```
clear all
clc
% clf

%% INITIALIZATION
%BASIC PARAMETERS
gamma = 1.4;
p4 = 2.0;
p1 = 1.0;
rho4 = 2.0;
rho1 = 1.0;
imax = 41;
xmin = 0;
xmax = 2.;
dx = 2/(imax-1);

global rho_vector
global p_vector
global velocity_vector


%SET UP VECTORS

x = 0:dx:2;
x0 = find(x==1.0);

u = zeros(1,imax);

%SET UP STATE VECTORS
USTATE = zeros(3,imax);
USTATE_UPDATE = zeros(3,imax);
F_STATE = zeros(3,imax);
F_STATEPLUS = zeros(3,imax);
F_STATEMINUS = zeros(3,imax);


% USTATE_PLUSONE= zeros(3,imax);
% USTATE_MINUSONE= zeros(3,imax);

%SET UP MATRICES

lamda_plus_i_plus = zeros(3,3);
lamda_minus_i_plus = zeros(3,3);
c_half_plus = zeros(3,3);
c_inv_half_plus = zeros(3,3);
s_half_plus = zeros(3,3);
s_inv_half_plus = zeros(3,3);
FPLUS = zeros(3,imax);
xplus = zeros(3,3);
xplusinv = zeros(3,3);
xminus= zeros(3,3);
xminusinv= zeros(3,3);
```

```
% FPLUS = zeros(3,3);


lamda_plus_i_minus = zeros(3,3);
lamda_minus_i_minus = zeros(3,3);
Ca_i_minus = zeros(3,3);
Ca_inverse_minus = zeros(3,3);
S_i_minus = zeros(3,3);
S_inverse_minus = zeros(3,3);
FMINUS = zeros(3,imax);
Abarplus_plus= zeros(3,3);
Abarminus_plus= zeros(3,3);

Abarplus_minus= zeros(3,3);
Abarminus_minus= zeros(3,3);




%% ROE STUFF

Big_lamda_plus_plus_half = zeros(3,3);
Big_lamda_minus_plus_half = zeros(3,3);


Big_lamda_plus_minus_half = zeros(3,3);
Big_lamda_minus_minus_half = zeros(3,3);

Jacobi_hat_plus_plus = zeros(3,3);
Jacobi_hat_minus_plus = zeros(3,3);

Jacobi_hat_plus_plus = zeros(3,3);
Jacobi_hat_minus_minus = zeros(3,3);

s_half_plus = zeros(3,3);
c_half_plus = zeros(3,3);
s_inv_half_plus = zeros(3,3);
c_inv_half_plus = zeros(3,3);

Jacobi_hat_plus_minus = zeros(3,3);
s_half_minus = zeros(3,3);
c_half_minus = zeros(3,3);
s_inv_half_minus = zeros(3,3);
c_inv_half_minus = zeros(3,3);

A_HAT_PLUS = zeros(3,3);
A_HAT_MINUS = zeros(3,3); %equal Aplusplus half - Aminusplus half



%%
```

```
% INITIAL CONDITIONS
% RHO
USTATE(1,1:x0) = rho4;
USTATE(1,x0:imax) = rho1;


% FOR E
USTATE(3,1:x0) = p4/(gamma-1);
USTATE(3,x0:imax) = p1/(gamma-1);

USTATE_UPDATE = USTATE;

timestep = 0;
maxtimestep = 19;


%% MAIN LOOP
while timestep <maxtimestep
  %% PART 1: CALCULATE DT


  for i = 1:imax
     u_at_i = USTATE_UPDATE(2,:)./USTATE_UPDATE(1,:); %%rho u by rho
     rho_at_i = USTATE_UPDATE(1,:);
     e_at_i = USTATE_UPDATE(3,:);
     p_at_i = (e_at_i-0.5.*rho_at_i.*u_at_i.^2);
     a_at_i = sqrt(gamma*p_at_i./rho_at_i);
     abs_u_plus_a_at_i= abs(u_at_i+a_at_i);
     dt_at_i = dx./abs_u_plus_a_at_i;%this is an array of dt at i
     dt_smallest = min(dt_at_i); %smallest dt in dt array at i
     real_dt = 0.9*dt_smallest;

  end

  dt = real_dt;


   %% PART 2: DEAL WITH U PLUS HALF
   for i = 2:imax-1
      USTATE_PLUSONE(:,:) = USTATE(:,i+1);  %at i+1

      USTATE_I(:,:) = USTATE(:,i);

