Computational Fluid Dynamics

Francisco Castillo
Homework 8

March 27, 2018

1 Introduction

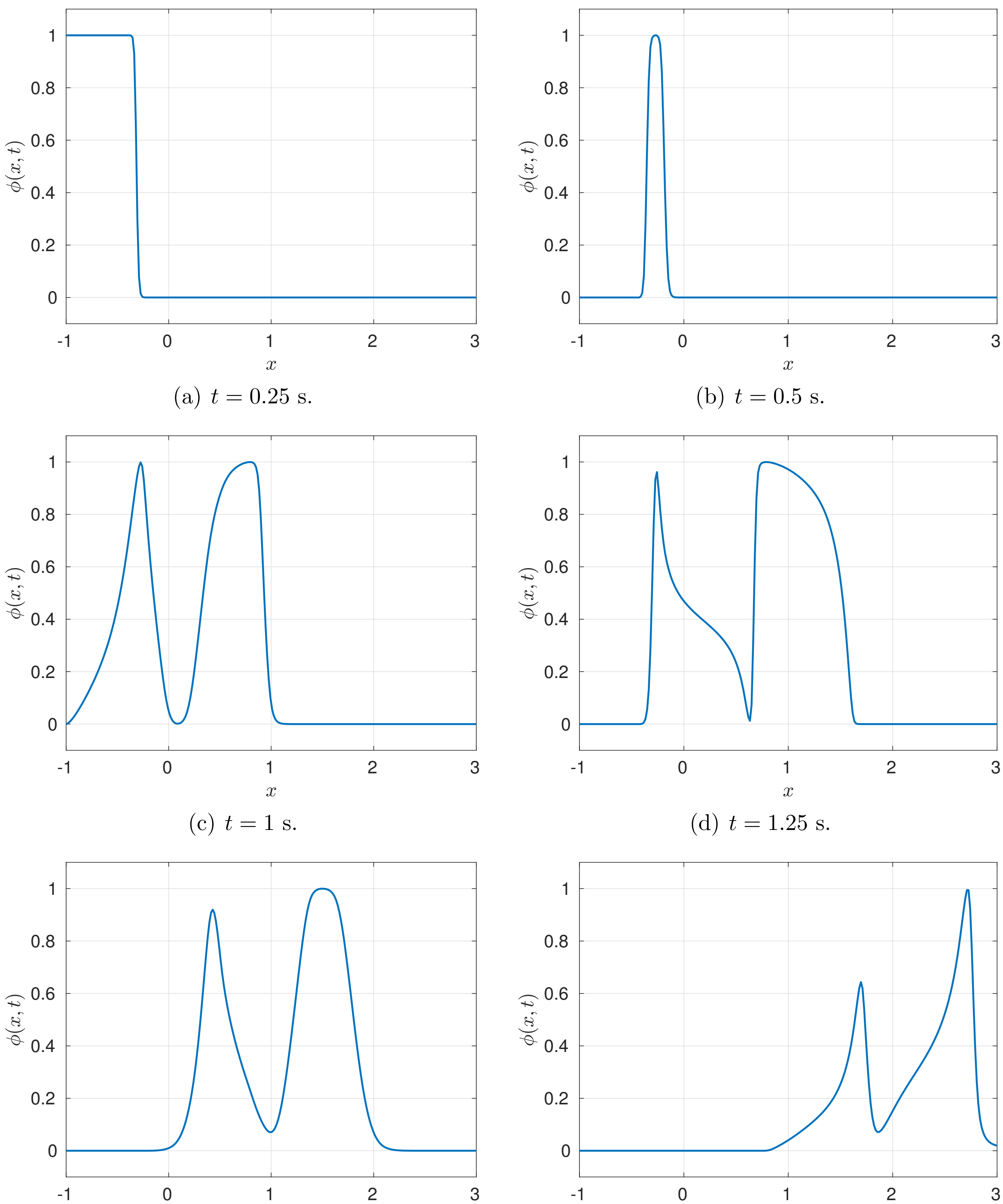
In this assignment we will study a one dimensional hyperbolic PDE

$$\frac{\partial \phi}{\partial t} + a(x,t)\frac{\partial \phi}{\partial x} = 0,$$

with a discontinuous boundary condition and a sinusoidal velocity $a(x,t)$. We will use a the total variation diminishing (TVD) third order Runge-Kutta method, abbreviated $TVD\ RK3$, to solve the PDE in time and a fith order Weighted Essentially Non-Oscillatory Scheme, abreviated $WENO-5$, to solve the spatial derivative.

