MATH 521

Assignment 4

February 1, 2018

Problem 1. Consider the steady advection-diffusion equation in 1D

$$\begin{aligned} au' - Du'' &= 0 \qquad \text{in }]0,1[\\ u(0) &= 0 \\ u(1) &= 1 \end{aligned}$$

where the advection velocity $a \geq 0$ and the diffusivity $D > 0$ are constant throughout the domain.

(a) Discretise the advection-diffusion problem using the $N+1$ grid points

$$x = 0,\ h,\ 2h,\ 3h,\ \ldots,\ (N-1)h,\ 1$$

(where $h = 1/N$) with the second-order consistent central difference approximation

$$u'(x) \approx \frac{u(x+h) - u(x-h)}{2h}.$$

Write down the 'big linear system' $L^h_{CD} u^h = f^h$ (the first two rows and the last row suffice).

For what range of h-values does the matrix L^h_{CD} satisfy the M-matrix criterion 'weakly chained diagonally dominant & L-matrix'?

Hint: The identity

$$|\alpha + \beta| + |\alpha - \beta| = 2\max\{|\alpha|, |\beta|\}$$

may be useful.

Solution. The matrix L^h_{CD} in the 'big linear system' $L^h_{CD} u^h = f^h_{CD}$ may be written as the sum of the advection matrix A^h_{CD} and the (negative-)diffusion matrix D^h, where

$$A^h_{CD} = \frac{a}{2h}\begin{bmatrix} 0 & 1 & & \\ -1 & 0 & 1 & \\ & & \ddots & \\ & & -1 & 0 \end{bmatrix} \qquad \text{and} \qquad D^h = \frac{D}{h^2}\begin{bmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & & \ddots & \\ & & -1 & 2 \end{bmatrix}.$$

The vector f^h_{CD} is given by all zeros except for the last entry,

$$f^h_{CD} = \left[0,\ \ 0,\ \ \ldots,\ \ 0,\ \ -\tfrac{a}{2h} + \tfrac{D}{h^2}\right]^T,$$

and so the total linear system $L^h_{CD} u^h = f^h_{CD}$ is given by

$$\begin{bmatrix} \frac{2D}{h^2} & \frac{a}{2h} - \frac{D}{h^2} & & \\ -\frac{a}{2h} - \frac{D}{h^2} & \frac{2D}{h^2} & \frac{a}{2h} - \frac{D}{h^2} & \\ & & \ddots & \\ & & -\frac{a}{2h} - \frac{D}{h^2} & \frac{2D}{h^2} \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ -\frac{a}{2h} + \frac{D}{h^2} \end{bmatrix}.$$

In order to check the requirements on h for L^h_{CD} to satisfy the M-matrix criteria, we first consider the range of h-values such that that L^h_{CD} is weakly chained diagonally dominant. In order for this to hold, we require that L^h_{CD} has the following three properties:

(i) L^h_{CD} is weakly diagonally dominant in all rows

(ii) L^h_{CD} is strongly diagonally dominant in at least one row

(iii) For all rows i_0 there exists a chain of indices $i_0 \to i_1 \to \cdots \to i_s$ to a strictly diagonally dominant row i_s such that all $a_{i_{l-1},i_l} \neq 0$ $(l = 1, ..., s)$

We will check property (i) first. Let $\alpha = a/2h \geq 0$ and $\beta = D/h^2 > 0$. For the intermediate rows, property (i) requires that

$$\begin{aligned} 2\beta &\geq |\alpha + \beta| + |\alpha - \beta| \\ &= 2\max\{|\alpha|, |\beta|\} \\ &= 2\max\{\alpha, \beta\} \end{aligned} \tag{1}$$

Since $\beta > 0$ and $\alpha \geq 0$, it must be that $\beta \geq \alpha$, or in terms of the original variables,

$$\frac{D}{h^2} \geq \frac{a}{2h}. \tag{2}$$

In other words, *diffusive effects must be at least as strong as advection effects.* Now, the conditions on the first and last rows respectively are

$$2\beta \geq |\alpha - \beta| \tag{3}$$

$$2\beta \geq |-\alpha - \beta| = |\alpha + \beta| = \alpha + \beta \tag{4}$$

The condition on the last row clearly follows from the fact that $\beta \geq \alpha$ (and that both α and β are non-negative). The condition on the first row follows afterwards from the fact that

$$|\alpha - \beta| \leq |\alpha + \beta| = \alpha + \beta \leq 2\beta.$$

Thus, condition 2 is sufficient to satisfy (i).

