Amath 586: Take-home Final

Due on June 11, 2015

Brian de Silva

Problem 1

Consider the method

$$U_j^{n+1} = U_j^n - \frac{ak}{2h}(U_j^n - U_{j-1}^n + U_j^{n+1} - U_{j-1}^{n+1}). \tag{1}$$

for the advection equation $u_t + au_x = 0$ on $0 \leq x \leq 1$ with periodic boundary conditions, on a grid with spacing $h = 1/(m+1)$ for some integer m.

(a) This method can be viewed as the trapezoidal method applied to an ODE system $U'(t) = AU(t)$ arising from a method of lines discretization of the advection equation. What is the matrix A? Don't forget the boundary conditions.

(b) Determine the eigenvalues of the matrix A. **Hint:** the eigenvectors are the same as for similar matrices you have seen.

(c) Suppose we want to fix the Courant number ak/h as $k,\ h \to 0$. For what range of Courant numbers will the method be stable if $a > 0$? If $a < 0$? Justify your answers in terms of eigenvalues of the matrix A and the stability region of the trapezoidal method.

(d) For what range of ak/h will the CFL condition be satisfied for this method (with periodic boundary conditions)? Consider both $a > 0$ and $a < 0$. Is satisfying the CFL condition sufficient for stability in all cases?

Solution:

(a) Consider the following ODE system:

$$U'(t) = -\frac{a}{2h}(U_j(t) - U_{j-1}(t)).$$

If we apply the trapezoidal method to this system we obtain (1). With periodic boundary conditions, this suggests that we can write this system as $U'(t) = AU(t)$ with A given by

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

(b) It is easy to check that if $\mathbf{u}^p = (e^{2\pi iph}, e^{4\pi iph}, \ldots, e^{2(m+1)\pi iph})^T$ then

$$A\mathbf{u}^p = -\frac{a}{h}\left(\frac{e^{2\pi iph} - 1}{e^{2\pi iph}}\right)\mathbf{u}^p, \qquad p = 1, 2, \ldots, m+1.$$

Thus $\mathbf{u}^p$ are the eigenvalues of A with eigenvalues $\lambda_p = -\frac{a}{h}\left(\frac{e^{2\pi iph}-1}{e^{2\pi iph}}\right), p = 1, 2, \ldots, m+1$. We can rewrite this expression in what is perhaps a more convenient form:

$$\lambda_p = -\frac{a}{h}(1 - e^{-2\pi iph}) = -\frac{a}{h}(1 - \cos(2\pi ph) + i\sin(2\pi ph)), \qquad p = 1, 2, \ldots, m+1.$$

(c) Let $c = ak/h$, the Courant number. Recall that for the trapezoidal method the expression dictating stability is given by

$$R(z) = \frac{1+\frac{1}{2}z}{1-\frac{1}{2}z}.$$

For a fixed value of k the method is stable in the above problem if $|R(k\lambda^p)| \leq 1$ for every eigenvalue λ^p, and is unstable otherwise. Therefore we need $|1+\frac{1}{2}k\lambda^p| \leq |1-\frac{1}{2}k\lambda^p|$ in order for the method to be stable. Substituting in our expression for λ^p gives the following condition for stability:

$$\begin{aligned} |1+c(1-\cos(2\pi ph)+i\sin(2\pi ph))/2| \;&\leq\; |1-c(1-\cos(2\pi ph)+i\sin(2\pi ph))/2| \\ \Longrightarrow (1+c(1-\cos(2\pi ph))/2)^2 \;&\leq\; (1-c(1-\cos(2\pi ph))/2)^2 \\ \Longrightarrow |1+c(1-\cos(2\pi ph))/2)| \;&\leq\; |1-c(1-\cos(2\pi ph))/2|, \qquad p = 1,2,\ldots,m+1. \end{aligned}$$

Notice that $1-\cos(2\pi ph) \leq 1$ for each $p = 1,2,\ldots,m+1$. Hence if $c > 0$ (i.e. $a > 0$), $c(1-\cos(2\pi ph))/2 \leq 0$ so the stability condition is satisfied for any positive value of k. If $c < 0$ (equivalently $a < 0$) then $c(1-\cos(2\pi ph))/2 \geq 0$ so the stability condition cannot be satisfied. We conclude that the method is stable for all $c > 0$ and unstable for all $c < 0$.