2 Results

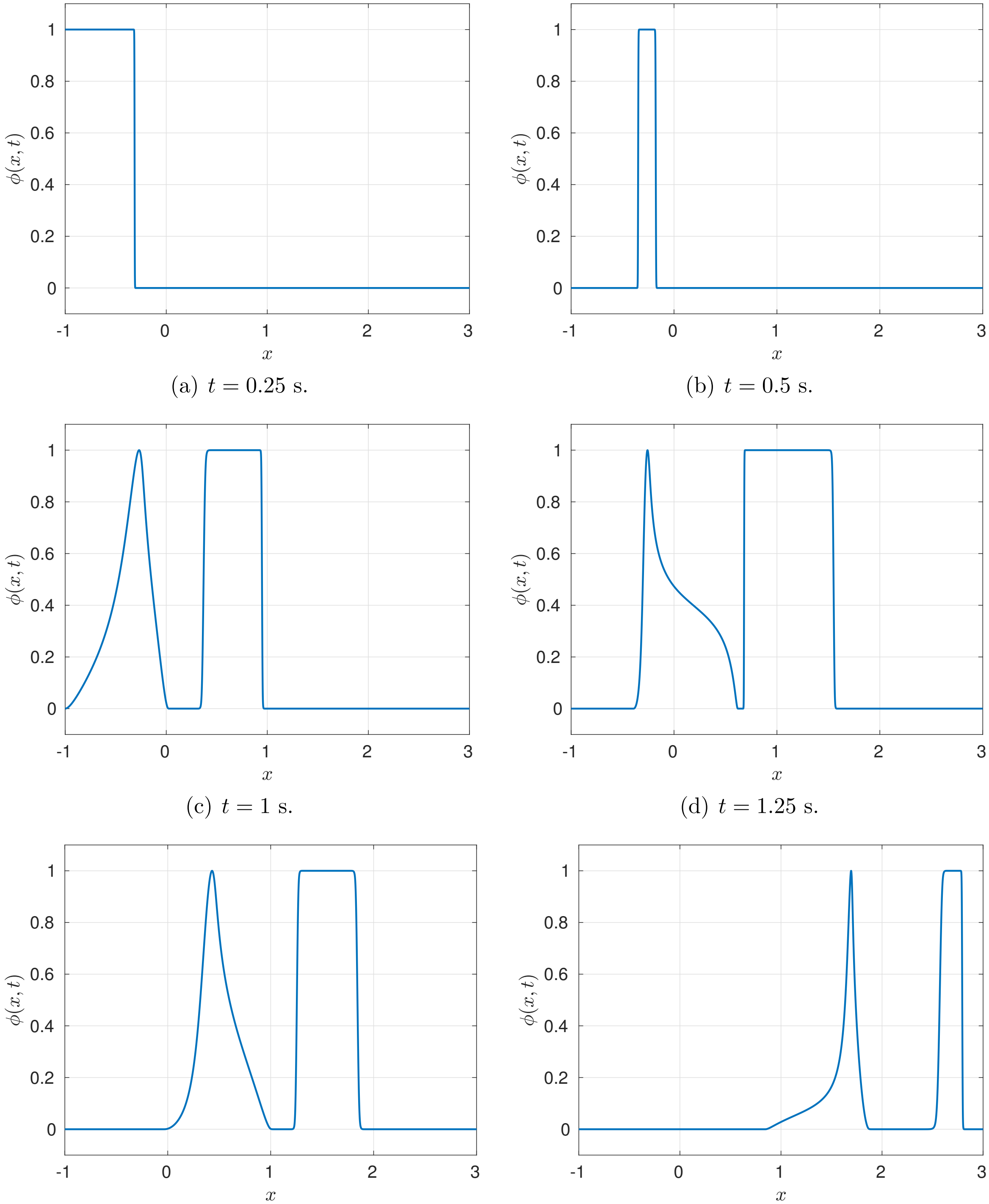
In the first figure we can see the solution $\phi(x,t)$ at different values of time for $M = 256$ and $CFL = 0.8$. We can see the strange solution profiles that are due to the form of the left boundary condition.