For condition (ii), we first note that the left-hand side of conditions 1, 3, and 4 are all the same, namely 2β. It is easy to see then that since

$$|\alpha - \beta| \leq \alpha + \beta \leq 2\max\{\alpha, \beta\},$$

the weakest requirement on L^h_{CD} would be to require the first row to be strictly diagonally dominant, i.e. require $2\beta > |\alpha - \beta|$. Since we additionally have that $\beta \geq \alpha \geq 0$, this is equivalent to $2\beta > \beta - \alpha$, or $\beta > -\alpha$. Since $\alpha \geq 0$, this is trivially satisfied by the fact that $\beta > 0$. Thus, the first row of the matrix is strictly diagonally dominant, following condition 2.

Condition (iii) is satisfied by the tridiagonal structure of L^h_{CD} if we further require that the first super-diagonal of each row is non zero. This is equivalent to requiring that, in combination with condition 2, $\beta > \alpha$. In terms of the original variables, we have that

$$\boxed{\frac{D}{h^2} > \frac{a}{2h}}. \tag{5}$$

Similarly, L^h_{CD} is clearly an L-matrix (positive diagonal entries and non-positive off-diagonal entries) by the fact that $\alpha < \beta$, $\beta > 0$, and $\alpha \geq 0$. Thus, condition 5 is sufficient to make L^h_{CD} satisfy the M-matrix criterion. Writing condition 5 as a condition on h, we require that

$$\boxed{h < \frac{2D}{a}}. \tag{6}$$

□

(b) Leaving everything else unchanged, discretise the transport term with the first-order consistent upwind differencing scheme

$$u'(x) \approx \frac{u(x) - u(x-h)}{h}$$

instead. Write down the 'big linear system' $L^h_{UD} u^h = f^h$.

For what range of h-values does the matrix L^h_{UD} satisfy the M-matrix criterion?

Solution. The advection matrix A^h_{CD} has now been modified to the upwind advection matrix A^h_{UD}, given by

$$A^h_{UD} = \frac{a}{h} \begin{bmatrix} 1 & 0 & & \\ -1 & 1 & 0 & \\ & & \ddots & \\ & & -1 & 1 \end{bmatrix}.$$

The (negative-)diffusion matrix D^h remains as before, but the right-hand side vector f^h_{UD} no longer depends on the advection process at all, and is given by

$$f^h_{UD} = \begin{bmatrix} 0, & 0, & \dots, & 0, & \frac{D}{h^2} \end{bmatrix}^T.$$

Intuitively, this is because the advection process 'transfers information from left to right', and so the right endpoint is only affected by it's neighbour through the diffusive process, as diffusion transfers information isotropically in both directions.

The new 'big linear system' $L^h_{UD} u^h = f^h_{UD}$ is now given by

$$\begin{bmatrix} \frac{2D}{h^2} + \frac{a}{h} & -\frac{D}{h^2} & & \\ -\frac{a}{h} - \frac{D}{h^2} & \frac{2D}{h^2} + \frac{a}{h} & -\frac{D}{h^2} & \\ & & \ddots & \\ & & -\frac{a}{h} - \frac{D}{h^2} & \frac{2D}{h^2} + \frac{a}{h} \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \frac{D}{h^2} \end{bmatrix}.$$

Now, it is trivial to see that L^h_{UD} is a L-matrix, as since $D > 0$ and $a \geq 0$ we see that the diagonal is strictly positive and the off diagonals are strictly negative.

Using the same notation as in problem 1(a), the condition on the middle rows to be weakly diagonally dominant is

$$2\beta + \alpha \geq |-\alpha - \beta| + |-\beta| = 2\beta + \alpha, \tag{7}$$

which is trivially satisfied.

For the first and last row respectively, we have

$$2\beta + \alpha \geq |-\beta| = \beta \tag{8}$$
$$2\beta + \alpha \geq |-\alpha - \beta| = \alpha + \beta, \tag{9}$$

both of which are trivially satisfied by the non-negativity of α and β. In fact, as $\beta > 0$, both the first and last rows are *strictly* diagonally dominant.