(d) Using method (1), the approximation U_j^{n+1} depends on the points U_j^{n+1}, U_{j-1}^{n+1}, U_j^n, and U_{j-1}^n. Because of the dependence on points at time t_{n+1} (i.e. because the method is explicit) as the grid is refined the numerical domain of dependence of a point (X,T) approaches $(-\infty, X]$. Since we have periodic boundary conditions on $[0,1]$ in this case, the numerical domain of dependence converges to the entire interval $[0,1]$ as $k,h \to 0$. The domain of dependence of the PDE is clearly always contained in the numerical one. Therefore the CFL condition is always satisfied, regardless of the sign of a. So knowing that the CFL condition is satisfied does not actually provide us with any information.

If $a > 0$ the method is stable, as we saw in part (c), and the CFL condition is automatically satisfied, so in this case satisfying the CFL condition is sufficient for stability. For $a < 0$ the CFL condition is satisfied, but the method is unstable, so satisfying the CFL condition is insufficient for stability.

Problem 2

Consider the dispersive PDE $u_t + \gamma u_{xxx} = 0$ with periodic boundary conditions on an interval $a \leq x \leq b$.

(a) The third-derivative term could be approximated by either

$$u_{xxx}(x_j,t_n) \approx \frac{1}{h^3}(-U_{j-2}^n + 3U_{j-1}^n - 3U_j^n + U_{j+1}^n) \tag{2}$$

or by

$$u_{xxx}(x_j,t_n) \approx \frac{1}{h^3}(-U_{j-1}^n + 3U_j^n - 3U_{j+1}^n + U_{j+2}^n) \tag{3}$$

Show that either one is first-order accurate.

(b) Suppose you plan to use a trapezoid discretization in time for a problem where $\gamma > 0$. Which form of spatial discretization (2) or (3) would be better? Justify your answer.

(c) Implement a Crank-Nicolson type method for this equation, which uses the trapezoid method in time together with the spatial discretization from above that you determined is best for the case $\gamma > 0$. Your implementation should allow arbitrary values of the endpoints a and b of the spatial interval, since two different choices are required below.

Test your method by showing it is first order accurate for initial data $u(x,0) =$ $textsin(px)$ for integer p such as $p = 2$ on the interval $0 \leq x \leq 2\pi$. Note that you can compute the exact solution for such data.

Problem 2 continued on next page...

Solution:

(a) Let u denote $u(x,t)$. We use the following Taylor expansions:

$$\begin{aligned} u(x+2h,t) &= u + 2hu_x + 2h^2 u_{xx} + \frac{4h^3}{3}u_{xxx} + \frac{16h^4}{24}u_{4x} + O(h^5), \\ u(x+h,t) &= u + hu_x + \frac{h^2}{2}u_{xx} + \frac{h^3}{6}u_{xxx} + \frac{h^4}{24}u_{4x} + O(h^5), \\ u(x-h,t) &= u - hu_x + \frac{h^2}{2}u_{xx} - \frac{h^3}{6}u_{xxx} + \frac{h^4}{24}u_{4x} + O(h^5), \\ u(x-2h,t) &= u - 2hu_x + 2h^2 u_{xx} - \frac{4h^3}{3}u_{xxx} + \frac{16h^4}{24}u_{4x} + O(h^5). \end{aligned}$$

Substituting these into the right-hand side of (2) and grouping terms in of powers of h gives

$$\frac{1}{h^3}\left((3-3)u + h(1-3+2)u_x + h^2\left(\frac{1}{2}+\frac{3}{2}-2\right)u_{xx} + h^3\left(\frac{1}{6}-\frac{1}{2}+\frac{4}{3}\right)u_{xxx} + h^4\left(\frac{1}{24}+\frac{3}{24}-\frac{16}{24}\right)u_{4x} + O(h^5)\right)$$
$$= u_{xxx} - \frac{h}{2}u_{4x} + O(h^2).$$

Carrying out the same procedure for the right-hand side of (3) yields

$$\frac{1}{h^3}\left((3-3)u + h(2-3+1)u_x + h^2\left(2-\frac{3}{2}-\frac{1}{2}\right)u_{xx} + h^3\left(\frac{4}{3}-\frac{1}{2}+\frac{1}{6}\right)u_{xxx} + h^4\left(\frac{16}{24}-\frac{3}{24}-\frac{1}{24}\right)u_{4x} + O(h^5)\right)$$
$$= u_{xxx} + \frac{h}{2}u_{4x} + O(h^2).$$

Hence both methods give first order approximations to u_{xxx}.

