```
% Lax-Friedrich to solve 1-D Euler equations with Mie-Gruneisen EoS
% Josh Bevan 2013
% Num Methods for PDEs, 22.520
% Based on code by Anand Dhariya, UMich:
% http://sitemaker.umich.edu/anand/home

clear
clc
close all
n=301;              %Number of grid points
L=1;                %Length of domain
h=L/(n-1);          %Spatial step size
CFL=0.95;           %CFL number for stability
t_final=150e-6;     %Final time
x=0:h:L;

%ICs
p_l=7.98e9;                     %Pressure t=0
p_r=0;
rho_l=4000;                     %Density t=0
rho_r=2785;
u_l=0;                          %Velocity t=0
u_r=-2000;
p(1:1:(n+1)/2)=p_l;
p((n+3)/2:1:n)=p_r;

p_0=p;                          %saved ICs

rho(1:1:(n+1)/2)=rho_l;
rho((n+3)/2:1:n)=rho_r;
rho_0=rho;                      %saved ICs
rho_nat=2785*(rho./rho);
u(1:1:(n+1)/2)=u_l;
u((n+3)/2:1:n)=u_r;
E=0.5*p.*(1./rho_nat-1./rho)+0.5*u.^2;     %Total Energy
a=5328*(rho./rho);              %Speed of sound
dt=CFL*h/max(abs(u+a));
step=0;

% Time integration begins
for t=dt:dt:t_final
    %Define q & F matrix
    q=[rho; rho.*u; rho.*E];
    F=[rho.*u; rho.*u.^2+p; u.*(rho.*E+p)];
    %Update q matrix and flow parameters
    q(1:3,2:n-1)=0.5*(q(1:3,3:n)+q(1:3,1:n-2))-dt/(2*h)*(F(1:3,3:n)-F(1:3,1:n-2));
    rho=q(1,1:n);
    u=q(2,1:n)./rho(1:n);
    E=q(3,1:n)./rho;
    p=((2*E-u.^2)/(1./rho_0-1./rho))-p_0;
    step=step+1;
```

```
end
%calculation of flow parameters
a=5328*(rho./rho); %sqrt(gamma*p./rho);
M=-u./a;
s=(log(p./p_0).*log(rho_0./rho))+rho; %Entropy w.r.t reference condition
Q=rho.*u;               %Mass Flow rate per unit area

%------------------------------------------------------------------------%
offset=0.05;
subplot(231);plot(x,p,'k');xlabel('X-Coordinate (m)');ylabel('Pressure (Pa)');ylim([min↙
(p)-(offset)*max(p) (1+offset)*max(p)]);
subplot(232);plot(x,s,'k');xlabel('X-Coordinate (m)');ylabel('Entropy');ylim([min(s)-↙
(offset)*max(s) (1+offset)*max(s)]);
subplot(233);plot(x,u,'k');xlabel('X-Coordinate (m)');ylabel('Velocity (m/s)');ylim([min↙
(u)-(offset)*max(u) (1+offset)*max(u)]);
subplot(234);plot(x,M,'k');xlabel('X-Coordinate (m)');ylabel('Mach number');ylim([min(M)↙
-(offset)*max(M) (1+offset)*max(M)]);
subplot(235);plot(x,rho,'k');xlabel('X-Coordinate (m)');ylabel('Density (kg/m^3)');ylim↙
([min(rho)-(offset)*max(rho) (1+offset)*max(rho)]);
subplot(236);plot(x,Q,'k');xlabel('X-Coordinate (m)');ylabel('Mass Flow (kg/m^2s)');ylim↙
([min(Q)-(offset)*max(Q) (1+offset)*max(Q)]);
```