Therefore, we have that

(i) L^h_{UD} is weakly diagonally dominant in all rows

(ii) L^h_{UD} is strongly diagonally dominant in at least one row (both the first and the last)

(iii) L^h_{UD} is weakly chained, following the tridiagonal structure, the first row being strictly diagonally dominant, and all off-diagonal rows being non-zero (in fact, strictly negative)

Therefore L^h_{UD} is weakly chained diagonally dominant, and since L^h_{UD} is also an L-matrix, it follows that L^h_{UD} is an M-matrix.

$\boxed{\textbf{There is no restriction on the grid spacing } \boldsymbol{h}}$, as we have only assumed that $D > 0$ and $a \geq 0$.

$\square$

(c) Download the file `advection_diffusion.m` and read the output of `help advection_diffusion`. Everything else is intentionally obfuscated so that you still have to do (a) and (b) yourself! You can however use this function to check your answers for the admissible ranges of h in (a) and (b).

What do you observe if the matrix L^h is not an M-matrix? Use your knowledge from lectures and the technical language you have learnt so far to *describe* and *explain* your observations.

Solution. The expectation in the investigation of the behaviour of the solution is that if L^h is an M-matrix, we have the monotonicity property (where $\succeq$ denotes entry-wise $\geq$):

$$L^h u^h \succeq 0 \Rightarrow u^h \succeq 0.$$

We have shown that the matrix L^h_{UD} is an M-matrix for all h, and the matrix L^h_{CD} is an M-matrix only for $h < 2D/a$. Now, in considering the right-hand side vector f^h_{CD} and f^h_{UD}, we note additionally that

$$f^h_{CD} \succeq 0 \Leftrightarrow h < \frac{2D}{a} \qquad \text{and} \qquad f^h_{UD} \succeq 0 \quad \forall h > 0.$$

Therefore, the solutions computed using the upwind differentiation scheme should be strictly positive for any grid size h. For the centred scheme, however, there is no such guarantee when $h \geq 2D/a$. This is due to neither the matrix L^h_{CD} being an M-matrix in this case, nor the right-hand side vector f^h_{CD} being non-negative.

Thus, although we expect the L^h_{CD} system to converge to the true solution faster as $h \to 0$ (second order in h vs. first order), we have no guarantees that the solution will behave well for $h \geq 2D/a$. The solution will likely be erratic and non-physical.

Lastly, we expect the elliptic operator L^h to obey the elliptic maximum/minimum principle when L^h is an M-matrix, but this is not guaranteed when L^h is not an M-matrix.

The script in Appendix A was used to investigate the behaviour of the respective solutions for varying values of h both above and below the critical value $2D/a$. As expected, the main results were as follows:

- The solution to the upwind system $L^h_{UD} u^h = f^h_{UD}$ was strictly positive for all grid sizes
- The solution to the centred system $L_h CD u^h = f^h_{CD}$ was positive only for $h < 2D/a$, and was non-negative for $h \leq 2D/a$
- The centered system solution was additionally highly erratic for $h > 2D/a$, while the upwind system was smooth for all h
- As $h \to 0$, the centred system converged faster than the upwind system
- Both system obey the elliptic maximum/minimum principles when the L^h matrices satisfy the M-criterion, but as solutions to to the centred system produce negative values for $h < 2D/a$, the elliptic minimum principle is clearly not satisfied in this case.

□

(d) Even though the upwind differencing scheme is only first-order consistent as an approximation of the equation

$$au' - Du'' = 0$$

it is second-order consistent as an approximation of a slightly different equation,

$$au' - (D + \tilde{D})u'' = 0,$$

provided that $u \in C^4([0, 1])$. Calculate this number $\tilde{D}$.

Hint: Use Taylor expansions as done in class or video #7 to determine the truncation error

$$au'(x) \quad - \quad \text{(upwind difference approximation of this term).}$$

$\tilde{D}$ will depend on h.

