MEEN 689 – CFD
Homework 5
Name: Chinmay Rajesh Chavan
UIN: 633002377

Solution to the Inviscid Burgers' Equation using Upwind and Lax-Wendroff method

A. Using **Upwind** (FTBS) <u>low order</u> method:

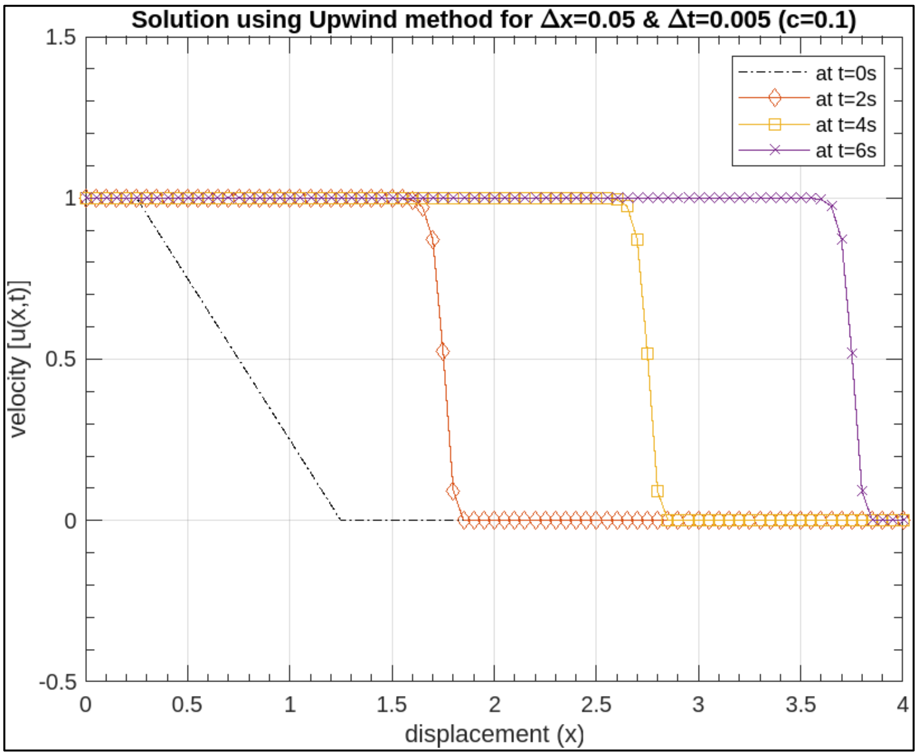

Fig.1. Solution to the inviscid Burgers' equation using **upwind** method for Δx=0.05 & Δt=0.005 (**c=0.1**) plotted at time levels 0, 2, 4, and 6.

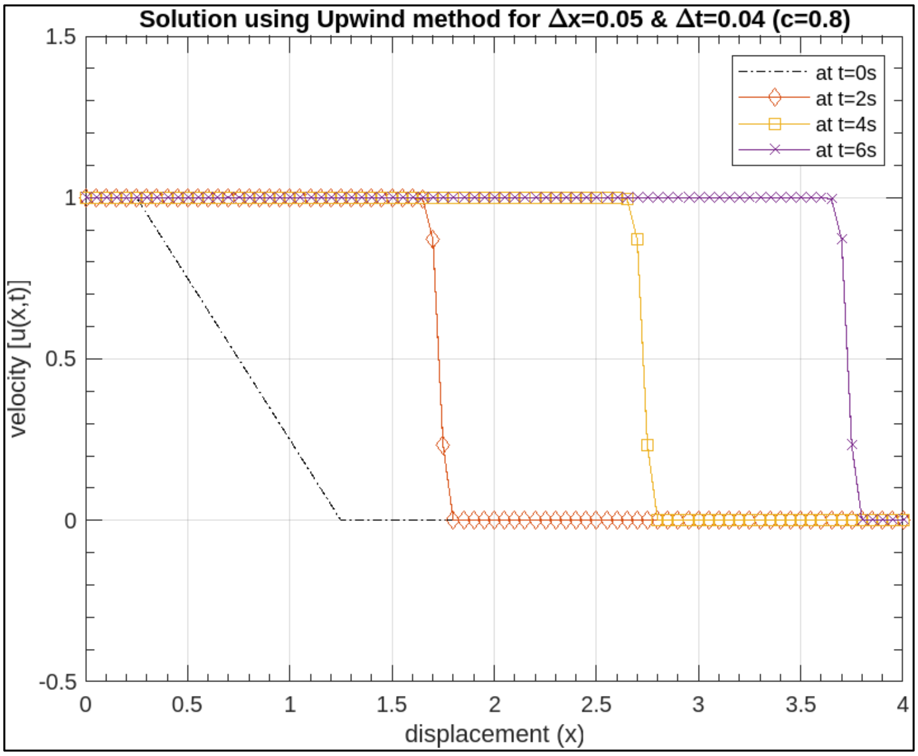

Fig.2. Solution to the inviscid Burgers' equation using **upwind** method for Δx=0.05 & Δt=0.04 (**c=0.8**) plotted at time levels 0, 2, 4, and 6.

