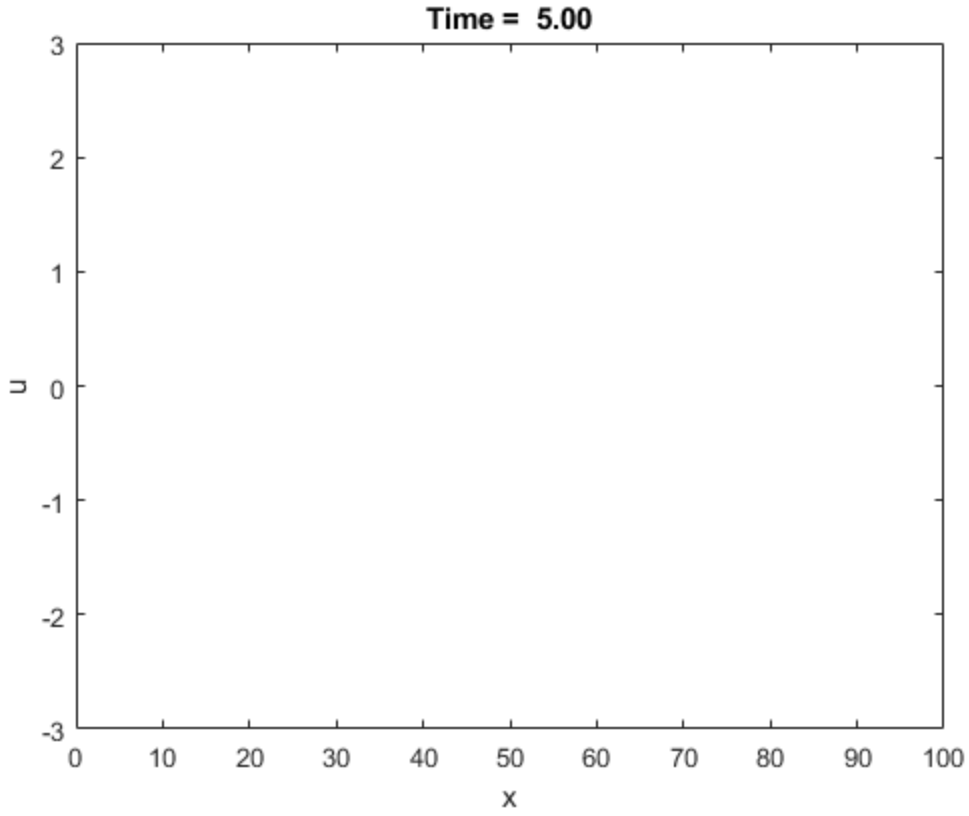
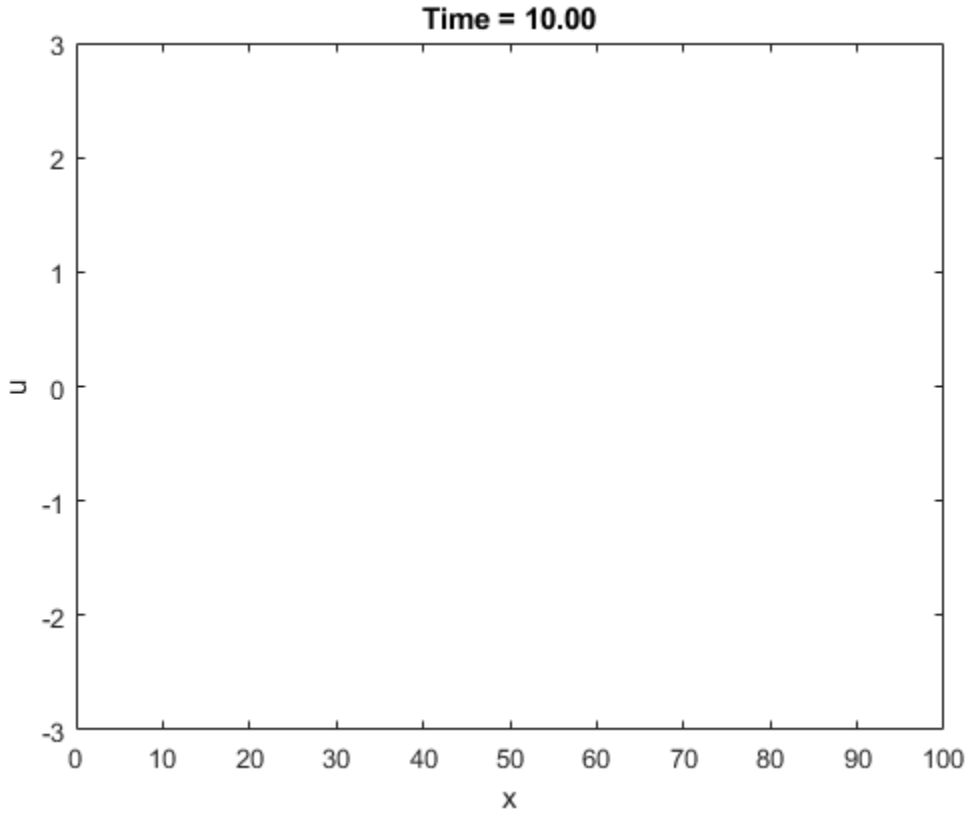
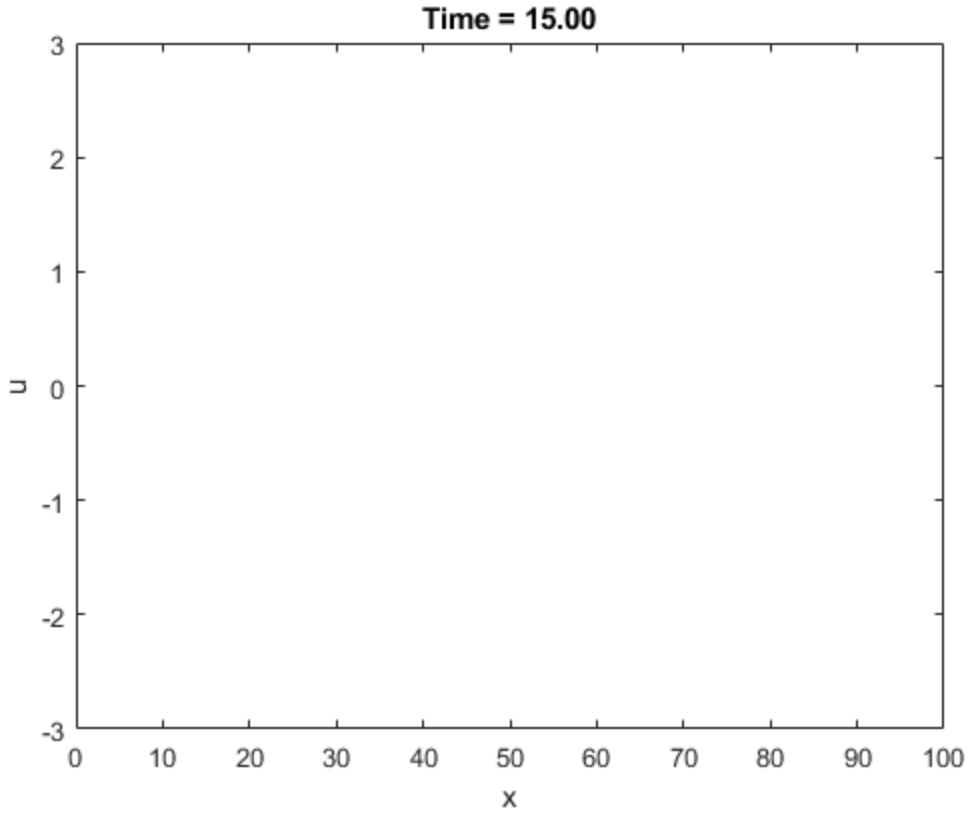
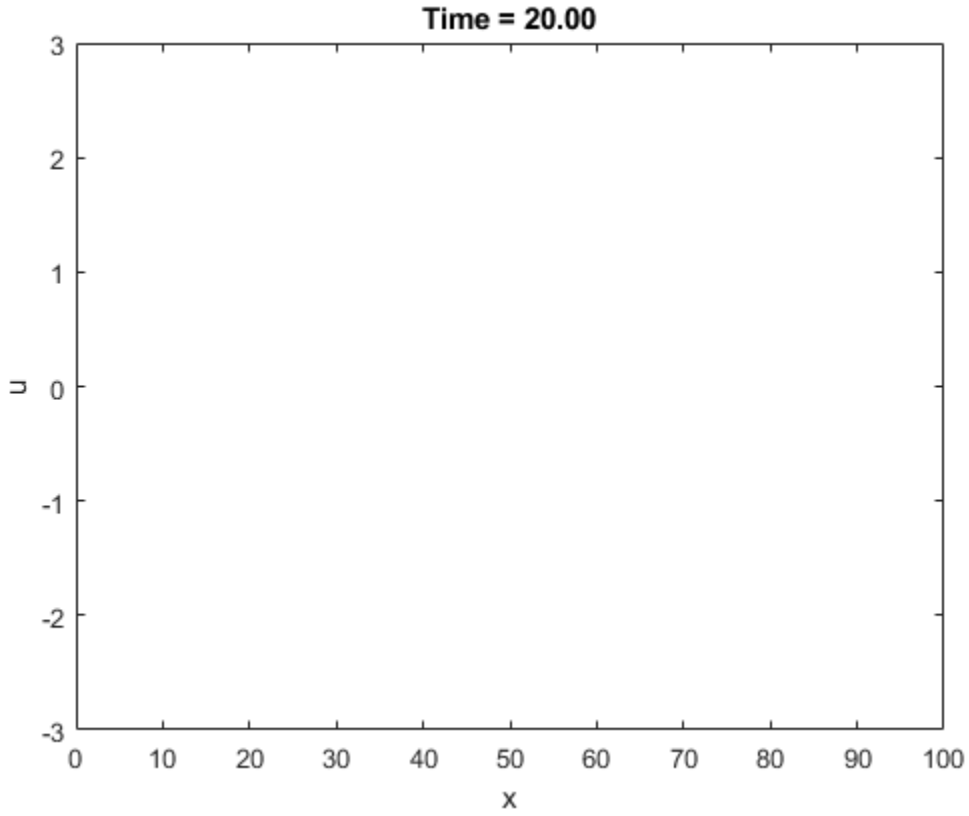
```
function Burgers_IMEXRKCB3c
% function <a href="matlab:Burgers_IMEXRKCB3c">Burgers_IMEXRKCB3c</a>
% Simulate the 1D Burgers on 0<x<L with homogeneous Dirichlet BCs using CN/
RKW3 in time

% (explicit on nonlinear terms, implicit on linear terms)

% Initialize the simulation parameters (user input)
L = 100;
Tmax = 50;
N = 100;
dt = 0.5;
PlotInterval=10;
dx = L / N;
x = (0:N) .* dx; % length N + 1
% STEP 1: Discretization of unknown variable on spatial grid
u = -sin(pi * x / L) - sin(2 * pi * x / L) + sin(6 * pi * x / L);