      rho_plus = USTATE_PLUSONE(1,:);   %at i + 1

      rho_i = USTATE_I(1,:); %at i

      u_plus = USTATE_PLUSONE(2,:)./USTATE_PLUSONE(1,:);

      u_i = USTATE_I(2,:)./USTATE_I(1,:);
```

```
e_plus = USTATE_PLUSONE(3,:);
e_i = USTATE_I(3,:);

p_plus = (gamma-1).*e_plus+(0.5).*rho_plus.*u_plus.^2;
p_i = (gamma-1).*e_i+(0.5).*rho_i.*u_i.^2;

h_plus_one = e_plus+(p_plus)./(rho_plus);
h_i = e_i + (p_i)/(rho_i);

rho_hat_plus = sqrt(rho_i).*sqrt(rho_plus);
u_hat_plus = (sqrt(rho_i).*u_i+(sqrt(rho_plus).*u_plus))./ (sqrt(rho_i) …

h_hat_plus = (sqrt(rho_i).*(h_i) + sqrt(rho_plus).*(h_plus_one)) / (sqrt…

a_hat_plus = sqrt((gamma-1)*(h_hat_plus-0.5*u_hat_plus.^2));

%Form lamda minus at plus half
lamda1_plus_half = u_hat_plus;
lamda2_plus_half = u_hat_plus+a_hat_plus;
lamda3_plus_half = u_hat_plus-a_hat_plus;

lamda1_n_plus_half = 0.5*(lamda1_plus_half-abs(lamda1_plus_half));
lamda2_n_plus_half = 0.5*(lamda2_plus_half-abs(lamda2_plus_half));
lamda3_n_plus_half = 0.5*(lamda3_plus_half-abs(lamda3_plus_half));

lamda1_p_plus_half = 0.5*(lamda1_plus_half+abs(lamda1_plus_half));
lamda2_p_plus_half = 0.5*(lamda2_plus_half+abs(lamda2_plus_half));
lamda3_p_plus_half = 0.5*(lamda3_plus_half+abs(lamda3_plus_half));
Big_lamda_minus_plus_half(1,1) = lamda1_n_plus_half;
Big_lamda_minus_plus_half(2,2) = lamda2_n_plus_half;
Big_lamda_minus_plus_half(3,3) = lamda3_n_plus_half;

Big_lamda_plus_plus_half(1,1) = lamda1_p_plus_half;
Big_lamda_plus_plus_half(2,2) = lamda2_p_plus_half;
Big_lamda_plus_plus_half(3,3) = lamda3_p_plus_half;

c_half_plus(1,1) = 1.0;
c_half_plus(1,2) = 0.0;

c_half_plus(1,3) = -1./(a_hat_plus.^2);
c_half_plus(2,1) = 0.0;
c_half_plus(2,2) = rho_hat_plus*a_hat_plus;
c_half_plus(2,3) = 1.0;

c_half_plus(3,1) = 0.0;
c_half_plus(3,2) = -rho_hat_plus*a_hat_plus;
c_half_plus(3,3) = 1.0;

beta = gamma-1;
alpha_hat_plus = (u_hat_plus.^2)./2;
s_half_plus(1,1) = 1.0;
s_half_plus(1,2) = 0.0;
```

```
    s_half_plus(1,3) = 0.0;
    s_half_plus(2,1) = -u_hat_plus./rho_hat_plus;
    s_half_plus(2,2) = 1.0./rho_hat_plus;
    s_half_plus(2,3) = 0.0;
    s_half_plus(3,1) = alpha_hat_plus.*beta;
    s_half_plus(3,2) = -u_hat_plus.*beta;
    s_half_plus(3,3) = beta;

    c_inv_half_plus(1,1) = 1.0;
    c_inv_half_plus(1,2) = 1.0./(2.*a_hat_plus.^2);
    c_inv_half_plus(1,3) = 1.0./(2.*a_hat_plus.^2);

    c_inv_half_plus(2,1) = 0.0;

    c_inv_half_plus(2,2) = 1.0./(2.*rho_hat_plus.*a_hat_plus);
    c_inv_half_plus(2,3) = -1.0./(2.*rho_hat_plus.*a_hat_plus);

    c_inv_half_plus(3,1) = 0.0;
    c_inv_half_plus(3,2) = 0.5;
    c_inv_half_plus(3,3) = 0.5;
    s_inv_half_plus(1,1) = 1.0;
    s_inv_half_plus(1,2) = 0.0;
    s_inv_half_plus(1,3) = 0.0;
    s_inv_half_plus(2,1) = u_hat_plus;
    s_inv_half_plus(2,2) = rho_hat_plus;
    s_inv_half_plus(2,3) = 0.0;
    s_inv_half_plus(3,1) = alpha_hat_plus;