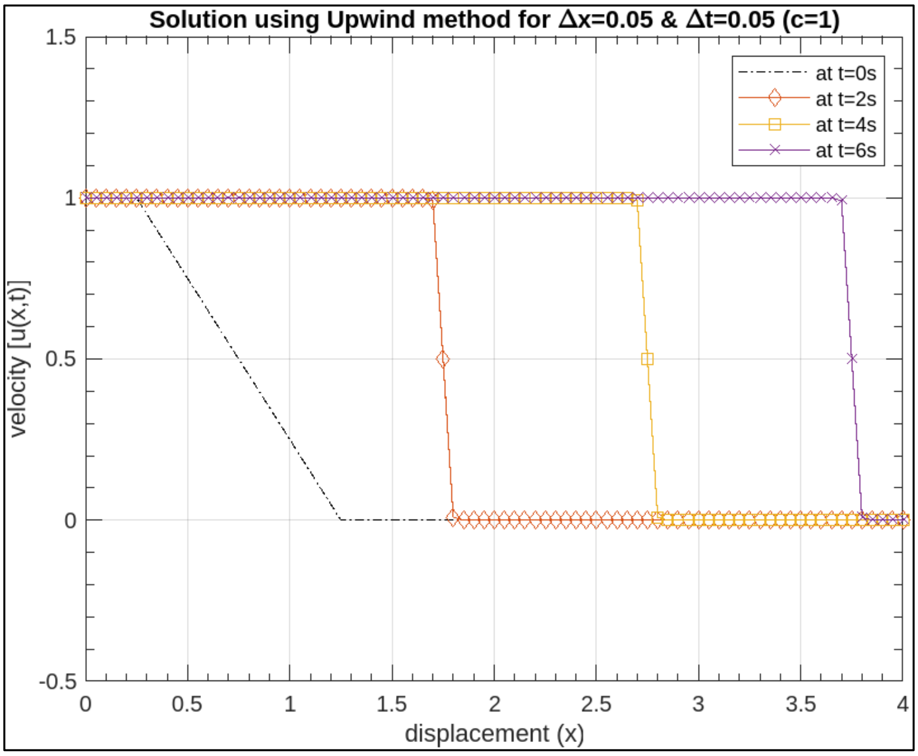

Fig.3. Solution to the inviscid Burgers' equation using **upwind** method for Δx=0.05 & Δt=0.05 (**c=1.0**) plotted at time levels 0, 2, 4, and 6.

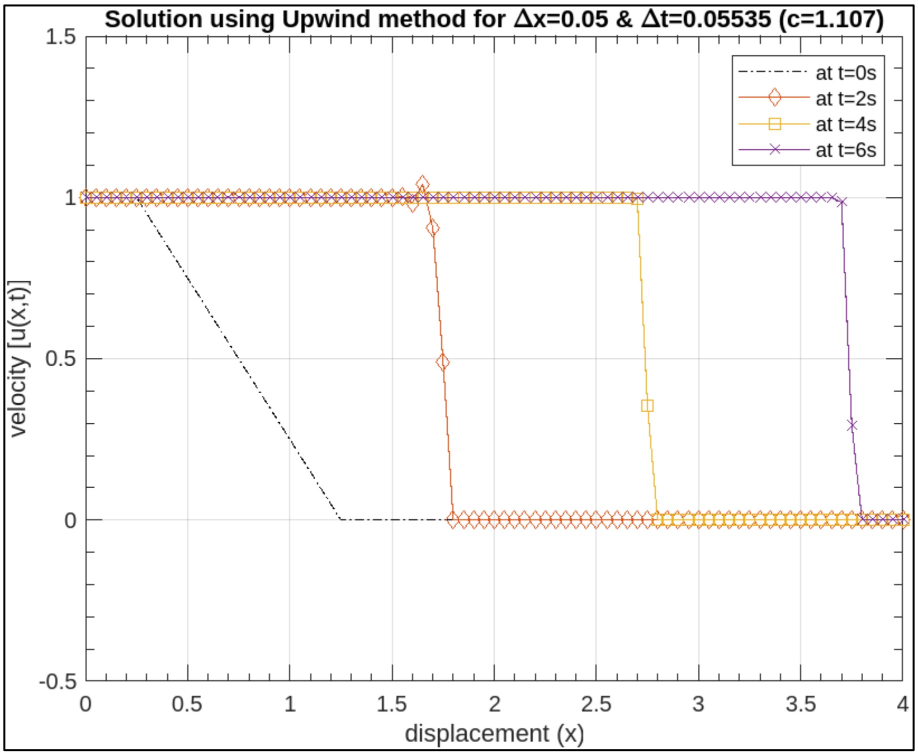

Fig.4. Solution to the inviscid Burgers' equation using **upwind** method for Δx=0.05 & Δt=0.05535 (**c=1.107**) plotted at time levels 0, 2, 4, and 6.