figure(1);
NR_PlotXY(x,u,0,0,L,-3,3)

% Precalculate the time-stepping coefficients used in the simulation
% Butcher tableau of IMEXRKCB3c from CB15.pdf
% Last two characters bt => Butcher tableau
bbt= [0, 673488652607 / 2334033219546, 493801219040 / 853653026979,
 184814777513 / 1389668723319];
cbt = [0, 3375509829940 / 42525919076317, 272778623835 / 1039454778728, 1];
a_exbt = [0, 0, 0, 0; ...
    cbt(2), 0, 0, 0 ; ...
    0, cbt(3), 0, 0; ...
    bbt];
a_imbt = [0, 0, 0, 0; ...
    0, 3375509829940 / 4525919076317, 0, 0;
    0, 11712383888607531889907 / 3269457049560210555624 8, 566138307881 /
 912153721139, 0; ...
    bbt(1), bbt(2), 1660544566939 / 2334033219546, 0];

cbt = [0, 3375509829940 / 42525919076317, 272778623835 / 1039454778728, 1];
h_bar = dt .* [cbt(2), cbt(3) - cbt(2), cbt(3) - cbt(2), 1 - cbt(4)];
% betabar = [aimbt(2, 1) / cbt(2), aimbt(3, 2) / (cbt(3) - cbt(2)), bbt(3) /
 (1 - cbt(3))];
zeta_bar = [0, zeta(2) / (cbt(3) - cbt(2)), zeta(3) / (cbt(4) - cbt(3)), ...
    zeta(4) / (1 - cbt(4))];
f = zeta_bar.*h_bar/(2*dx);
dxsquared = (dx)^2;
dxmult2 = 2 * dx;
atdiag = -h_bar ./ dxmult2;
btdiag = 1 + h_bar ./ dxsquared;
ctdiag = -h_bar ./ dxmult2;
y = zeros(size(x));
z = zeros(size(x));
% A = diag(btdiag .* ones(1, N + 1)) + diag(atdiag .* ones(1, N), -1) + ...
%     diag(ctdiag .* ones(1, N), 1);
```