(a) $t = 0.25$ s. (b) $t = 0.5$ s.
(c) $t = 1$ s. (d) $t = 1.25$ s.
(e) $t = 1.5$ s. (f) $t = 2.1$ s.

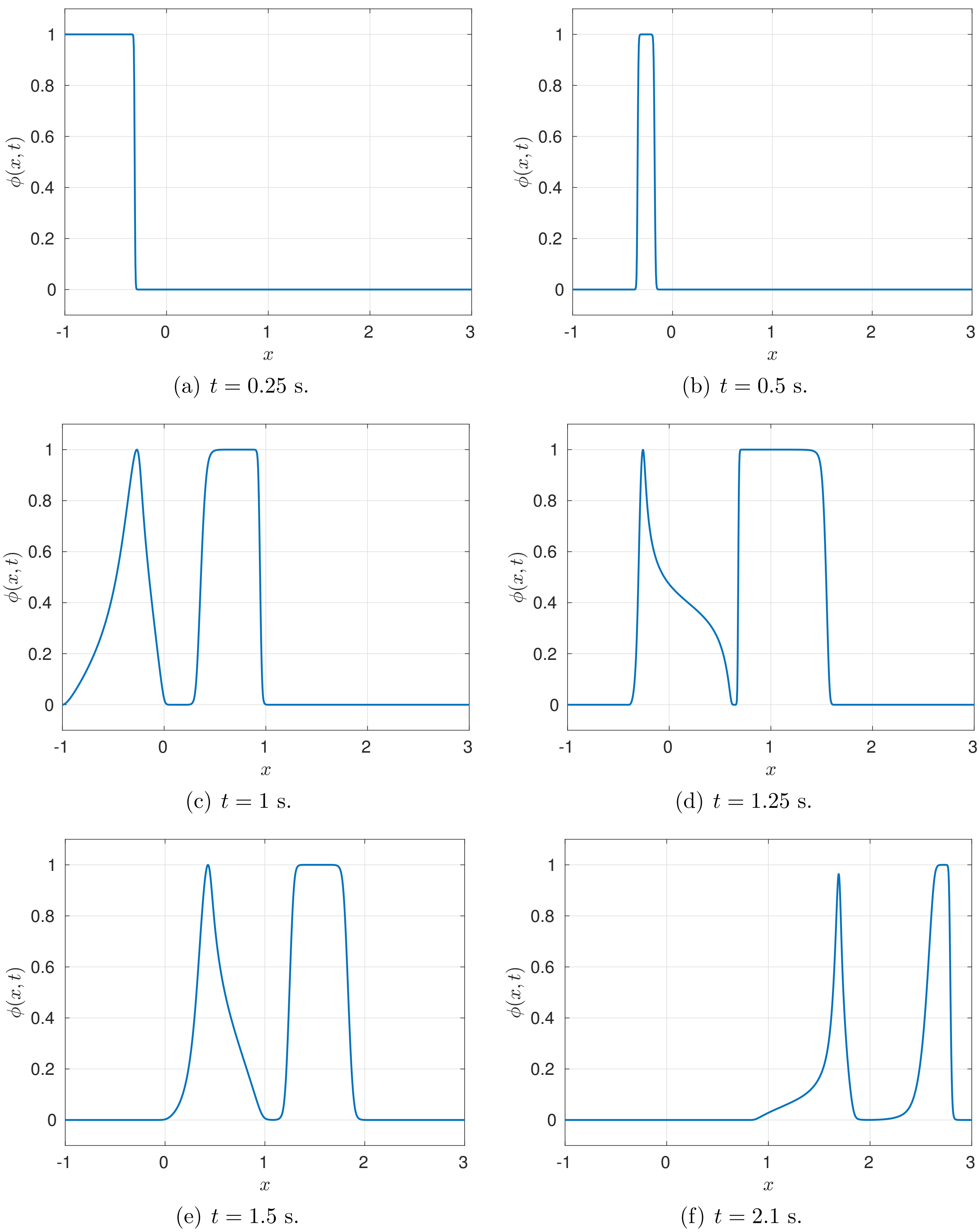
Figure 1: M=256.