(b) Notice that a method based on (2) draws more information from points on the left and (3) from points on the right. If $u(x,0) = u_0(x)$ then we have

$$\begin{aligned} & u_t = -\gamma u_{xxx} \\ \implies & \hat{u}_t = \gamma i \xi^3 \hat{u} \\ \implies & \hat{u} \ = e^{i\gamma\xi^3 t}\hat{u}_o. \end{aligned}$$

Taking the inverse Fourier transform gives

$$u(x,t) = \frac{1}{\sqrt{2\pi}}\int_{\infty}^{\infty} \hat{u}_o(\xi)e^{i\xi(x+\gamma\xi^2 t)}d\xi.$$

Hence the Fourier component corresponding to wavenumber ξ moves with velocity $-\gamma\xi^2$. It follows that solutions of the PDE will move to the left if $\gamma > 0$ and to the right if $\gamma < 0$. Since the trapezoidal method is absolutely stable and gives rise to a method which has a numerical domain of dependence including the entire interval $[a,b]$, we do not need to worry as much about stability and the CFL condition being satisfied when choosing between (2) and (3). Instead we should choose the scheme better suited to the PDE. In this case a numerical method using (3) is preferable since it samples more points to the right, which is consistent with our expectation that information will be travelling to the left as time evolves.

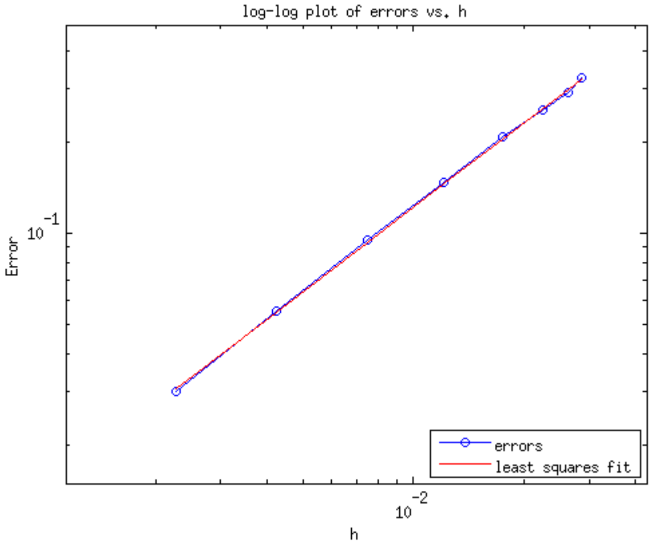


Figure 1: Error at $t = 1$ as h and k are refined for a method based on (3)

(c) Assuming a solution of the form $u(x, t) = \sin(px + a\gamma t)$ and substituting into the PDE to solve for a gives $u(x, t) = \sin(px + p^3\gamma t)$ as an exact solution.

We fix $p = 2$ and $\gamma = 1$ and approximate the solution from $t = 0$ to $t = 1$ on the interval $0 \leq x \leq 2\pi$. Since (3) is first order accurate and the trapezoidal method is second order accurate it is reasonable to expect the method obtained by combining the two is $O(k^2 + h)$ accurate. Hence we fix $k = \sqrt{h}/2$ as we refine the grid to create the error plot in Figure (1).

The numerically approximated order of accuracy is 0.93. If we relax the condition on k and allow $k = O(h)$ then the accuracy improves substantially. By modifying the relationship between h and k we obtain approximate orders of accuracy ranging from 0.88 to 1.63.

Just to check that we were correct in choosing to base our method off (3) rather than (2) we have computed the error using all the same parameters, but using the alternate method. The results are presented in Figure (2).