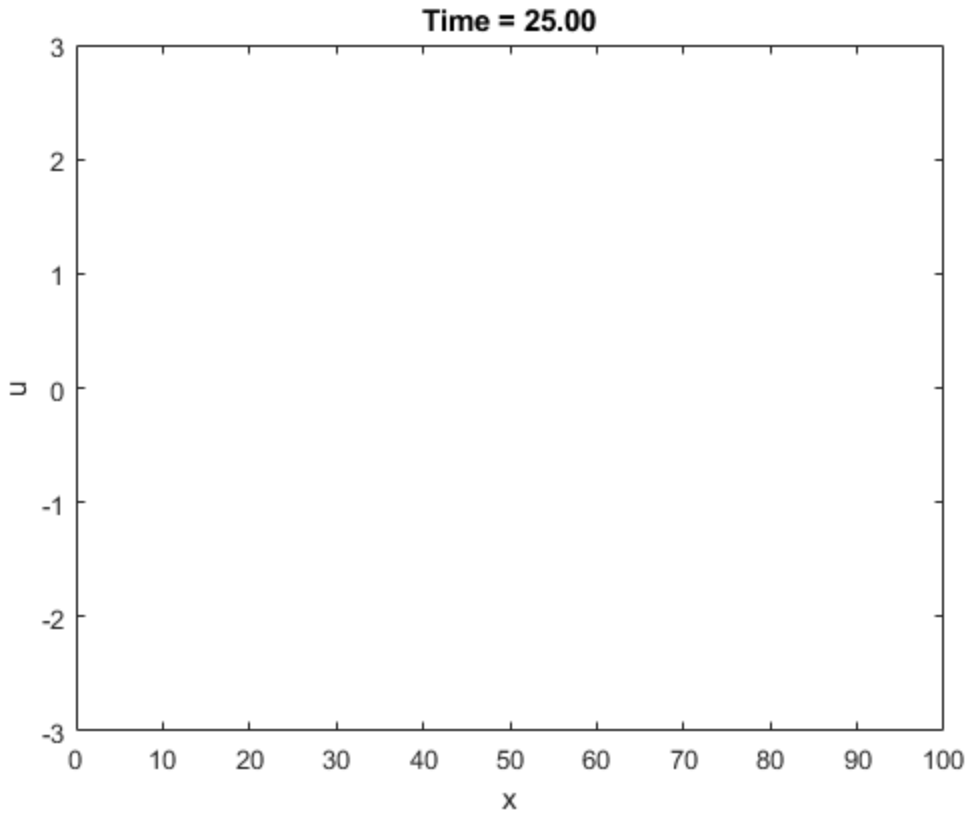
```
for tStep= 1:Tmax / dt
    for k = 1:4 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ALL 4 RK SUBSTEPS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        r = -u(2:N) .* (u(3:N + 1) - u(1:N - 1));   % nonlinear term
        if k == 1 % register 1
            y(2:N) = u(2:N);
        else    % register 2
            y(2:N) = u(2:N) + (a_imbt(k, k - 1) - bbt(k - 1)) .* dt .* ...
                (u(3:N + 1) - 2 * u(2:N) + u(1:N - 1)) + ...
                (a_exbt(k, k - 1) - bbt(k - 1)) .* dt .* r + f(k) .* y(2:N);
        end
        z(2:N) = (NR_ThomasTT(-a_imbt(k,k) * dt / (2 * dxsquared), ...
            1 + a_imbt(k,k) * dt / dxsquared, -a_imbt(k, k) * dt / (2 *
 dxsquared), ...
            y(2:N)', N - 1))' .* ...
            y(2:N);
        y(2:N) = -(y(2:N) + a_imbt(k, k) .* z(2:N)) .* ...
            (y(3:N + 1) + a_imbt(k, k) .* z(3:N + 1) - ...
            (y(1:N - 1) + a_imbt(k, k) .* z(1:N - 1)));

        u(2:N) = u(2:N) + bbt(k) * dt .* z(2:N) + bbt(k) * dt .* y(2:N);
    end %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% END OF RK LOOP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % plot
    if (mod(tStep,PlotInterval)==0)
        figure(tStep);
        NR_PlotXY(x,u,tStep*dt,0,L,-3,3);
    end
end
end
```