    s_inv_half_plus(3,2) = rho_hat_plus.*u_hat_plus;
    s_inv_half_plus(3,3) = 1.0./beta;


%% PART 3: NOW DEAL WITH I MINUS HALF


    USTATE_MINUSONE(:,:) = USTATE(:,i-1);  %at i+1

    rho_minus = USTATE_MINUSONE(1,:);   %at i + 1

    u_minus = USTATE_MINUSONE(2,:)./USTATE_MINUSONE(1,:);

    e_minus = USTATE_MINUSONE(3,:);

    p_minus = (gamma-1).*e_minus+(0.5).*rho_minus.*u_minus.^2;

    h_minus_one = e_minus+(p_minus)./(rho_minus);
    rho_hat_minus = sqrt(rho_i).*sqrt(rho_minus);
    u_hat_minus = (sqrt(rho_minus).*u_minus+(sqrt(rho_i).*u_i))./ (sqrt(rho_mi…
    h_hat_minus = (sqrt(rho_minus).*(h_minus_one) + sqrt(rho_i).*(h_i)) / (sqr…

    a_hat_minus = sqrt((gamma-1)*(h_hat_minus-0.5*u_hat_minus.^2));

    %    Form lamda plus at minus half
```

```
lamda1_minus_half = u_hat_minus;
lamda2_minus_half = u_hat_minus+a_hat_minus;
lamda3_minus_half = u_hat_minus-a_hat_minus;

lamda1_p_minus_half = 0.5*(lamda1_minus_half+abs(lamda1_minus_half));
lamda2_p_minus_half = 0.5*(lamda2_minus_half+abs(lamda2_minus_half));
lamda3_p_minus_half = 0.5*(lamda3_minus_half+abs(lamda3_minus_half));

Big_lamda_plus_minus_half(1,1) = lamda1_p_minus_half;
Big_lamda_plus_minus_half(2,2) = lamda2_p_minus_half;
Big_lamda_plus_minus_half(3,3) = lamda3_p_minus_half;

lamda1_n_minus_half = 0.5*(lamda1_minus_half-abs(lamda1_minus_half));
lamda2_n_minus_half = 0.5*(lamda2_minus_half-abs(lamda2_minus_half));
lamda3_n_minus_half = 0.5*(lamda3_minus_half-abs(lamda3_minus_half));

Big_lamda_minus_minus_half(1,1) = lamda1_n_minus_half;
Big_lamda_minus_minus_half(2,2) = lamda2_n_minus_half;
Big_lamda_minus_minus_half(3,3) = lamda3_n_minus_half;

c_half_minus(1,1) = 1.0;
c_half_minus(1,2) = 0.0;

c_half_minus(1,3) = -1./(a_hat_plus.^2);
c_half_minus(2,1) = 0.0;
c_half_minus(2,2) = rho_hat_minus*a_hat_minus;
c_half_minus(2,3) = 1.0;

c_half_minus(3,1) = 0.0;
c_half_minus(3,2) = -rho_hat_minus*a_hat_minus;
c_half_minus(3,3) = 1.0;

beta = gamma-1;
alpha_hat_minus = (u_hat_minus.^2)./2;
s_half_minus(1,1) = 1.0;
s_half_minus(1,2) = 0.0;
s_half_minus(1,3) = 0.0;
s_half_minus(2,1) = -u_hat_minus./rho_hat_minus;
s_half_minus(2,2) = 1.0./rho_hat_minus;
s_half_minus(2,3) = 0.0;
s_half_minus(3,1) = alpha_hat_minus.*beta;
s_half_minus(3,2) = -u_hat_minus.*beta;
s_half_minus(3,3) = beta;

c_inv_half_minus(1,1) = 1.0;
c_inv_half_minus(1,2) = 1.0./(2.*a_hat_minus.^2);
c_inv_half_minus(1,3) = 1.0./(2.*a_hat_minus.^2);

c_inv_half_minus(2,1) = 0.0;
c_inv_half_minus(2,2) = 1.0./(2.*rho_hat_minus.*a_hat_minus);
c_inv_half_minus(2,3) = -1.0./(2.*rho_hat_minus.*a_hat_minus);
```

```
c_inv_half_minus(3,1) = 0.0;
c_inv_half_minus(3,2) = 0.5;
c_inv_half_minus(3,3) = 0.5;

s_inv_half_minus(1,1) = 1.0;
s_inv_half_minus(1,2) = 0.0;
s_inv_half_minus(1,3) = 0.0;
s_inv_half_minus(2,1) = u_hat_minus;