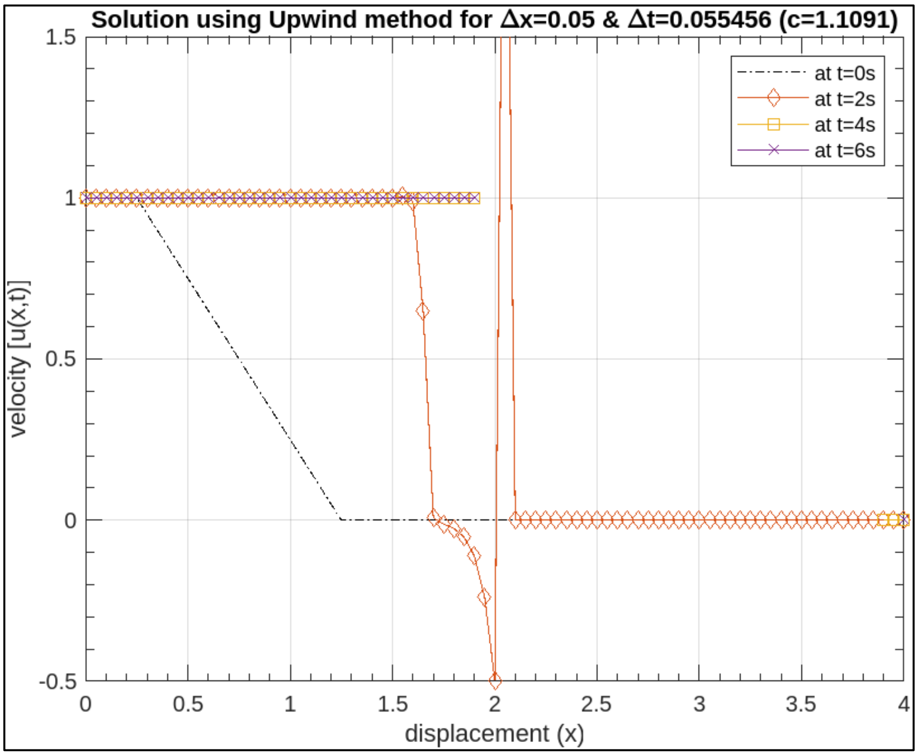

Fig.5. Solution to the inviscid Burgers' equation using **upwind** method for Δx=0.05 & Δt=0.055456 (**c=1.10911**) plotted at time levels 0, 2, 4, and 6.

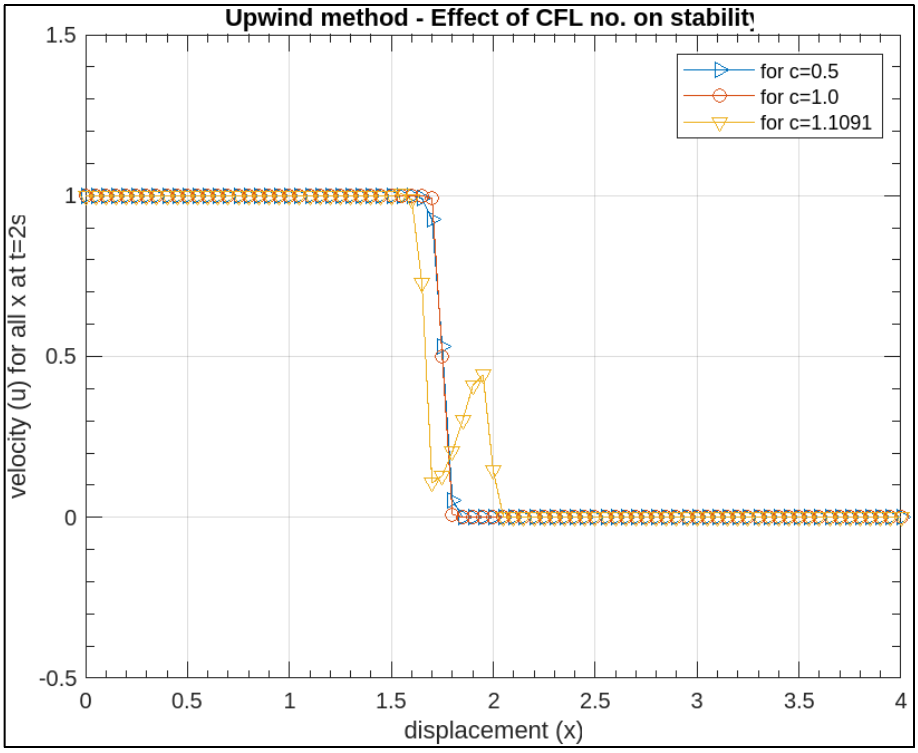

Fig.6. Stability comparison for solutions to the inviscid Burgers' equation using **upwind** method by varying CFL number for Δx=0.05 plotted at time level 2.