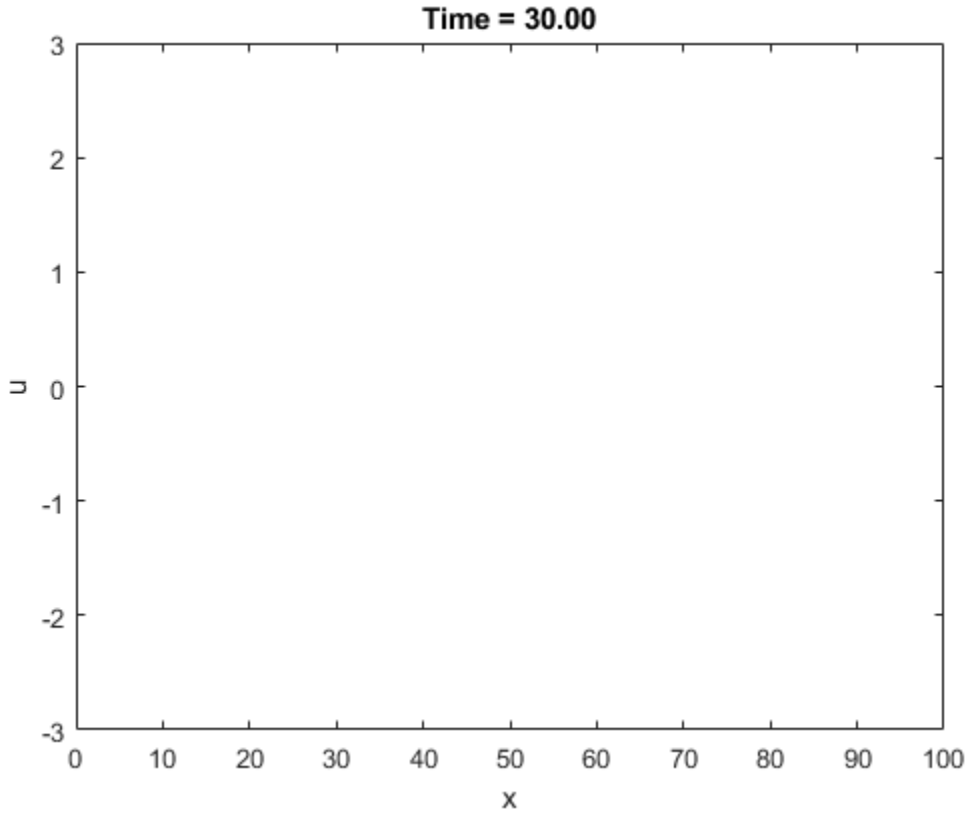

Time = 5.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x

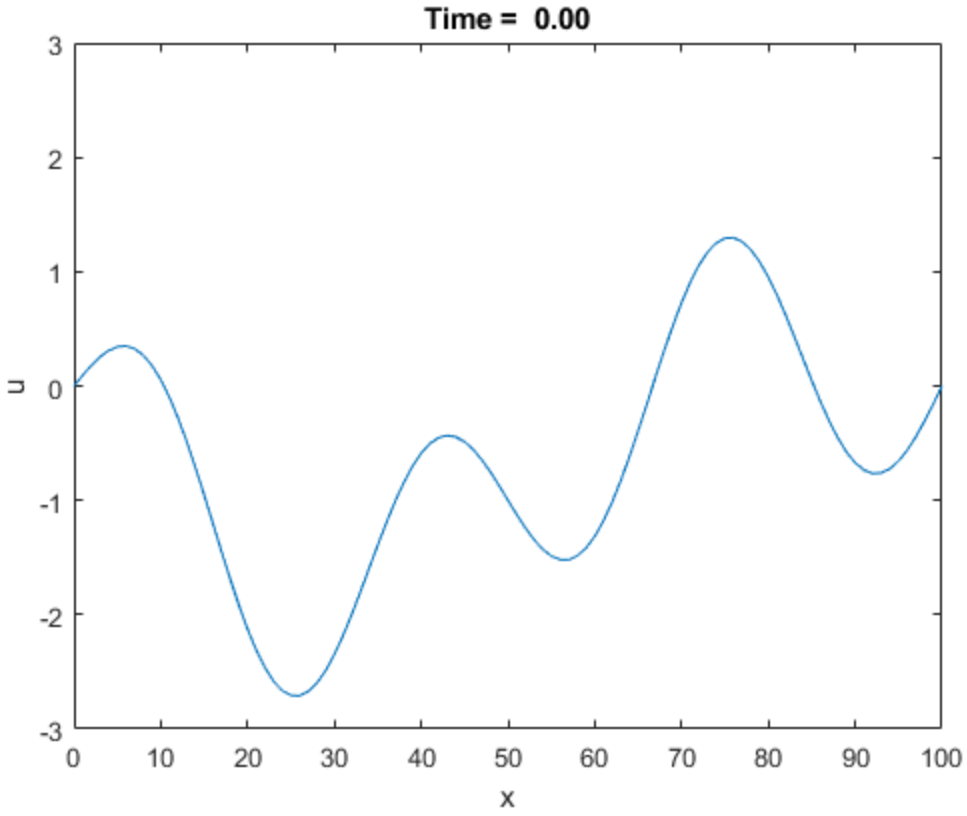

Time = 10.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x

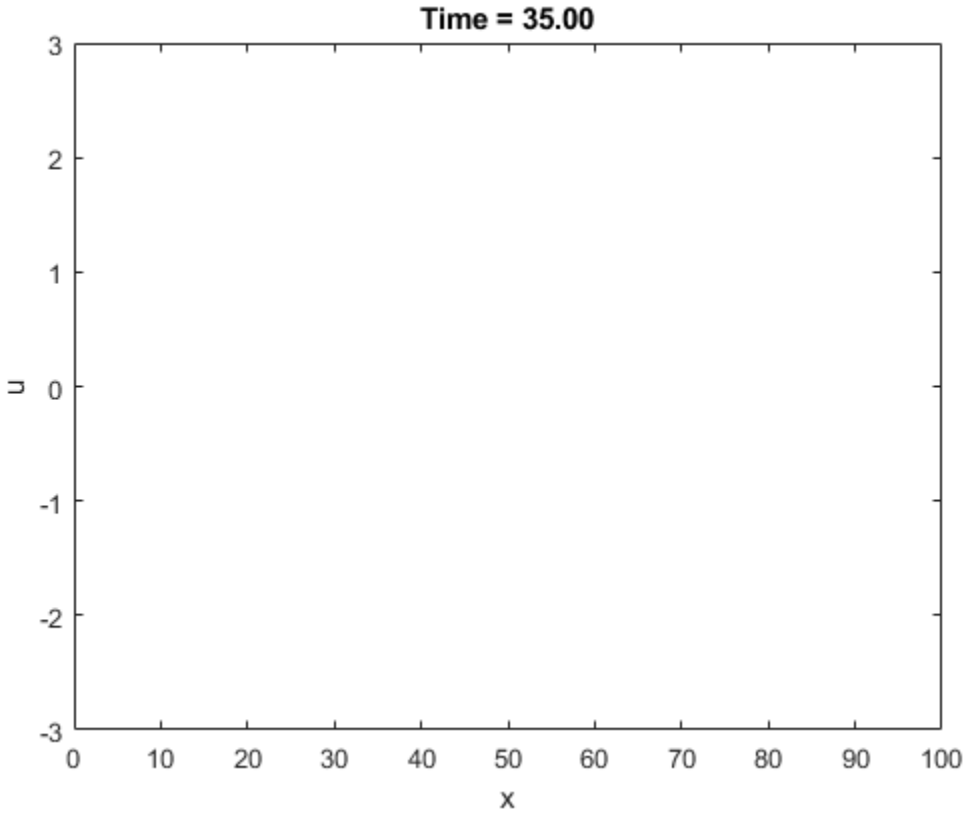