The code used to run the numerical trials can be found in the appendix.

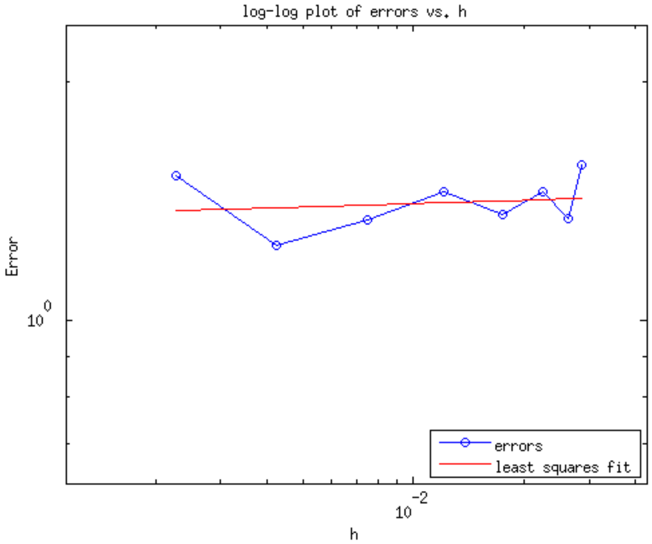


Figure 2: Error at $t = 1$ as h and k are refined for a method based on (2)

Problem 3

Now consider the KdV equation $u_t + 6uu_x + u_{xxx} = 0$. This Korteweg - de Vries equation is famous — read about it if you are not familiar with it. Even though this equation is nonlinear, there are exact "solitary wave" solutions that consist of a translating wave of the form $u(x,t) = \eta(x - \alpha^2 t)$ for special initial data of the form

$$\eta(x) = \frac{\alpha^2}{2}\operatorname{sech}^2(\alpha x/2) \tag{4}$$

for any choice of parameter $\alpha > 0$. Note that taller pulses propagate more rapidly. An amazing feature of such problems is the way two such pulses interact as "solitons".

Construct a first-order accurate method by combining your solution to Problem 2 with a first-order upwind method for the hyperbolic equation

$$u_t + 3(u^2)_x = 0. \tag{5}$$

Use the first-order fractional step approach to combine the methods — i.e. in each time step, first advance the solution by taking a step with the $u_t + u_{xxx} = 0$ to get an intermediate solution U^*, and then use this as initial data for a step in which you solve (5) by the upwind method for this nonlinear conservation law.

Again use periodic boundary conditions. The true solution is not periodic, but decays exponentially to zero so as long as the interval is long enough relative to the width of the solitary wave, so this should not affect errors computed using a periodic extension of (4).

Test your method on the interval $-1 \leq x \leq 3$ with $\alpha = 10$ out to time $t = 0.02$. Demonstrate that your method is stable and first-order accurate by refining the grid with $r = k/h$ fixed.

Estimate the maximum stable value of r based on stability theory, justifying your estimate, and confirm that this is consistent with what you observe from your numerical method.

Note that this is not a very accurate method! If you want to play around with soliton interaction (optional) you might try the following data:

$$\eta(x) = \frac{\alpha_1^2}{2}\operatorname{sech}^2(\alpha_1(x+2)/2) + \frac{\alpha_2^2}{2}\operatorname{sech}^2(\alpha_2(x+1)/2) \tag{6}$$

with $\alpha_1 = 25,\ \alpha_2 = 16$ on the interval $-3 \leq x \leq 2$.

Problem 3 continued on next page. . .

Solution:
It is stated in the text that if one can show $\|\mathcal{N}_A(U,k)\| \leq \|U\|$ and $\|\mathcal{N}_B(U,k)\| \leq \|U\|$ (where the same norm is used in both instances) then one can demonstrate the stability of a fractional step method using $\mathcal{N}_A$ and $\mathcal{N}_B$ as its first and second steps. It also alerted us to the possibility that the stability of the two methods independently may not be enough to guarantee a fractional step method which is stable. However the task of proving the norm condition proved to be too daunting for the author, so keeping the warning in mind, we cautiously attempt to see whether enforcing the separate stability conditions from each method can inform our numerical tests to determine $r = k/h$.