Solution. Given $u \in C^4([0, 1])$, we can expand $u(x \pm h)$ to third and fourth order taylor series in h as

$$u(x \pm h) = u(x) \pm hu'(x) + \frac{h^2}{2}u''(x) \pm \frac{h^3}{6}u'''(\zeta_\pm) \tag{10}$$

$$u(x \pm h) = u(x) \pm hu'(x) + \frac{h^2}{2}u''(x) \pm \frac{h^3}{6}u'''(x) + \frac{h^4}{24}u''''(\xi_\pm) \tag{11}$$

where ξ_+ and ζ_+ are in $[x, x + h]$, and ξ_- and ζ_- are in $[x - h, x]$.

Now, we consider the error between $au' - (D + \tilde{D})u''$ and the upwind differencing scheme for $au' - Du''$. Namely, we have that

$$\begin{aligned}
&\left| au'(x) - (D + \tilde{D})u''(x) - \left(a\frac{u(x) - u(x - h)}{h} - D\frac{u(x - h) - 2u(x) + u(x + h)}{h^2} \right) \right| \\
&= \left| a\left(u'(x) - \frac{u(x) - u(x - h)}{h} \right) - D\left(u''(x) - \frac{u(x - h) - 2u(x) + u(x + h)}{h^2} \right) - \tilde{D}u''(x) \right| \\
&= \left| a\left(\frac{h}{2}u''(x) - \frac{h^2}{6}u'''(\zeta_-) \right) + D\left(\frac{h^2}{24}(u''''(\xi_+) + u''''(\xi_-)) \right) - \tilde{D}u''(x) \right| \\
&= \left| \left(\frac{ah}{2} - \tilde{D} \right)u''(x) + h^2\left(\frac{D}{24}(u''''(\xi_+) + u''''(\xi_-)) - \frac{a}{6}u'''(\zeta_-) \right) \right| \\
&\le h^2 \max_{[x-h, x+h]} \frac{a}{6}\left| u''' \right| + \frac{D}{12}\left| u'''' \right|
\end{aligned}$$

where in the last line we have chosen $\tilde{D} = ah/2$ and used the triangle inequality to obtain an upper bound on the error in terms of the third and fourth derivatives on the whole interval.

Thus, the upwind differencing scheme is second-order consistent for the equation

$$au' - (D + \tilde{D})u'' = 0$$

if we choose $\boxed{\tilde{D} = \frac{ah}{2}}$.

□

Your Primer Talk Primer talk information:

- **I do not plan on using the document camera**, I plan on preparing a short slideshow presentation.
- By the end of this talk, the audience should understand what the Bloch-Torrey equation is and how it's solution is both useful to the MRI community in terms of modelling MR signals, and that solving it in realistic geometries is difficult due to the multiple scales of the problem.

Your Learning Progress *What is the one most important thing that you have learnt in this assignment?*

Question 1c was definitely the most illuminating part of this assignment. I knew intuitively that something like this must have been true, but I did not know before this the precise relation between stable solutions and the M-matrix criterion of the linear system and how vital it is to having well behaved solutions.

What is the most substantial new insight that you have gained from this course this week? Any aha moment?

The fact that first-order differencing schemes can be preferred over second order schemes on physical and mathematical grounds was definitely an *aha moment*! I have heard about upwind differencing being extremely important for stability in things like fluid equations, but I never knew precisely why until now!

Appendix A

```
% Investigating the effect of the step size h on the solution of the
% advection-diffusion equation in 1D using upwind and centered difference
% schemes for the first derivative
close all force; clear all; clc

%% Declare test variables
Ncrit = 128; % Ncrit = 1/hcrit = a/2D
D = pi; % Choose D arbitrarily
a = 2*D*Ncrit; % choose 'a' such that hcrit = 2D/a = 1/Ncrit above

% Set the range of N values to be = [4, 8, 16, ..., Ncrit, ..., Ncrit^2]
% (Ncrit is assumed to have been chosen to be a power of 2)
pmin = 2;
pmax = 2*round(log2(Ncrit));
Nrange = 2.^(pmin:pmax);

%% Solve the system below and above the critical point hcrit = 2D/a
for N = Nrange
    advection_diffusion(a,D,N);
    drawnow; title(sprintf('N = %d',N));
end
```