Time = 15.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x

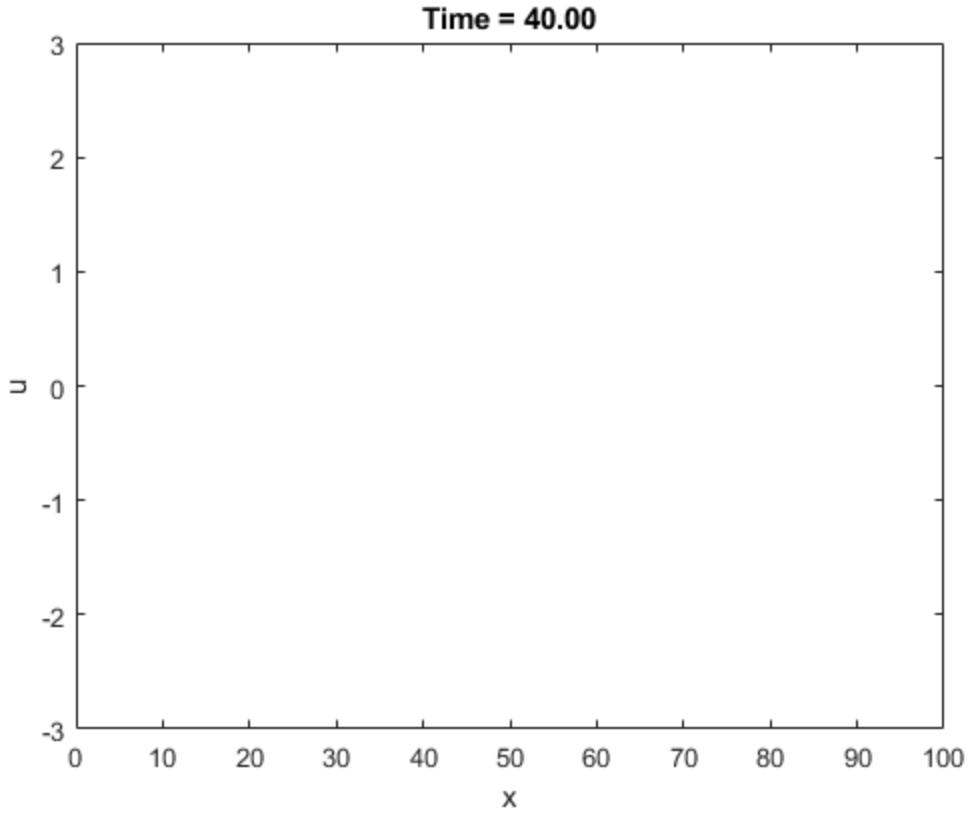

Time = 20.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x

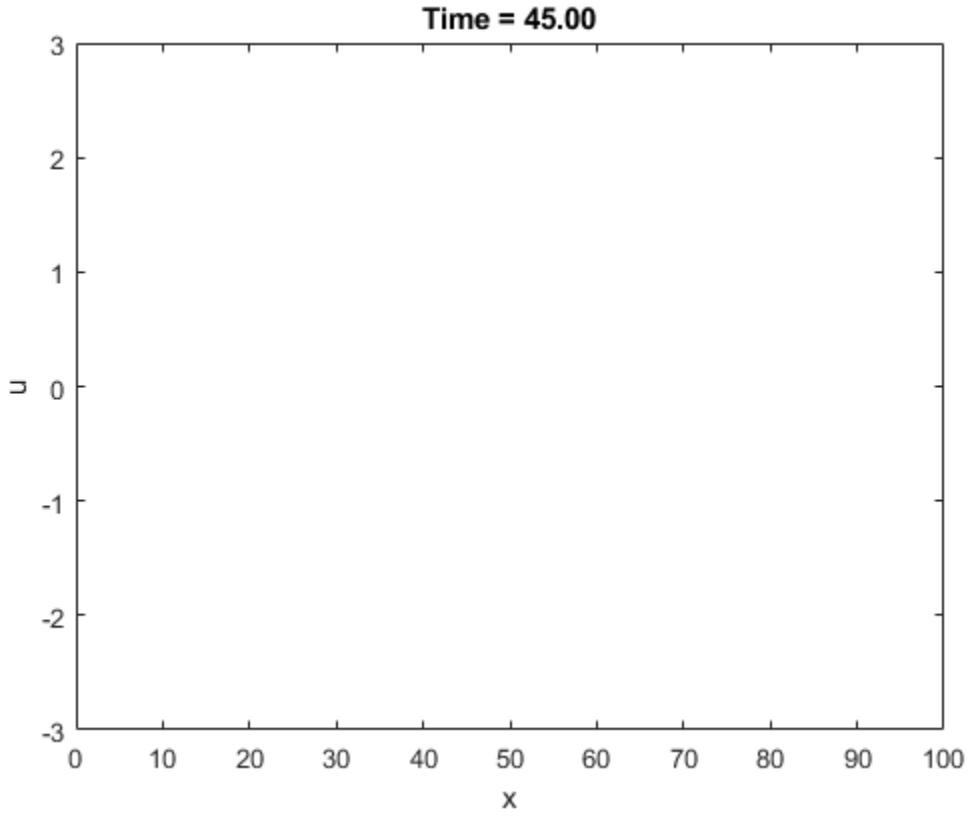