All the above plots are using upwind method for Δx=0.05 and varying CFL number (c). From Figs. (1)-(3), we can observe that for c<1 we get stable outputs and error is smaller as value of c gets closer to 1. Figures (4) & (5) show that we have instability in the solution for c>1, and the instability increases as we move away from 1. In Fig. (6), we see comparison of solution for c<1, c=1, and c>1 at time step of t=2 sec. From the solutions plotted, it is evident that in upwind method, for a stable and more accurate result we need to set value of CFL number less than 1 but closer to 1.

B. Using 2-step **Lax-Wendroff** high order method:

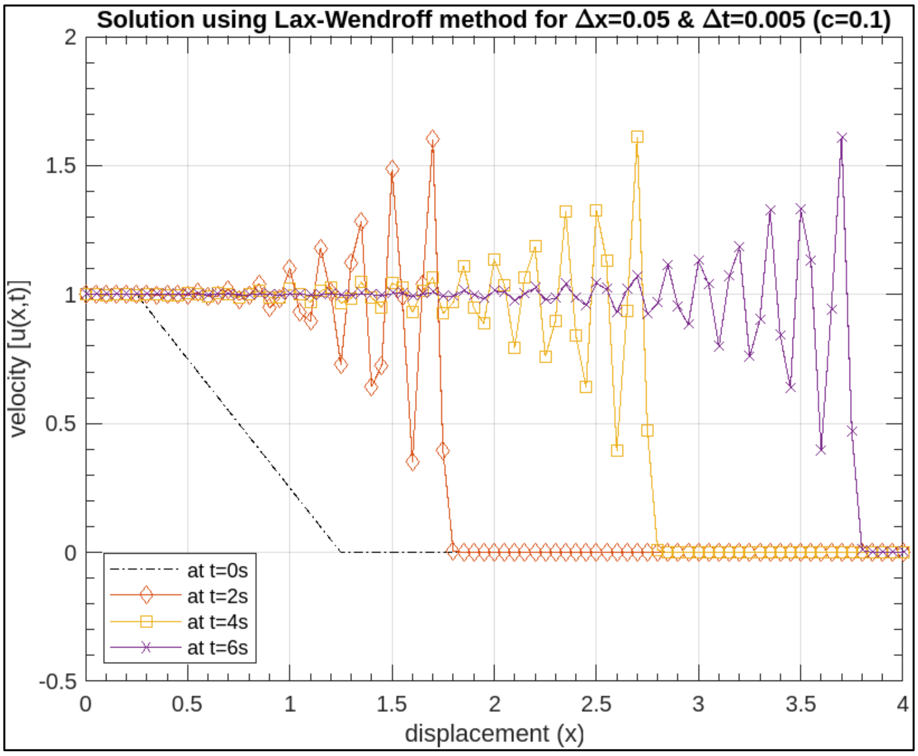

Fig.7. Solution to the inviscid Burgers' equation using **2-step Lax-Wendroff** method for Δx=0.05 & Δt=0.005 (**c=0.1**) plotted at time levels 0, 2, 4, and 6.