Appendix B

```
function advection_diffusion(a,D,N)
%ADVECTION_DIFFUSION computes and plots the solution of the advection-
%diffusion equation
%       a u' - D u" = 0 on ]0,1[
%       u(0) = 0
%       u(1) = 1
%a is a real number, D a positive number and N is the (integer) number of
%subintervals for the mesh.
%
%Example of use:
%advection_diffusion(10,1,25)

xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx = 1/N;
xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx = N;
xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx = 2;
xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx = 1;
xxxyxxyyxyxyxyxxxxxxyxyxyyyxyxxyxyyxxxx = 0;
xxxyxxyyxyxyxyxxxxxxyxyyyxyxxyxxyxyyxxxx = 1000;
xxxyxyxyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx = ...
    linspace(xxxyxxyyxyxyxyxxxxxxyxyxyxyyxyxxyxyyxxxx,...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx+...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)';
xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx = ...
    xxxyxxyyxyxyxyxxxxxxyxyxyyyxyxxyxyyxxxx*...
    ones(xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)/...
    xxxyxxyyxyxyxyxxxxxxyxyyyxyxxyxxyxyyxxxx - ...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx + ...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx;
xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx = a;
xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx = D;
xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx = ...
    xxxyxxyyxyxyxyxxxxxxyxyxyyyxyxxyxyyxxxx.*...
    xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx;
xxxyxxyxxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx = ...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx*...
    spdiags([-xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx ...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx*...
    xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx -...
    xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx],...
    -xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx:...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)./...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx^...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx;
```

```
xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx(end) =...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx(end) ...
    + xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx/...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx^...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx;
xxxyxxyxxyxyxyxxxxxxyxyxyxyxxyxxyxyxyxxx = ...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx; ...
    xxxyxxyxxyxyxxyxxxxxyxyxyxyxxyxxyxyxyxxx = ...
    xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx*...
    spdiags([-xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx ...
    xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx],...
    [-xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx ...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyyxyxxyxyyxxxx],...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)/...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx;
xxxyxxyxxyxyxxyxxxxxyxyxyxyxxyxxyxyyxxxx = ...
    xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx*...
    spdiags([-xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx ...
    xxxyxyyyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx],...
    [-xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx ...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx],...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyxxxxx-...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)/...
    (xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx...
    *xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx);
xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx(end) = ...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx(end) - ...
    xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx/...
    (xxxyxyxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx*...
    xxxyxyxyxyxyxyxxxxxxyxyxyxyxxxyxyxyxxxxx);
xxxyxxxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx = ...
    linspace(xxxyxxyyxyxyxyxxxxxxyxyxyxyyxyxxyxyyxxxx,...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxxyyxyxyxyxxxxxxyxyyyxyxxyxxyxyyxxxx);
xxxyxxxyxyxyxyxxxxxxyxyxyxyxyxxxyxyyxxxx = ...
    (exp(xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx/...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx*...
    xxxyxxxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx) - ...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)/...
    (exp(xxxyxyyyxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx/...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyxyxyxxxxx) - ...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx);
figure;
plot(xxxyxyxyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx,...
    [xxxyxxyyxyxyxyxxxxxxyxyxyxyyxyxxyxyyxxxx;...
    (xxxyxxyxxyxyxxyxxxxxyxyxyxyxxyxxyxyxyxxx+...
    xxxyxxyxxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)\...
    xxxyxxyxxyxyxyxxxxxxyxyxyxyxxyxxyxyxyxxx;...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx],'o-',...
```

```
    xxxyxyxyxyxyxyxxxxxxyxyyxxyxxxyxyxyxxxxx,...
    [xxxyxxyyxyxyxyxxxxxxyxyxyxyyxyxxyxyyxxxx;...
    (xxxyxxyxxyxyxxyxxxxxyxyxyxyxxyxxyxyyxxxx+...
    xxxyxxyxxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx)\...
    xxxyxyyxxyxyxyxxxxxyxxyyxxyxxxyyyxyxxxxx;...
    xxxyxxyyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx],'s-',...
    xxxyxxxyxyxyxyxxxxxxyxyxyxyxxyxxyxyyxxxx,...
    xxxyxxxyxyxyxyxxxxxxyxyxyxyxyxxxyxyyxxxx);
xlabel('{\itx}')
ylabel('{\itu}')
legend('upwind differencing','central differencing',...
    'analytical solution','location','northwest');
```