In the next two figures we see the solution profiles for $M = 4096$, $CFL = 0.8$ (Figure 2) and $M = 1024$, $CFL = 0.5$ (Figure 3). Both configurations satisfy the accuracy requirement as we show in the tables below. In this case I have found more beneficial when it comes to computational cost, to reduce the time step instead of going to such fine meshes. However, we can see in figure 4 that if we want an outstanding precision in space to catch the discontinuities, we should increase the number of elements. However, reducing the time step, and using 4 times less elements (Figure 4(c)), gives us a very similar result much faster.



(a) $t = 0.25$ s.
(b) $t = 0.5$ s.
(c) $t = 1$ s.
(d) $t = 1.25$ s.
(e) $t = 1.5$ s.
(f) $t = 2.1$ s.

Figure 2: M=4096, CFL=0.8

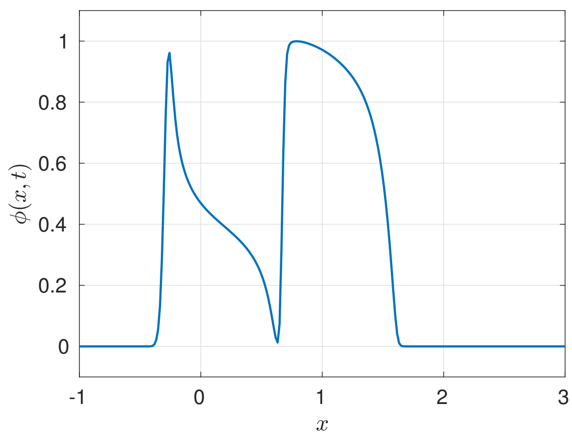


(a) $t = 0.25$ s. (b) $t = 0.5$ s.

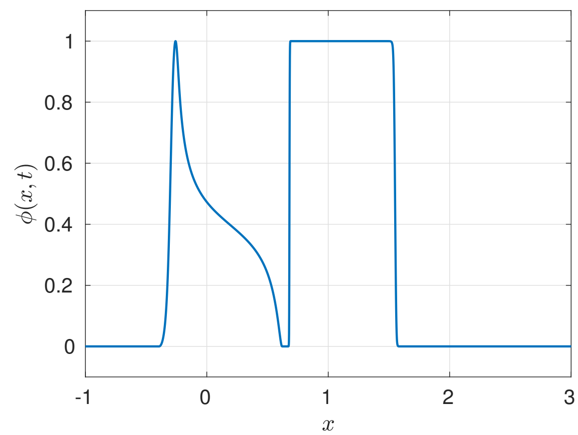
(c) $t = 1$ s. (d) $t = 1.25$ s.

(e) $t = 1.5$ s. (f) $t = 2.1$ s.

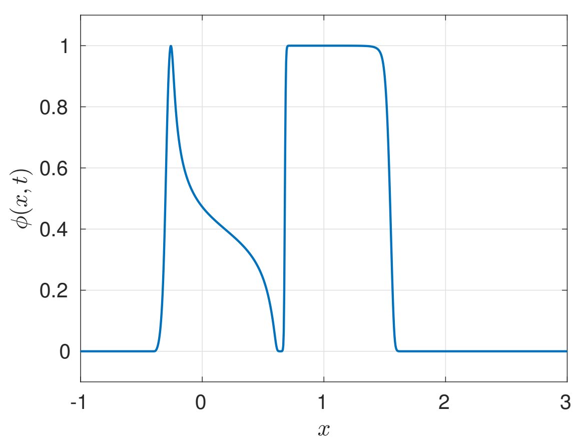
Figure 3: M=4096, CFL=0.5



(a) $M = 256$, $CFL = 0.8$.



(b) $M = 4096$, $CFL = 0.8$.



(c) $M = 1024$, $CFL = 0.5$.

Figure 4: Comparison of the different solutions.

The GCI analysis details for both CFL values are shown in the tables below. Note that

$$\beta = \frac{GCI_{12}}{GCI_{23}} r^p,$$

and $\phi_{h=0}$ is obtained by Richardson extrapolation. We can see that $\beta \in [0.95, 1.05]$ which implies that we are in the asymptotic range of convergence, and for the last mesh we have a GCI_{12} value less than 0.1%, the requested accuracy.

M	$\phi(0, 1.25)$
128	0.464517670619715
256	0.469387263974194
512	0.471586585046651
1024	0.472388657559251
2048	0.472790327155522
4096	0.472989985610756

Table 1: GCI analysis data for $CFL = 0.8$.