As we argued in problem 2 we are not very worried about the stability of our solution to problem 2 since it always satisfies the CFL condition and has very good stability properties due its use of the trapezoidal method, at least for $\gamma > 0$. In this particular instance it is used in a fractional step method to solve the equation $u_t + u_{xxx} = 0$, in which case $\gamma = 1 > 0$. Hence we expect it to behave well.

We saw in class that the stability of the upwind method agrees with the restriction imposed by the CFL condition. Thus if we can determine the CFL condition for the hyperbolic equation being solved by the upwind method at each step, $u_t + 3(u^2)_x = 0$, then we will be able to get a rough estimate of the maximal value r can take. First note that the PDE can be rewritten

$$u_t + 6uu_x = 0$$

which we recognize as Burger's equation. This simple quasilinear PDE can be solved using the method of characteristics. If we use $\eta(x)$ for the initial data we obtain

$$u(x,t) = \eta(\xi)$$

where ξ is given implicitly by

$$\xi = x - 3\alpha^2 \text{sech}^2(\alpha\xi/2)t.$$

From the form of the solution we can see that the initial data is simply propagated to the right at varying speeds determined by $3\alpha^2\text{sech}^2(\alpha\xi/2)$. Since $0 \leq \text{sech}(x) \leq 1$ we can infer that the fastest moving data will move to the right with speed $3\alpha^2 = 300$ (for $\alpha = 10$). In fact this occurs for the initial data corresponding to $\xi = 0$. Therefore, at worst, the solution at a point (X,T) has domain of dependence $X - 300T$. For other values of ξ the domain of dependence of a point (X,T) will be contained in the interval $[X - 300T, X]$. Since the numerical domain of dependence fills in the interval $[X - T/r, X]$ as the grid is refined, we require $300 \leq 1/r$ or $r \leq 1/300$.

This restriction on r agrees well with the numerical test. As we see in Figure (3) for $r = 1/300$ the method exhibits first order accuracy (approximate order of accuracy: 0.87). If we leave off the two data points corresponding to the coarsest grids the approximate order of accuracy increases to 0.91. If we allow r to be slightly larger, say 1/275, the method appears to be stable for smaller numbers of time steps, but as the grid is refined it eventually becomes unstable and the approximate solution blows up. It was difficult for me to test numerically exactly where the threshold lies due to the time needed to solve the problem on very fine grids. The largest value of r for which I numerically verified instability was $r = 1/277$. The method remained stable even for large numbers of time steps when r was taken to be 1/300. Note that the initial data fed into the upwind method is actually the result of applying the method devised in problem 2 to the actual initial data $\eta(x)$, so we do not expect the analytic upper bound on r to exactly match the numerical one. In any case, the numerical results agree with the restriction found above.

The code used to run the numerical trials can be found in the appendix.