Time = 25.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x

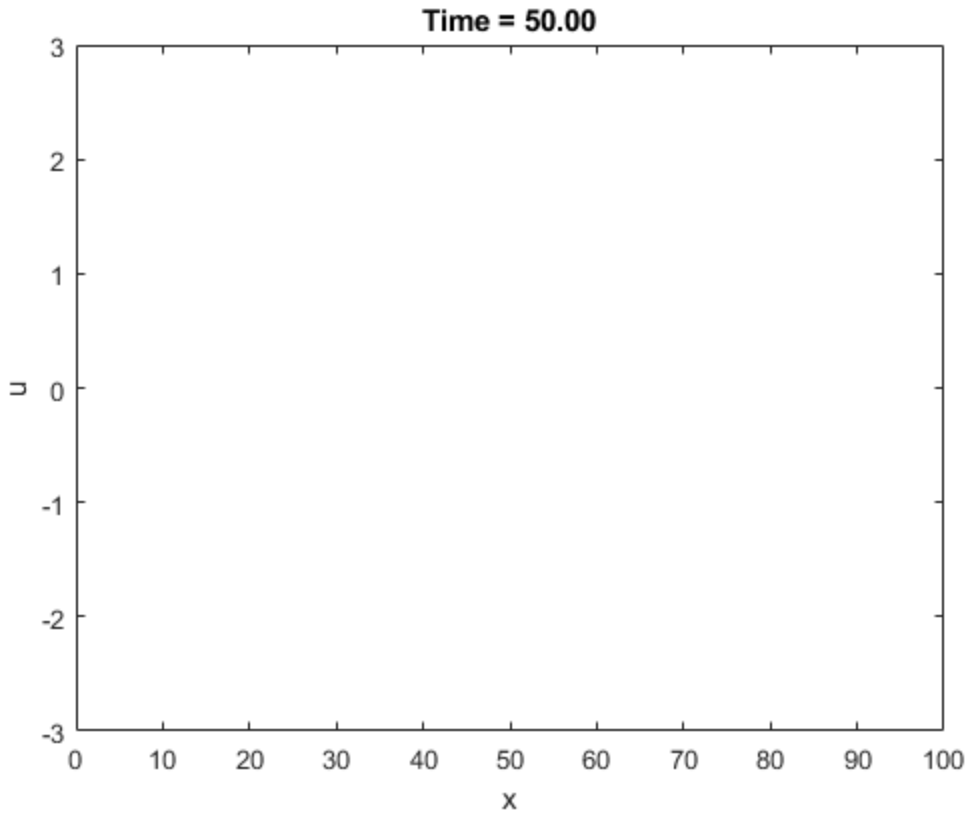

Time = 30.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x


Time = 0.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x


Time = 35.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x


Time = 40.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x


Time = 45.00
3
2
1
0
-1
-2
-3
u
0
10
20
30
40
50
60
70
80
90
100
x



Published with MATLAB® R2022a