M	$\phi_{h=0}$	p	GCI_{12} (%)	GCI_{23} (%)	β
128	-	-	-	-	-
256	-	-	-	-	-
512	0.473398015743182	1.146743067686559	0.4801426593676	1.0680817516835	0.99533
1024	0.472849076939131	1.455253657646967	0.1218327780824	0.3346394735644	0.99830
2048	0.473193267438895	0.997723408586615	0.1065324997757	0.2129099277600	0.99915
4096	0.473187318780311	1.008475093835296	0.0521504618381	0.1049597469500	0.99957

Table 2: GCI analysis results for $CFL = 0.8$.

M	$\phi(0, 1.25)$
128	0.470667415947558
256	0.471166273852559
512	0.472263797941595
1024	0.472683229511829

Table 3: GCI analysis data for $CFL = 0.5$.

M	$\phi_{h=0}$	p	GCI_{12} (%)	GCI_{23} (%)	β
128	-	-	-	-	-
256	-	-	-	-	-
512	0.470251726509905	-1.137551764106874	-0.5325602556401	-0.2426286094229	0.99767
1024	0.472942667327806	1.387745244157744	0.0686077376399	0.1796848999356	0.99911

Table 4: GCI analysis results for $CFL = 0.5$.

HOMEWORK 8 - FRANCISCO CASTILLO

Contents

Defined functions

```
function a = velocity(x,t)
a = 2.3+1.7*sin(2*pi*x)+1.5*sin(5*pi*t);
end


function phi = updateGhostCells(phi,t,M)
% Left boundary, Dirichlet BC
phi(3)=2*phi_left(t)-phi(4);
phi(2)=2*phi_left(t)-phi(5);
phi(1)=2*phi_left(t)-phi(6);
% Right boundary, zero Neumann BC
phi(M+4)=phi(M+3);
phi(M+5)=phi(M+2);
phi(M+6)=phi(M+1);
end


function phileft = phi_left(t)
if 0<=t && t<=0.25
    phileft=1;
elseif 0.25<t && t<=0.5
    phileft=0;
elseif 0.5<t && t<=1
    phileft=(1-cos(4*pi*t))/2;
elseif t>1
    phileft=0;
else
    error('t negative')
end
end


function DphiDx = WENO5(phi,i,hx,a)
if a>0
    DphiDx = (1/(12*hx))*(phi(i-2)-8*phi(i-1)+8*phi(i+1)-phi(i+2))...
        -psiWENO((phi(i-1)-2*phi(i-2)+phi(i-3))/hx,...
        (phi(i)-2*phi(i-1)+phi(i-2))/hx,...
        (phi(i+1)-2*phi(i)+phi(i-1))/hx,...
        (phi(i+2)-2*phi(i+1)+phi(i))/hx);
elseif a<0
    DphiDx = (1/(12*hx))*(phi(i-2)-8*phi(i-1)+8*phi(i+1)-phi(i+2))...
        +psiWENO((phi(i+3)-2*phi(i+2)+phi(i+1))/hx,...
        (phi(i+2)-2*phi(i+1)+phi(i))/hx,...
        (phi(i+1)-2*phi(i)+phi(i-1))/hx,...
        (phi(i)-2*phi(i-1)+phi(i-2))/hx);
end
end


function psi = psiWENO(a,b,c,d)
eps=1e-6;
IS0=13*(a-b)^2+3*(a-3*b)^2;
IS1=13*(b-c)^2+3*(b+c)^2;
IS2=13*(c-d)^2+3*(3*c-d)^2;

a0=(eps+IS0)^(-2);
a1=6*(eps+IS1)^(-2);
a2=3*(eps+IS2)^(-2);

w0=a0/(a0+a1+a2);
w2=a2/(a0+a1+a2);

psi = (a-2*b+c)*w0/3+(w2-0.5)*(b-2*c+d)/6;
end
```

```
function phi3 = TVDRK3(phi0,M,hx,dt,t,a)
% Constants and prealocation
a10=1;
a20=-3/4; a21=1/4;
a30=-1/12; a31=-1/12; a32=2/3;
phi1=zeros(M+6,1);
phi2=zeros(M+6,1);
phi3=zeros(M+6,1);

%%% STEP 1 %%%
for i=4:M+3
    phi1(i)=phi0(i)-a10*a(i)*dt*WENO5(phi0,i,hx,a(i));
end
% Update ghost cells
phi1=updateGhostCells(phi1,t,M);

%%% STEP 2 %%%
for i=4:M+3
    phi2(i)=phi1(i)-a20*a(i)*dt*WENO5(phi0,i,hx,a(i))...
        -a21*a(i)*dt*WENO5(phi1,i,hx,a(i));
end
% Update ghost cells
phi2=updateGhostCells(phi2,t,M);

%%% STEP 3 %%%
for i=4:M+3
    phi3(i)=phi2(i)-a30*a(i)*dt*WENO5(phi0,i,hx,a(i))...
        -a31*a(i)*dt*WENO5(phi1,i,hx,a(i))...
        -a32*a(i)*dt*WENO5(phi2,i,hx,a(i));
end
% Update ghost cells
phi3=updateGhostCells(phi3,t+dt,M);
end
```