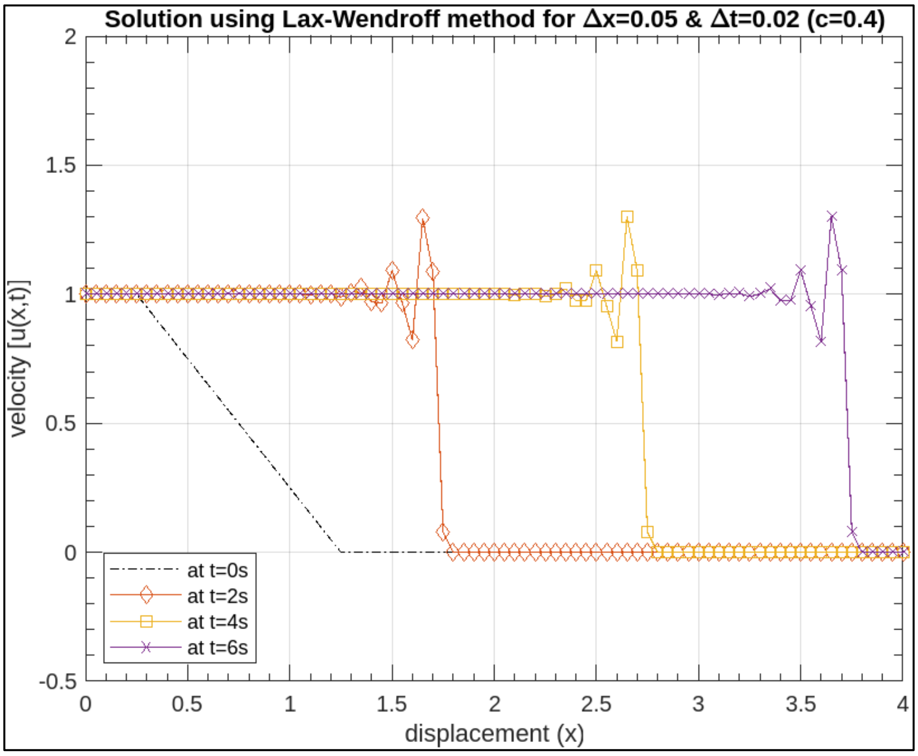

Fig.8. Solution to the inviscid Burgers' equation using **2-step Lax-Wendroff** method for Δx=0.05 & Δt=0.02 (**c=0.4**) plotted at time levels 0, 2, 4, and 6

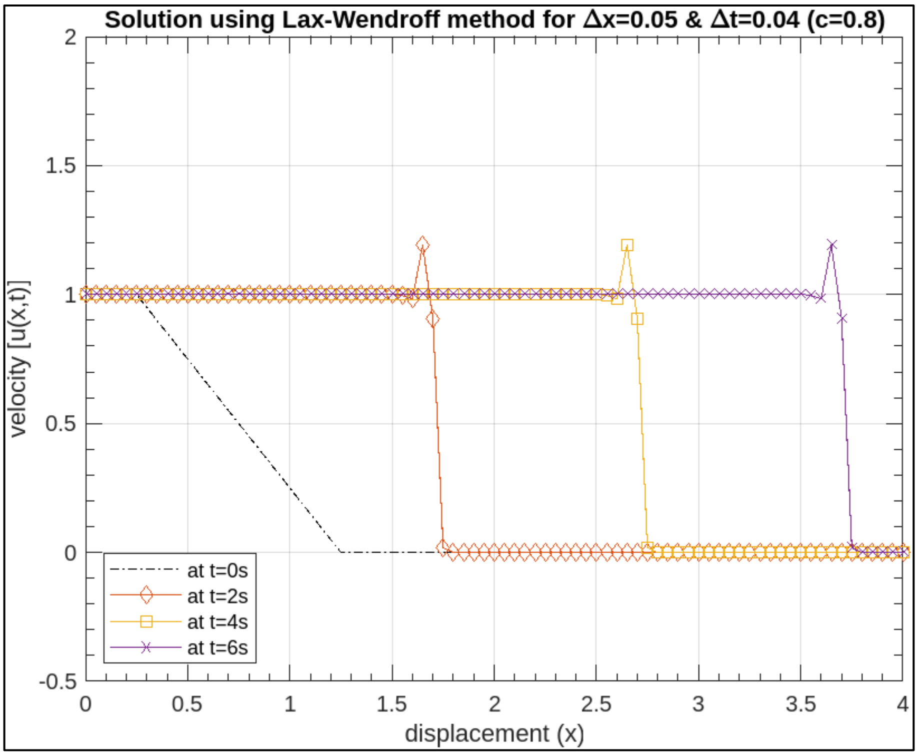

Fig.9. Solution to the inviscid Burgers' equation using **2-step Lax-Wendroff** method for Δx=0.05 & Δt=0.04 (**c=0.8**) plotted at time levels 0, 2, 4, and 6

