectures.

The main objectives of the session are:

- to illustrate and compare the properties of three explicit methods for determining an approximate solution of the advection equation;
- to compare the numerical results with a known exact solution (and calculate the errors) when the initial condition and exact solution contain are discontinuous;
- to practice some of the common procedures (including numerical calculations using formulas, plotting and printing two-dimensional solutions) that can be used for the numerical techniques in these sessions.

Specifically, we compare three methods of determining an approximate solution of the advection equation

$$\frac{\partial u}{\partial t}+\frac{\partial u}{\partial x}=0 \quad \text{over } 0<x<3 \text{ for } t>0$$

when the initial condition is *discontinuous*, with $u(x,0)=1$ for $x<1$ and $u(x,0)=-1$ for $x>1$. The boundary condition $u(0,t)=1$ is applied at the left-hand boundary and, for simplicity, we assume that $u(3,t)=-1$ for $t\le 2$ (as applies to the exact solution) at the right-hand boundary. To avoid potential difficulties at that boundary, the solutions will only be calculated until $t=1$.

The approximate numerical solution will be compared with the ***exact solution*** for this problem, which can be determined using the method of characteristics to be

$$u(x,t)=1 \text{ for } 0\le x<1+t \text{ and } u(x,t)=-1 \text{ otherwise.}$$

[Another way to express this is $u(x,t)=-\mathrm{sgn}(x-1-t)$ for $t\le 2$ where ‘sgn’ is the sign function.]

Forward-time centred-space (FTCS) method

As seen in lectures, and on Computer laboratory sheet 2, the forward-difference method in time with centred spatial differences gives that an approximate solution u_i^k for the exact solution $u(x_i,t_k)$ at $x_i=i\Delta x$ and $t_k=k\Delta t$ is obtained from the recurrence relation

$$u_i^{k+1}=u_i^k-\Delta t\left(\frac{u_{i+1}^k-u_{i-1}^k}{2\Delta x}\right) \quad \text{for } i=1,2,3,\dots,(n_x-1) \text{ and } k=0,1,2,3,\dots,$$

starting from the given initial condition $u_i^0=-\mathrm{sgn}(x-1)$ and using the boundary conditions $u_0^k=1$ and assumed values $u_{nx}^k=-1$ up to $t=1$. Here $\Delta x=3/n_x$ is the spatial step length and the calculations proceed for n_t time steps of length Δt.

Leith's method

Leith's method for the partial differential equation above gives that an approximate solution u_i^k for the exact solution $u(x_i,t_k)$ is determined by

$$u_i^{k+1} = u_i^k - \Delta t\left(\frac{u_{i+1}^k - u_{i-1}^k}{2\Delta x}\right) + \frac{(\Delta t)^2}{2}\left(\frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{(\Delta x)^2}\right) \quad \text{for } i = 1,2,3,\ldots,(n_x - 1) \text{ and } k = 0,1,2,3,\ldots$$

with the same initial and boundary conditions as used above for the FTCS method.

First-order upwind difference method

The first-order upwind-difference method for the partial differential equation above, gives that an approximate solution u_i^k is obtained from

$$u_i^{k+1} = u_i^k - \Delta t\left(\frac{u_i^k - u_{i-1}^k}{\Delta x}\right) \quad \text{for } i = 1,2,3,\ldots,n_x \text{ and } k = 0,1,2,3,\ldots$$

with the same initial and boundary conditions for $u_0^k = 1$ as used above. Note that the value of u_{nx}^k can be calculated using this formula, rather than being assumed to remain fixed up to $t = 2$.

Class exercise (to be completed during the laboratory session for assessment)

Download the sample template file for this week

1. Log in to the my.monash portal using your favourite browser, start Moodle and then go to the 'Teaching materials/Computer laboratory 3 sheets /MATLAB version' sub-section for this unit.

2. Click on the link for 'matlabsheet3.zip' and then click 'Open' to reveal the file `labsheet3.m` in Winzip. Double-click on this to open the file in MATLAB and then ***immediately save the file*** as `labsheet3.m` in your `MTH3011` directory.

Familiarise yourself with the grid of space and time values at which the approximate solution will be found

3. The value $n_x = 30$ will be used initially for this problem, so the `linspace` command (and the transpose operator `'`) is used in lines 6-8 to create a ***column*** vector `x` containing 31 equally-spaced values of x over $[0,3]$, with a spatial step `dx` of $\Delta x = 0.1$.

4. The value $n_t = 20$ will be used for this problem, so the `linspace` command is used in lines 10-12 to create a ***row*** vector `t` of 21 equally-spaced values of t over $[0,1]$, with a time step `dt` of $\Delta t = 0.05$.

Graph the exact solution of the advection equation as a surface plot

5. Using the exact solution given on page 1 of this sheet, create a 31×21 matrix vector `u_exact` at line 22 of the template file which contains the values of the exact solution at each (x,t). [Hint: use the values of `x(i)` and `t(k)` as well as the `SIGN` function defined in MATLAB.]

6. Modify the `mesh` command at line 37 so that it plots all of the values of `u_exact` as a surface plot over the entire (x,t) domain. Note that some initial labelling of the plot has been done in the template file, but you can modify or enhance it if you wish.

7. Save your edited m-file as `labsheet3.m` in your `MTH3011` directory then type `labsheet3` in the Command Window to run the commands in the file. Type `Ctrl-C` to stop the file at the first `pause` statement in line 42 and then inspect the plotted values of `u_exact` to see whether they seem to be sensible (bearing in mind the properties of the given exact solution).