Problem

```
clear variables
close all
clc
format long

axisSize=14;
linewidth=1.5;
L=4;
CFL=0.8;
M=64;
i=0;
T=nan(3,7);
check=1;
while check>0.1
```

```
    i=i+1;
    M=2*M
    hx=L/M;
    x=linspace(-1-2.5*hx,3+2.5*hx,M+6)'; % Cell centered mesh with three
                                         % ghost cells at each side
```

Initialization

```
    time=0;
    a=velocity(x,time);
    phi = zeros(M+6,1);
    phi(3)=2*phi_left(time);
    phi(2)=2*phi_left(time);  % Since phi(4:6)=0 it is not necessary
    phi(1)=2*phi_left(time);  % to include them

    dt=CFL*hx/(max(abs(a)));
    outputTime=[0.25 0.5 1 1.25 1.5 2.1];
    endtime=outputTime(end);

    n=1;
    step=0;
    while time < endtime
```

```
        a=velocity(x,time);
        if (time < outputTime(n) && time+dt >= outputTime(n))
            dt=outputTime(n)-time;
            n=n+1;
        else
```

```
        dt=CFL*hx/(max(abs(a)));
    end
    phi = TVDRK3(phi,M,hx,dt,time,a);
    time=time+dt;
```

Plot required at part 5

```
    if (M==256 && CFL==0.8 && ismember(time,outputTime))
        figure(n-1)
        plot(x,phi,'linewidth',linewidth)
        grid on
        axis([-1 3 -0.1 1.1])
        xlabel('$x$','Interpreter','latex')
        ylabel('$\phi(x,t)$','Interpreter','latex')
        set(gca,'fontsize',axisSize)
        txt=['Latex/FIGURES/phi_' num2str(n-1)];
        saveas(gcf,txt,'epsc')
    end
```

Plot required at part 6

For CFL=0.8

```
    if (M==4096 && CFL==0.8 && ismember(time,outputTime))
        figure(n+6)
        plot(x,phi,'linewidth',linewidth)
        grid on
        axis([-1 3 -0.1 1.1])
        xlabel('$x$','Interpreter','latex')
        ylabel('$\phi(x,t)$','Interpreter','latex')
        set(gca,'fontsize',axisSize)
        txt=['Latex/FIGURES/phi08_' num2str(n-1)];
        saveas(gcf,txt,'epsc')
    end
    % For CFL=0.5
    if (M==1024 && CFL==0.5 && ismember(time,outputTime))
        figure(n+6)
        plot(x,phi,'linewidth',linewidth)
        grid on
        axis([-1 3 -0.1 1.1])
        xlabel('$x$','Interpreter','latex')
        ylabel('$\phi(x,t)$','Interpreter','latex')
        set(gca,'fontsize',axisSize)
        txt=['Latex/FIGURES/phi05_' num2str(n-1)];
        saveas(gcf,txt,'epsc')
    end
```

Phi at x=0 and t=1.25

```
    if time==1.25
        phiGCI(i)=(phi(find(x<=0,1,'last'))+phi(find(x>=0,1)))/2;
    end
```

```
end
```