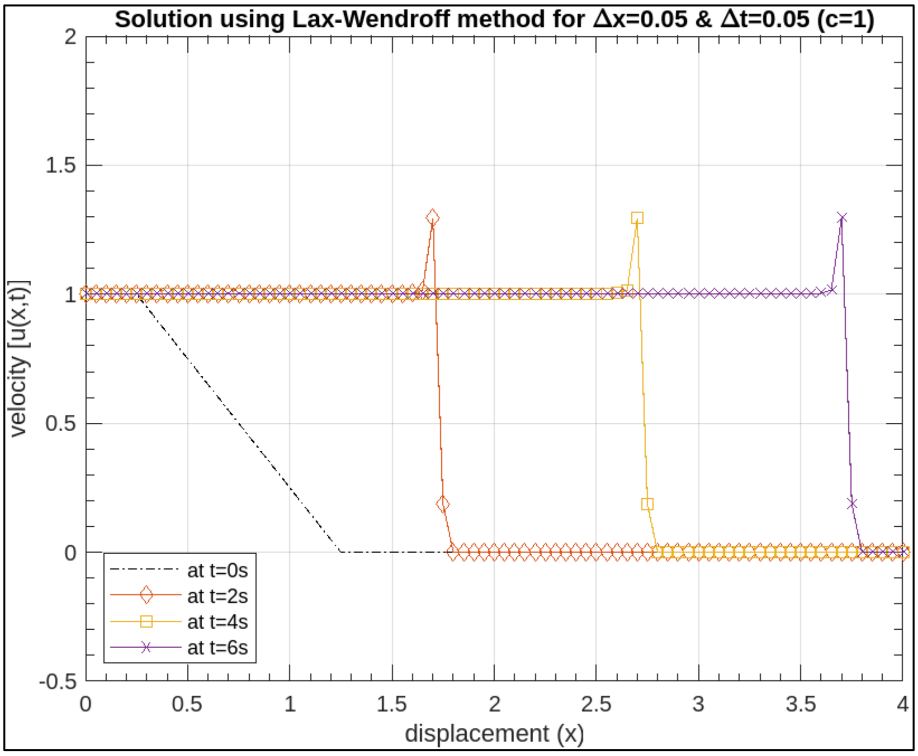

Fig.10. Solution to the inviscid Burgers' equation using **2-step Lax-Wendroff** method for Δx=0.05 & Δt=0.05 (**c=1.0**) plotted at time levels 0, 2, 4, and 6

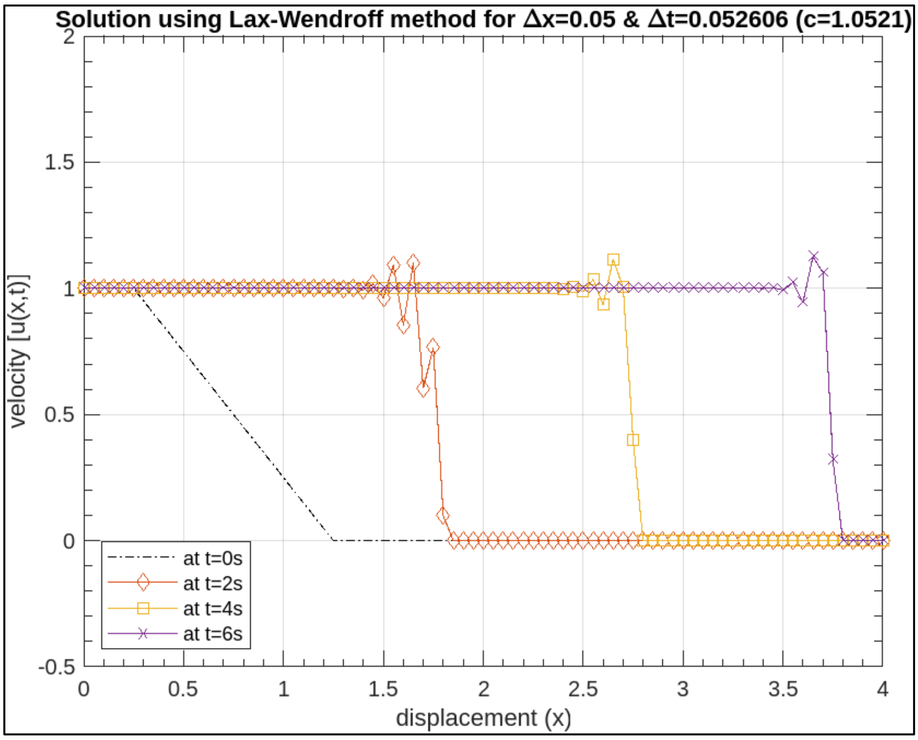

Fig.11. Solution to the inviscid Burgers' equation using **2-step Lax-Wendroff** method for Δx=0.05 & Δt=0.052606 (**c=1.05212**) plotted at time levels 0, 2, 4, and 6

