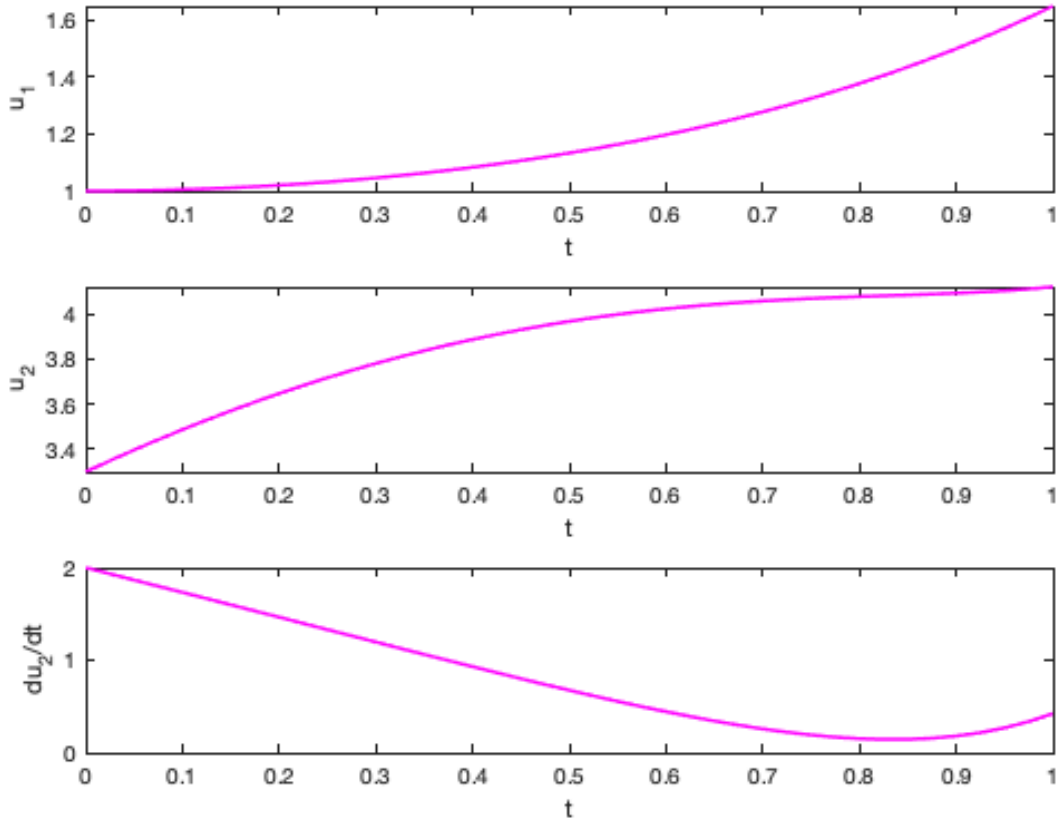
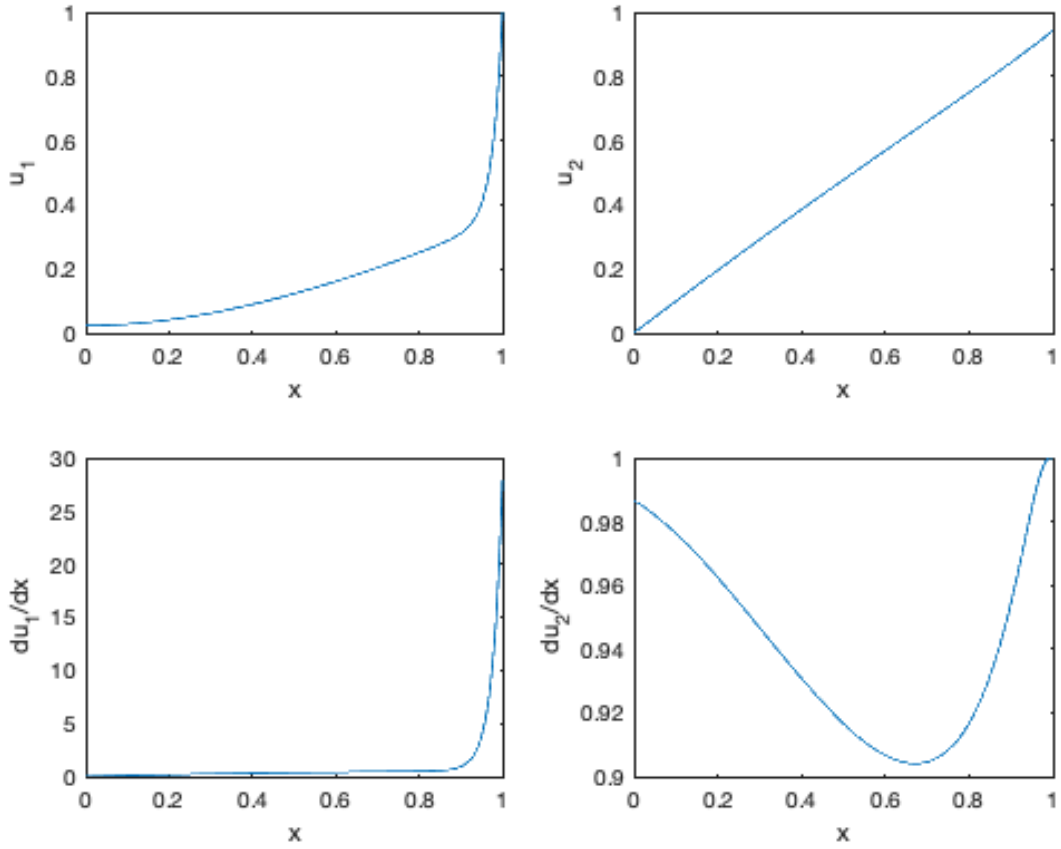
```
% recast the system into a first-order system by introducing du_2/dt=u_3

tspan=[0 1];
yzero=[1;3.3;2];
[tt,yy]=ode45(@myode,tspan,yzero);
figure;
subplot(3,1,1)
plot(tt,yy(:,1),'m','LineWidth',1.5),xlabel('t'),ylabel('u_1')
subplot(3,1,2)
plot(tt,yy(:,2),'m','LineWidth',1.5),xlabel('t'),ylabel('u_2')
subplot(3,1,3)
plot(tt,yy(:,3),'m','LineWidth',1.5),xlabel('t'),ylabel('du_2/dt')



%Q2: recast the equations into first-order system by introducing du1/dx=u3 and
 du2/dx=u4;
%the order of the variable could be different, here I use
%u_1,u_2,u_1',u_2'.
x = linspace(0,1,1000);
s0 = bvpinit(x,@myigs);
s = bvp4c(@myde,@mybc,s0);
figure;
subplot(2,2,1)
plot(s.x,s.y(1,:),'LineWidth',1.5)
xlabel('x');
ylabel('u_1');
subplot(2,2,2)
plot(s.x,s.y(2,:),'LineWidth',1.5)
xlabel('x');
ylabel('u_2');
subplot(2,2,3)
plot(s.x,s.y(3,:),'LineWidth',1.5)
xlabel('x');
ylabel('du_1/dx');
subplot(2,2,4)
plot(s.x,s.y(4,:),'LineWidth',1.5)
xlabel('x');
ylabel('du_2/dx');

%function for Q1
function sol=myode(t,y)
a=1/4;
b=4/5;
c=8/3;

sol=[t*y(1);
    y(3);
    y(2)*(t*y(1)-a*y(2)+b)^2-c*y(1)];% here we use tu_1 to replace du_1/dt
end
```

```
% function for Q2

function y = myigs(x)
  y = [1; 1;1;1];
end

function f = myde(x,y)

  f = [y(3); y(4);(y(3)+sin(y(1)-y(2)))/0.024;(y(4)-cos(y(1)-y(2)))/0.17];
end

function r = mybc(ya,yb)
 r = [ya(2);ya(3);yb(1)-1;yb(4)-1];
end
```





Published with MATLAB® R2021b