GCI analysis

```
if i>=3
    r=2;
    Fsec=1.25;
    p(i)=log(abs(phiGCI(i-2)-phiGCI(i-1))/abs(phiGCI(i-1)-phiGCI(i)))/log(r);
    phi_h0(i)=phiGCI(i)+(phiGCI(i)-phiGCI(i-1))/(r^p(i)-1);
    GCI12(i)=Fsec*abs(1-phiGCI(i-1)/phiGCI(i))/(r^p(i)-1);
    GCI23(i)=Fsec*abs(1-phiGCI(i-2)/phiGCI(i-1))/(r^p(i)-1);
    coeff(i)=GCI12(i)*r^p(i)/GCI23(i);
    percent(i)=GCI12(i)*100;
    check=abs(percent(i));
    % Include results in a table
    T(i,2)=p(i);
    T(i,3)=phi_h0(i);
    T(i,4)=GCI12(i);
    T(i,5)=GCI23(i);
    T(i,6)=coeff(i);
    T(i,7)=percent(i);
end
% Include results in a table
T(i,1)=M;
```

```
end
T=array2table(T,'VariableNames',{'M','p','phi0','GCI12','GCI23','coeff','Check'})
if CFL==0.8
    save('Case1_CFL08');
elseif CFL==0.5
    save('Case2_CFL05');
end
```

Published with MATLAB® R2017a

Solve PDE:
$$\frac{\partial \phi}{\partial t} + a(x,t)\frac{\partial \phi}{\partial x} = 0$$

~~Discretization~~

Index form:
$$\left.\frac{\partial \phi^n}{\partial t}\right|_i = -a_i^n \left.\frac{\partial \phi^n}{\partial x}\right|_i$$

For the time derivative: TVD-RK3 method.

$$\phi_i^{(1)} = \phi_i^n - \alpha_{1,0}\left(a_i^n \Delta t \left.\frac{\partial \phi^n}{\partial x}\right|_i\right)$$
$$\phi_i^{(2)} = \phi_i^{(1)} - \alpha_{2,0}\left(a_i^n \Delta t \left.\frac{\partial \phi^n}{\partial x}\right|_i\right) - \alpha_{2,1}\left(a_i^n \Delta t \left.\frac{\partial \phi^{(1)}}{\partial x}\right|_i\right)$$
$$\phi_i^{n+1} = \phi_i^{(2)} - \alpha_{3,0}\left(a_i^n \Delta t \left.\frac{\partial \phi^n}{\partial x}\right|_i\right) - \alpha_{3,1}\left(a_i^n \Delta t \left.\frac{\partial \phi^{(1)}}{\partial x}\right|_i\right) - \alpha_{3,2}\left(a_i^n \Delta t \left.\frac{\partial \phi^{(2)}}{\partial x}\right|_i\right)$$

with $\alpha_{1,0} = 1$

$\alpha_{2,0} = \frac{3}{4}$; $\alpha_{2,1} = \frac{1}{4}$

$\alpha_{3,0} = -\frac{1}{12}$; $\alpha_{3,1} = -\frac{1}{12}$; $\alpha_{3,2} = \frac{3}{2}$

To do this we need the space derivative, obtained with the WENO-5 method.

For the space derivative: WENO-5 method.

- For $a_i^n > 0$

$$\frac{\partial \phi^n}{\partial x}\Big|_i^- = \frac{1}{12\Delta x}\left(-\Delta^+\phi_{i-2}^n + 7\Delta^+\phi_{i-1}^n + 7\Delta^+\phi_i^n - \Delta^+\phi_{i+1}^n\right)$$

$$- \Psi_{WENO}\left(\frac{\Delta^-\Delta^+\phi_{i-2}^n}{\Delta x}, \frac{\Delta^-\Delta^+\phi_{i-1}^n}{\Delta x}, \frac{\Delta^-\Delta^+\phi_i^n}{\Delta x}, \frac{\Delta^-\Delta^+\phi_{i+1}^n}{\Delta x}\right)$$

Nothe that:

$$-\Delta^+\phi_{i-2} + 7\Delta^+\phi_{i-1} + 7\Delta^+\phi_i - \Delta^+\phi_{i+1} =$$

$$= -\phi_{i-1} + \phi_{i-2} + \cancel{7\phi_i} - 7\phi_{i-1} + 7\phi_{i+1} - \cancel{7\phi_i} - \phi_{i+2} + \phi_{i+1}$$

$$= \phi_{i-2} - 8\phi_{i-1} + 8\phi_{i+1} - \phi_{i+2}.$$

Further,

$$\frac{\Delta^-\Delta^+\phi_i}{\Delta x} = \frac{\Delta^-(\phi_{i+1} - \phi_i)}{\Delta x} = \frac{\Delta^-\phi_{i+1} - \Delta^-\phi_i}{\Delta x} = \frac{\phi_{i+1} - \phi_i - \phi_i + \phi_{i-1}}{\Delta x}$$