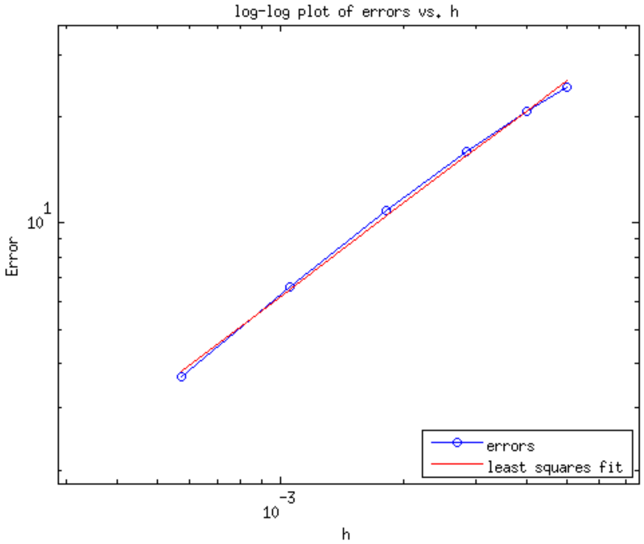


Figure 3: Error at $t = 0.02$ as h and k are refined with $k = 300h$

Appendix: Code

Amath 586 Take-home Final Matlab Notebook

Table of Contents

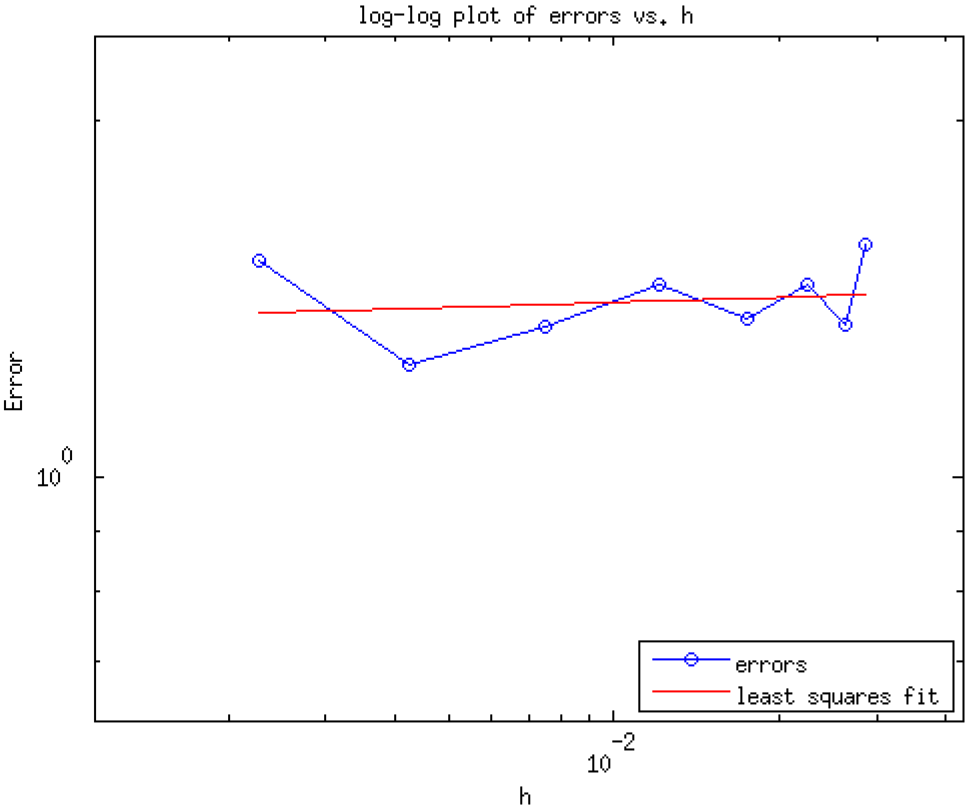
Brian de Silva

Problem 2 (c)

```
% Throughout we use p=2

numTrials =8;

% Create vectors in which to store h values and error
hVec = zeros(numTrials,1);
errorVec = zeros(numTrials,1);

% Approximate PDE using an increasing number of grid points (200,220,260,320,...)
% with k=sqrt(h)/2
for j=1:numTrials
    [hVec(j),~,errorVec(j)] = dispersion_modified_CN(200+20*2^(j-1) - 1,0,2*pi);
end

% Create loglog plot of error as grid is refined
error_loglog(hVec,errorVec);

% Computed order of accuracy: 0.93
% Note: Choosing different relationships between h and k gave approximate
% orders of accuracy ranging from 0.88-1.63.
% Overall this suggests first order accuracy

% Aside
%----------------------------------------------------------
% Just to be sure we made the right choice, let's try the method
% corresponding to using equation (2) instead of (3):

% Create vectors in whic to store h values and error
hVec = zeros(numTrials,1);
errorVec = zeros(numTrials,1);

% Approximate PDE using an increasing number of grid points (200,220,260,320,...)
% with k=sqrt(h)/2
for j=1:numTrials
    [hVec(j),~,errorVec(j)] = dispersion_modified_CN_worse(200+20*2^(j-1) - 1,0,2*
end
```

```
% Create loglog plot of error as grid is refined
error_loglog(hVec,errorVec);
%-----------------------------------------------------------
```

```
        Least squares fit gives E(h) = 8.80005 * h^0.929562


        Least squares fit gives E(h) = 1.50299 * h^0.0145533
```