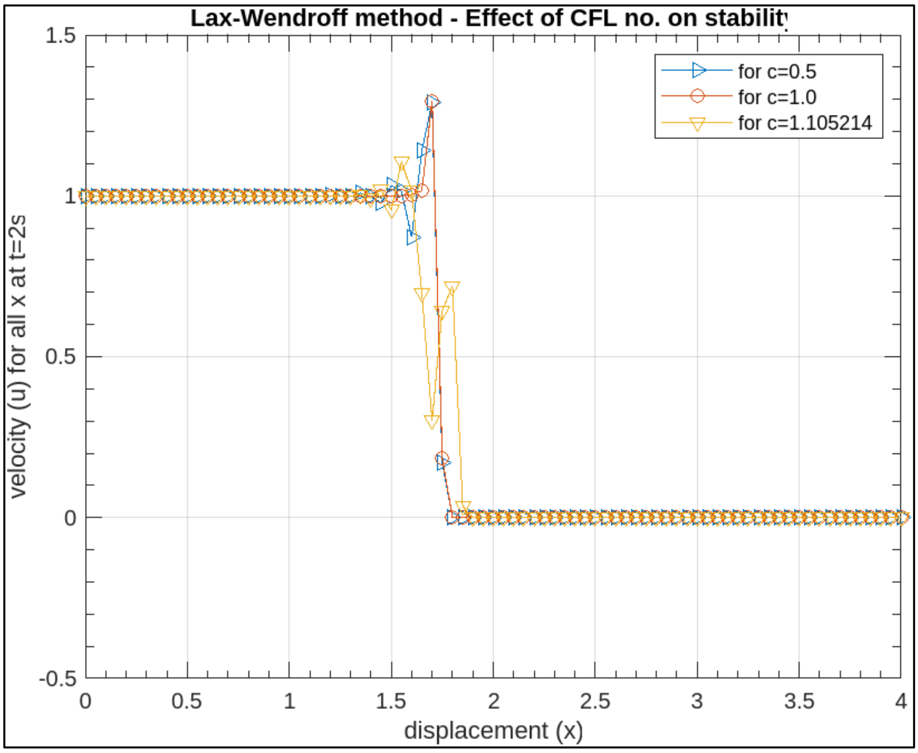

Fig.12. Stability comparison for solutions to the inviscid Burgers' equation using **2-step Lax-Wendroff** method by varying CFL number for Δx=0.05 plotted at time level 2.

All the above plots are using Lax-Wendroff method for Δx=0.05 and varying CFL number (c). From Figs. (7)-(10), we can observe that for c<1 we get overall stable outputs with some ringing (local instabilities) and error is smaller as value of c gets closer to 1. Even at c=1, we can observe one sharp peak at discontinuity. Figure (11) shows that we have instability in the solution for c>1. In Fig. (12), we see comparison of solution for c<1, c=1, and c>1 at time step of t=2 sec. From the solutions plotted, it is evident that in Lax-Wendroff method, for an overall stable, smaller ringing and more accurate result we need to set value of CFL number less than 1 but closer to 1.

APPENDIX

MATLAB codes used for CFD HW#5:-

1.) Code to plot solution to inviscid Burgers' equation using **Upwind (FTBS)** method for different CFL numbers at t=0, 2, 4, 6 sec:

```
clc;
%% Inputs
dx=0.05;
for c=[0.1 0.8 1 1.107 1.10911]
dt=c*dx;
x0=0;
xn=4;
t0=0;
tn=6;
NT=floor((tn-t0)/dt+1);
NX=floor((xn-x0)/dx+1);
t=0;
x=0;
%% Initialise u with I.C.
for it=1:NT+1
    for j=1:NX
        x=x0+dx*(j-1);
        if x<0.25
            u(it,j)=1;
        elseif x>1.25
            u(it,j)=0;
        else
            u(it,j)=1.25-x;
        end
    end
end
figure;
X=linspace(x0,xn,NX);
plot(X,u(1,:),"-.black"); %Plot solution at t=0
hold on;
%% Upwind (FTBS) method
for it=1:NT
    t=t0+dt*it;
    for i=1:NX
        E(it,i)=u(it,i)*u(it,i)/2;
    end
    for k=2:NX-1
        u(it+1,k)=u(it,k)-c*(E(it,k)-E(it,k-1));
    end
end
%% Plotting solution
it2=floor(2/dt+1);
plot(X,u(it2,:),"-diamond"); %Plot solution at t=2
it4=floor(4/dt+1);
plot(X,u(it4,:),"-square"); %Plot solution at t=4
it6=floor(6/dt+1);
plot(X,u(it6,:),"-x"); %Plot solution at t=6
legend('at t=0s','at t=2s','at t=4s','at t=6s');
ylim([-0.5 1.5]);
xlabel('displacement (x)');
ylabel('velocity [u(x,t)]');
title("Solution using Upwind method for \Deltax="+dx+" & \Deltat="+dt+" (c="+c+")");
set(gca,'XMinorTick','on','YMinorTick','on','TickLength',[0.02,0.005]);
grid on;
hold off;
end
```

2.) Code to plot solution using **Upwind (FTBS)** method for **varying CFL number** at t=2 sec:

```
clc;
figure;
mkrdata =["->" "-o" "-v"];
count=0;
%% Inputs
dx=0.05;
for c=[0.5 1 1.1091]
dt=c*dx;
x0=0;
xn=4;
t0=0;
tn=6;
NT=floor((tn-t0)/dt+1);
NX=floor((xn-x0)/dx+1);
t=0;
x=0;
%% Initialise u with I.C.
for it=1:NT+1
    for j=1:NX
        x=x0+dx*(j-1);
        if x<0.25
            u(it,j)=1;
        elseif x>1.25
            u(it,j)=0;
        else
            u(it,j)=1.25-x;
        end
    end
end
X=linspace(x0,xn,NX);
%% Upwind (FTBS) method
for it=1:NT
    t=t0+dt*it;
    for i=1:NX
        E(it,i)=u(it,i)*u(it,i)/2;
    end
    for k=2:NX-1
        u(it+1,k)=u(it,k)-c*(E(it,k)-E(it,k-1));
    end
end
%% Plotting solution
it2=floor(2/dt+1);
count=count+1;
plot(X,u(it2,:),mkrdata(1,count)); %Plot solution at t=2
hold on;
end
legend('for c=0.5','for c=1.0','for c=1.1091');
ylim([-0.5 1.5]);
xlabel('displacement (x)');
ylabel('velocity (u) for all x at t=2s');
title('Upwind method - Effect of CFL no. on stability');
set(gca,'XMinorTick','on','YMinorTick','on','TickLength',[0.02,0.005]);
grid on;
hold off;
```

