Numerical Solutions of PDEs Project

R. L. Herman

April 23, 2021

Part I - Linear Equations

In class we first looked at numerically solving the linear advection equation,

$$u_t + cu_x = 0, \quad u(x,0) = f(x),$$

using finite difference schemes. Applying forward differences in both space and time, we obtained the scheme

$$\begin{aligned} u_{j,n+1} &= u_{j,n} - \alpha(u_{j+1,n} - u_{j,n}), \\ u_{j,0} &= f(x_j), \end{aligned} \tag{1}$$

where $\alpha = \frac{c\Delta t}{\Delta x}$. This scheme proved to be unconditionally unstable.

We then used a backward difference approximation in time to obtain

$$\begin{aligned} u_{j,n+1} &= (1-\alpha)u_{j,n} + \alpha u_{j-1,n}, \\ u_{j,0} &= f(x_j), \end{aligned} \tag{2}$$

Prob. 1. A MATLAB implementation of this scheme is given in Table 1. Copy this code into MATLAB and call the file **advec1.m**. Run the program to make sure it runs and then change the number of time steps to determine when the scheme becomes unstable. Record alpha. [Note, typing alpha in the command window will give its value.] Also, you can play the save movie by typing **movie(M)** in the command window.

Prob. 2. We had used fixed boundary conditions in the above, but periodic boundary conditions ($u(a,t) = u(b,t)$) are sometimes employed. How would you change the line **u(1,n+1)=0;** so as to have periodic boundary conditions? Implement these conditions and discuss your observations.

Prob. 3. The Leapfrog scheme is given by

$$u_{j,n+1} = u_{j,n-1} - \alpha(u_{j+1,n} - u_{j-1,n})$$

Answer the following and carry out the scheme in MATLAB as indicated.

1. What type of approximations of the derivatives were used to obtain this scheme?

```
% First Order - Advection Backward Difference in Time

clear
% Specify grid
a=-5;
b=10;
Tmax=2;
Nx=50;
Nt=25;
dx=(b-a)/Nx;
dt=Tmax/Nt;
x=linspace(a,b,Nx);
c=2;
alpha=c*dt/dx;

% Initial profile
u0=exp(-(x-.5).^2);

u=zeros(Nx,Nt);
u(:,1)=u0;

plot(x,u0)
M(1)=getframe;

hold

for n=1:Nt-1;
    for j=2:Nx
        u(j,n+1)=(1-alpha)*u(j,n)+alpha*u(j-1,n);
    end
    u(1,n+1)=0;
    plot(x,u(:,n+1))
    M(n+1)=getframe;
end
hold
```

Table 1: Numerical solution of the linear advection equation using backward difference in time and forward difference in space.

2. What is α?
3. Determine the leading or truncation error terms.
4. Modify the code in Table 1 to implement the Leapfrog scheme. Change the the number of time steps to determine if the scheme becomes unstable. The scheme takes the form

```
 for j=2:Nx
     u(j,n+1)=u(j,n-1)-alpha*(u(j+1,n) - u(j-1,n));
 end
```

   This scheme cannot be started using the initial condition alone. One also needs to find the solution at $t = \Delta t$. This can be accomplished by using a previous scheme for one time step such as

```
for j=2:Nx
    u(j,2)=u(j,1)-alpha*(u(j,1) - u(j-1,1));
end
u(1,2)=0;
```

Prob. 4. Consider the Linearized KdV equation,

$$u_t + au_x + u_{xxx} = 0.$$

1. Create a scheme to solve the linearized KdV. Try a forward difference in time and centered difference in space for the first derivatives and

$$u_{xxx} \approx \frac{u_{j+2,n} - 2u_{j+1,n} + 2u_{j-1,n} - u_{j-2,n}}{2(\Delta x)^3}.$$

2. Find the local truncation error for this scheme.
3. Find the stability criterion for this scheme.
4. Modify the MATLAB advection file to numerically solve the linearized KdV using periodic boundary conditions.

Part II - KdV Solitons Solutions

We are now ready to tackle the nonlinear KdV equation. Zabusky and Kruskal (1965, Phys. Rev. Lett. 15, 240) numerically investigated the KdV equation

$$u_t + uu_x + \delta^2 u_{xxx} = 0$$

with initial condition $u(x, 0) = \cos \pi x$, $0 \leq x \leq 2$, and $\delta = 0.022$. The initial condition and emergence of solitons are shown in Figure 1. In this part of the project you will take the ZK scheme and test it on several initial conditions.

It is well known that the KdV equation has soliton solutions of the form $u(x, t) = A \operatorname{sech}^2(kx - \omega t - \eta_0)$, where $A = 2k^2$, $\omega = 4k^3$ and η_0 is a constant. Other methods have also been used to study. A recent analysis of several schemes is given in *Applying Explicit Schemes to the Korteweg-de Vries*

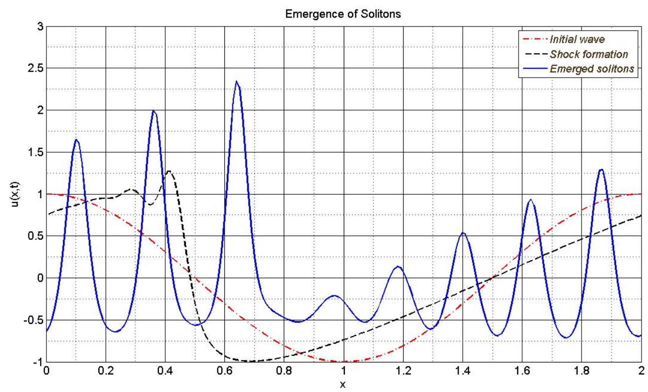

Figure 1: Zabusky and Kruskal demonstration of recurrence.

Equation by Masitah Shahrill, Maureen Siew Fang Chong, Hajah Norhakimah Haji Mohd Nor. See `www.ccsenet.org/journal/index.php/mas/article/download/46132/24929`. The Zabusky-Kruskal scheme is provided in Equation (4.10) in that paper.

$$\begin{aligned} u_n^{m+1} &= u_n^{m-1} - \frac{1}{3}\frac{\Delta t}{\Delta x}\left(u_{n+1}^m + u_n^m + u_{n-1}^m\right)\left(u_{n+1}^m - u_{n-1}^m\right) \\ &\quad - \frac{\delta^2 \Delta t}{(\Delta x)^3}\left(u_{n+2}^m - 2u_{n+1}^m + 2u_{n-1}^m - u_{n-2}^m\right), \quad n = 0, 1, \ldots, N. \end{aligned} \tag{3}$$

This is carried out using periodic boundary conditions.

Use the above scheme to answer the following:

a. Determine the local truncation error for this scheme.

b. Numerically solve the KdV equation using an initial condition of the form $u(x, 0) = A\,\mathrm{sech}^2(x)$, $|x| \leq 20$, for $A = 1$.

c. Increase the amplitude of the initial condition and note what happens. Can you see the development of two or three solitons?

d. Apply the scheme for the condition $u(x, 0) = \cos \pi x$, $0 \leq x \leq 2$, and see if you can find the behavior as shown in Figure 1.