s_inv_half_minus(2,2) = rho_hat_minus;
s_inv_half_minus(2,3) = 0.0;
s_inv_half_minus(3,1) = alpha_hat_minus;
s_inv_half_minus(3,2) = rho_hat_minus.*u_hat_minus;
s_inv_half_minus(3,3) = 1.0./beta;

%% CALCULATE FLUX F VECTOR

F_STATE(1,:)= rho_i.*u_i;
F_STATE(2,:)= rho_i.*u_i.^2 + p_i;
F_STATE(3,:) = (e_i+p_i).*u_i;

F_STATEPLUS(1,:)= rho_plus.*u_plus;
F_STATEPLUS(2,:)= rho_plus.*u_plus.^2 + p_plus;
F_STATEPLUS(3,:) = (e_plus+p_plus).*u_plus;


F_STATEMINUS(1,:) = rho_minus.*u_minus;
F_STATEMINUS(2,:) = rho_minus.*u_minus.^2+p_minus;
F_STATEMINUS(3,:) = (e_minus+p_minus).*u_minus;


%% JACOBI

Jacobi_hat_minus_plus(:,:) = s_inv_half_plus(:,:)*c_inv_half_plus(:,:)*Big…
Jacobi_hat_plus_plus(:,:) = s_inv_half_plus(:,:)*c_inv_half_plus(:,:)*Big_…

Jacobi_hat_plus_minus(:,:) = s_inv_half_minus(:,:)*c_inv_half_minus(:,:)*B…
Jacobi_hat_minus_minus(:,:) = s_inv_half_minus(:,:)*c_inv_half_minus(:,:)*…


%% A HAT PLUS AND A HAT MINUS (inside abs like sign)

A_HAT_MINUS(:,:)= Jacobi_hat_plus_minus(:,:) - Jacobi_hat_minus_minus(:,:);

A_HAT_PLUS(:,:) = Jacobi_hat_plus_plus(:,:)-Jacobi_hat_minus_plus(:,:);

%% FPLUS AND FMINUS

FPLUS(:,i) = 0.5*(F_STATE(:,i)+F_STATEPLUS(:,i)) - (0.5*A_HAT_PLUS(:,:)*(U…
FMINUS(:,i) = 0.5*(F_STATEMINUS(:,i)+F_STATE(:,i)) - (0.5*A_HAT_MINUS(:,:)…

%% FINITE DIFFERENCE EQUATION
```

```
    USTATE_UPDATE(:,i) = USTATE(:,i) - (dt/dx)*(FPLUS(:,i)-FMINUS(:,i));

   end

   %% SET BC AND PLOT VARIABLES FOR NUMERICAL

    USTATE_UPDATE(:,imax) = USTATE_UPDATE(:,imax-1);
    USTATE = USTATE_UPDATE;
    ENERGY = USTATE_UPDATE(3,:);
    RHO = USTATE_UPDATE(1,:);

    VELOCITY = USTATE_UPDATE(2,:)./RHO;
    PRESSURE = ((gamma-1)*ENERGY-(gamma-1)*0.5.*(((USTATE_UPDATE(2,:)).^2)./(R…
%
  %% PART 5: ANALYTICAL PART  (call analytical, use updated dt);

  max_ANALYTICAL_shock_tube(dt);
  timestep = timestep+1;
end


%% PLOTTING

figure(1)

  plot(x,PRESSURE);
  ylim([0.5 2]);
  hold on
  grid on
  plot(x,p_vector);
  legend({'ROE','Analytical'},'FontSize',14);
  xlabel('X','FontSize',18);
  title('X vs PRESSURE ROE EXPLICIT SCHEME','FontSize',18);
  ylabel('PRESSURE','FontSize',18);
  xt = get(gca, 'XTick');
  set(gca, 'FontSize', 16)

  figure(2)
  plot(x,RHO);
  ylim([0.5 2]);
  hold on
  grid on
  plot(x,rho_vector);
  legend({'ROE','Analytical'},'FontSize',14);
  xlabel('X','FontSize',18);
  title('X vs DENSITY ROE EXPLICIT SCHEME','FontSize',18)
  ylabel('DENSITY','FontSize',18);
  xt = get(gca, 'XTick');
  set(gca, 'FontSize', 16);

  figure(3)
  plot(x,VELOCITY);
```

```
hold on
grid on
plot(x,velocity_vector);
legend({'ROE','Analytical'},'FontSize',14);
xlabel('X','FontSize',18);
ylabel('VELOCITY','FontSize',18);
title('X vs VELOCITY ROE EXPLICIT SCHEME','FontSize',18)
xt = get(gca, 'XTick');
set(gca, 'FontSize', 16)
```