3.) Code to plot solution to inviscid Burgers' equation using **Lax-Wendroff** method for different CFL numbers at t=0, 2, 4, 6 sec:

```
clc;
%% Inputs
dx=0.05;
```

```
for c=[0.1 0.4 0.8 1 1.05212]
%c=1.05212;
dt=c*dx;
x0=0;
xn=4;
t0=0;
tn=6;
NT=floor((tn-t0)/dt+1);
NX=floor((xn-x0)/dx+1);
t=0;
x=0;
%% Initialise u with I.C.
for it=1:NT+1
    for j=1:NX
        x=x0+dx*(j-1);
        if x<0.25
            u(it,j)=1;
        elseif x>1.25
            u(it,j)=0;
        else
            u(it,j)=1.25-x;
        end
    end
end
figure;
X=linspace(x0,xn,NX);
plot(X,u(1,:),"-.black"); %Plot solution at t=0
hold on;
%% 2-step Lax-Wendroff method
for it=1:NT
    t=t0+dt*it;
    for i=1:NX
        E(it,i)=u(it,i)*u(it,i)/2;
    end
    for j=1:NX-1
        u1(it,j)=((u(it,j)+u(it,j+1))-c*(E(it,j+1)-E(it,j)))/2;
    end
    for m=1:NX-1
        E1(it,m)=u1(it,m)*u1(it,m)/2;
    end
    for k=2:NX-1
        u(it+1,k)=u(it,k)-c*(E1(it,k)-E1(it,k-1));
    end
end
%% Plotting solution
it2=floor(2/dt+1);
plot(X,u(it2,:),"-diamond"); %Plot solution at t=2
it4=floor(4/dt+1);
plot(X,u(it4,:),"-square"); %Plot solution at t=4
it6=floor(6/dt+1);
plot(X,u(it6,:),"-x"); %Plot solution at t=6
legend('at t=0s','at t=2s','at t=4s','at t=6s',"Location","southwest");
ylim([-0.5 2]);
xlabel('displacement (x)');
ylabel('velocity [u(x,t)]');
title("Solution using Lax-Wendroff method for \Deltax="+dx+" & \Deltat="+dt+"
(c="+c+")");
set(gca,'XMinorTick','on','YMinorTick','on','TickLength',[0.02,0.005]);
grid on;
hold off;
end
```

4.) Code to plot solution using **Lax-Wendroff** method for **varying CFL number** at t=2 sec:

```
clc;
figure;
mkrdata =["->" "-o" "-v"];
count=0;
%% Inputs
dx=0.05;
for c=[0.5 1 1.05214]
dt=c*dx;
x0=0;
xn=4;
t0=0;
tn=6;
NT=floor((tn-t0)/dt+1);
NX=floor((xn-x0)/dx+1);
t=0;
x=0;
%% Initialise u with I.C.
for it=1:NT+1
    for j=1:NX
        x=x0+dx*(j-1);
        if x<0.25
            u(it,j)=1;
        elseif x>1.25
            u(it,j)=0;
        else
            u(it,j)=1.25-x;
        end
    end
end
X=linspace(x0,xn,NX);
%% 2-step Lax-Wendroff method
for it=1:NT
    t=t0+dt*it;
    for i=1:NX
        E(it,i)=u(it,i)*u(it,i)/2;
    end
    for j=1:NX-1
        u1(it,j)=((u(it,j)+u(it,j+1))-c*(E(it,j+1)-E(it,j)))/2;
    end
    for m=1:NX-1
        E1(it,m)=u1(it,m)*u1(it,m)/2;
    end
    for k=2:NX-1
        u(it+1,k)=u(it,k)-c*(E1(it,k)-E1(it,k-1));
    end
end
%% Plotting solution
it2=floor(2/dt+1);
count=count+1;
plot(X,u(it2,:),mkrdata(1,count)); %Plot solution at t=2
hold on;
end
legend('for c=0.5','for c=1.0','for c=1.105214');
ylim([-0.5 1.5]);
xlabel('displacement (x)');
ylabel('velocity (u) for all x at t=2s');
title('Lax-Wendroff method - Effect of CFL no. on stability');
set(gca,'XMinorTick','on','YMinorTick','on','TickLength',[0.02,0.005]);
grid on;
hold off;
```