$$= \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta x}$$

Hence,

$$\frac{\Delta^-\Delta^+\phi_{i-2}}{\Delta x} = \frac{\phi_{i-1} - 2\phi_{i-2} + \phi_{i-3}}{\Delta x},$$

$$\frac{\Delta^-\Delta^+\phi_{i-1}}{\Delta x} = \frac{\phi_i - 2\phi_{i-1} + \phi_{i-2}}{\Delta x},$$

$$\frac{\Delta^-\Delta^+\phi_{i+1}}{\Delta x} = \frac{\phi_{i+2} - 2\phi_{i+1} + \phi_i}{\Delta x}.$$

Thus,

$$\frac{\partial \phi^n}{\partial x}\bigg|_i^- = \frac{1}{12\Delta x}\left(\phi^n_{i-2} - 8\phi^n_{i-1} + 8\phi^n_{i+1} - \phi^n_{i+2}\right) - \Psi_{WENO}\left(\frac{\phi^n_{i-1} - 2\phi^n_{i-2} + \phi^n_{i-3}}{\Delta x}, \frac{\phi^n_{i} - 2\phi^n_{i-1} + \phi^n_{i-2}}{\Delta x}, \frac{\phi^n_{i+1} - 2\phi^n_{i} + \phi^n_{i-1}}{\Delta x}, \frac{\phi^n_{i+2} - 2\phi^n_{i+1} + \phi^n_{i}}{\Delta x}\right)$$

- <u>For $a_i^n < 0$</u>

$$\frac{\partial \phi^n}{\partial x}\bigg|_i^+ = \frac{1}{12\Delta x}\left(-\Delta^+\phi_{i-2} + 7\Delta^+\phi_{i-1} + 7\Delta^+\phi_i - \Delta^+\phi_{i+1}\right) + \Psi_{WENO}\left(\frac{\Delta^-\Delta^+\phi_{i+2}}{\Delta x}, \frac{\Delta^-\Delta^+\phi_{i+1}}{\Delta x}, \frac{\Delta^-\Delta^+\phi_{i}}{\Delta x}, \frac{\Delta^-\Delta^+\phi_{i-1}}{\Delta x}\right)$$

Mirroing the results from $a_i^n > 0$,

$$\frac{\partial \phi^n}{\partial x}\bigg|_i^+ = \frac{1}{12\Delta x}\left(\phi_{i-2} - 8\phi^n_{i-1} + 8\phi^n_{i+1} - \phi^n_{i+2}\right) + \Psi_{WENO}\left(\frac{\phi^n_{i+3} - 2\phi^n_{i+2} + \phi^n_{i+1}}{\Delta x}, \frac{\phi^n_{i+2} - 2\phi^n_{i+1} + \phi^n_{i}}{\Delta x}, \frac{\phi^n_{i+1} - 2\phi^n_{i} + \phi^n_{i-1}}{\Delta x}, \frac{\phi^n_{i} - 2\phi^n_{i-1} + \phi^n_{i-2}}{\Delta x}\right)$$

After every step of the RK method, we need to update the ghost cells using the BCs.

- ~~left~~ ghost cells: Dirichlet

MATLAB Indices

$$\phi_3^n = 2\phi_{BC}^n - \phi_4^n$$
$$\phi_2^n = 2\phi_{BC}^n - \phi_5^n$$
$$\phi_1^n = 2\phi_{BC}^n - \phi_6^n$$

where $\phi_{BC}^n = \phi_{BC}(t_n) = \phi(x=-1, t_n)$ given in the problem.

- Right ghost cells: zero Neumann

$$\phi_{M+4}^n = \phi_{M+3}^n$$
$$\phi_{M+5}^n = \phi_{M+2}^n$$
$$\phi_{M+6}^n = \phi_{M+1}^n.$$

Time step.

Stable time step for 1st order upwind:

$$\Delta t \le \frac{\Delta x}{a}$$

For our method we include the CFL factor and, since a depends on x ~~and~~ for each value of time,

$$\Delta t \le CFL \frac{\Delta x}{\max\limits_{1 \le i \le M+6} |a_i^n|}$$