Set up the initial and boundary conditions for the approximate solution of the advection equation

8. The 31×21 matrix `u_FTCS` will eventually contain the approximate solution of the advection equation using the FTCS method, with $u_i^k =$ `u_FTCS(i+1,k+1)`. Using the `SIGN` function that is defined in MATLAB, modify line 52 of the template file to store the 31 values of the ***initial condition*** $u(x,0)=-\text{sgn}(x-1)$ in `u_FTCS(1:31,1)` at corresponding locations to those in the vector `x`.

9. At lines 54 and 55 of the template file, use the `ones` command to store the corresponding values of both $u(0,t)$ and $u(3,t)$ so that they are given by the relevant **boundary conditions** described above.

10. Save these commands in your edited file `labsheet3.m`.

Use the forward-difference centred-space formula to generate the approximate solutions at every time step

11. At line 61 of the template file, remove the `%` sign and then modify this line so that at each appropriate value of `i` and `k` it stores the approximate values of `u_FTCS(i,k+1)` based on the recurrence relation given on page 1 of this sheet. [Notice that $\alpha = \Delta t/\Delta x$ is stored at line 46 to help simplify the formulae.]

Graph the approximate solution as a surface plot

12. Modify the `plot` and `mesh` commands at lines 68 and 74 of the template file so that they plot the exact solution in the same way as for the exact solution earlier.

13. Save your edited m-file as `labsheet3.m` in your `MTH3011` directory then type `labsheet3` in the Command Window to run the commands in the file. Type a space once and then type `Ctrl-C` to stop the file at the second `pause` statement in line 79 and then comment on the properties of the approximate solution for the FTCS solution – are the values what you expected?

14. Once your results look reasonable, proceed to the next step below – otherwise check your file to make sure that steps 8, 9 and 11 have been implemented correctly.

Modify your commands so that they calculate and plot the numerical solution using Leith's method

15. At lines 85-88 of the template file, enter the initial and boundary conditions for a new 31×21 matrix `u_Leith` that will eventually contain the approximate solution of the advection equation using Leith's method, with $u_i^k =$ `u_Leith(i+1,k+1)`. This should be done in a similar manner to steps 8-9 above, so that it contains it the 31 values of $u(x,0)=-\text{sgn}(x-1)$ in `u_Leith(1:31,1)`, the 20 values of $u(0,t)$ from the relevant boundary condition in `u_Leith(1,2:21)`, and the 20 values of $u(3,t)$ in `u_Leith(31,2:21)`.

16. Based on a similar process to that in step 12 above, modify the statement in the `for` loops at line 94 of the template file so it stores the approximate values of `u_Leith(i,k+1)` for the remaining 30×19 approximate values of the solution $u(x_i,t_k)$ at $t=t_1,\ldots,t_{20}$ for each $x_i = i\Delta x$ with $i=1,\ldots,19$.

17. Modify the `plot` and `mesh` commands at lines 101 and 107 of the template file, in the same way as earlier, so that they plot your approximate solution based on Leith's method.

18. Save your edited m-file as `labsheet3.m` in your `MTH3011` directory then type `labsheet3` in the Command Window to run the commands in the file. Type a space twice and then type `Ctrl-C` to stop the file at the third `pause` statement in line 112. Comment on the properties of the solution based on Leith's method – are the values what you expected? How do they compare to the FTCS method?

Modify your commands so they calculate and plot the numerical solution using the first-order upwind method

19. At lines 118-120 of the template file, enter the initial and boundary conditions for a new 31×21 matrix `u_upwind` that will eventually contain the approximate solution of the advection equation using the upwind-difference method, with $u_i^k =$ `u_upwind(i+1,k+1)`. This should be done in a similar manner

to previously, so that it contains it the 31 values of $u(x,0) = -\text{sgn}(x-1)$ in `u_upwind(1:31,1)`, the 20 values of $u(0,t)$ from the relevant boundary condition in `u_upwind(1,2:21)`, but note that the values of $u(3,t)$ are ***not*** stored in `u_upwind(31,2:21)` for this method – do you see why?

20. Based on a similar process to that in steps 12 and 16 above, modify the statement in the `for` loops at line 126 of the template file so it stores the approximate values of `u_upwind(i,k+1)` for the remaining 30×20 values of the approximate solution at $t = t_1, \ldots, t_{20}$ for $x_i = i\Delta x$ with $i = 1, \ldots, 19$.

21. Modify the `plot` and `mesh` commands at lines 133 and 139 of the template file, in the same way as earlier, so that they plot your approximate solution based on the upwind-difference method.

22. Save your edited m-file as `labsheet3.m` in your `MTH3011` directory then type `labsheet3` in the Command Window to run the commands in the file. Type a space three times so that the file runs all the way to the end, and then comment on the properties of the solution based on the upwind-difference method. Are the values what you expected?

23. Now compare the results for the three numerical methods with each other, and also with the exact solution. Identify some of the advantages and disadvantages of each approximate method.

Optional further exercises

1. Change the time step to $\Delta t = 0.025$ and recalculate your results for Leith's method up to $t = 1$. Comment on the differences.

2. Change the time step to $\Delta t = 0.1$ for Leith's method and calculate for 10 steps up to $t = 1$. Comment on the differences.

MAP/map
1/3/16