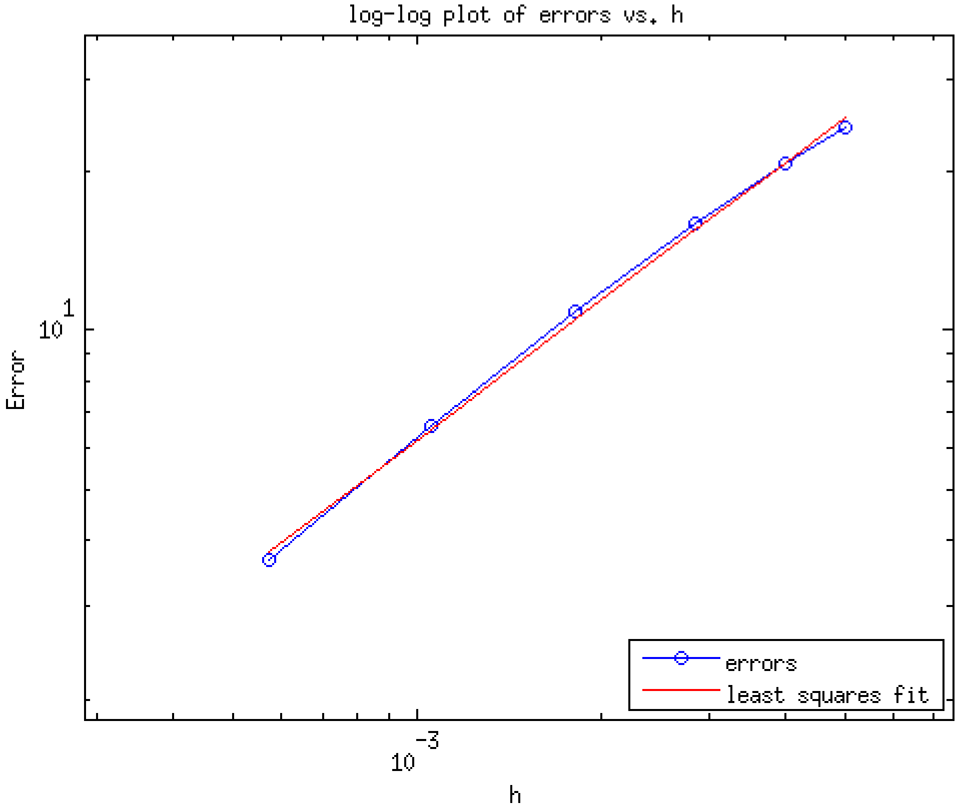


Problem 3

```
clear all

numTrials = 6;

% Create vectors in which to store h values and error
hVec = zeros(numTrials,1);
errorVec = zeros(numTrials,1);
tic
% Approximate PDE using an increasing number of grid points (600,800,1200,... )
% with k=h/300
for j=1:numTrials
    [hVec(j),~,errorVec(j)] = kdv_modified_CN(600+200*2^(j-1) - 1,-1,3);
    if isnan(errorVec(j))
```

```
        disp('NaN encountered')
        break
    end
end
toc
% Create loglog plot of error as grid is refined
error_loglog(hVec,errorVec);

% Computed order of accuracy: 0.91
```

```
Elapsed time is 210.612351 seconds.


Least squares fit gives E(h) = 2573.22 * h^0.87287
```



Published with MATLAB® 7.14

586 Take – homeFinal

Brian de Silva

Problem 2

```
(* Automate  single variable Taylor expansion to fourth order *)
U[j_] :=
    u + j * h * u_x + ((j * h) ^ 2) * u_xx / 2 + (1 / 6) * u_xxx (j * h) ^ 3 + (1 / 24) * u_xxxx (j * h) ^ 4 + hots2;

(* Check answer in 2 (a) *)
eqn2 = Simplify [(1/h^3) * (-U[-2] + 3 U[-1] - 3 U[0] + U[1])]
eqn3 = Simplify [(1/h^3) * (-U[-1] + 3 U[0] - 3 U[1] + U[2])]
```

$$u_{xxx} - \frac{h\,u_{xxxx}}{2}$$

$$u_{xxx} + \frac{h\,u_{xxxx}}{2}$$