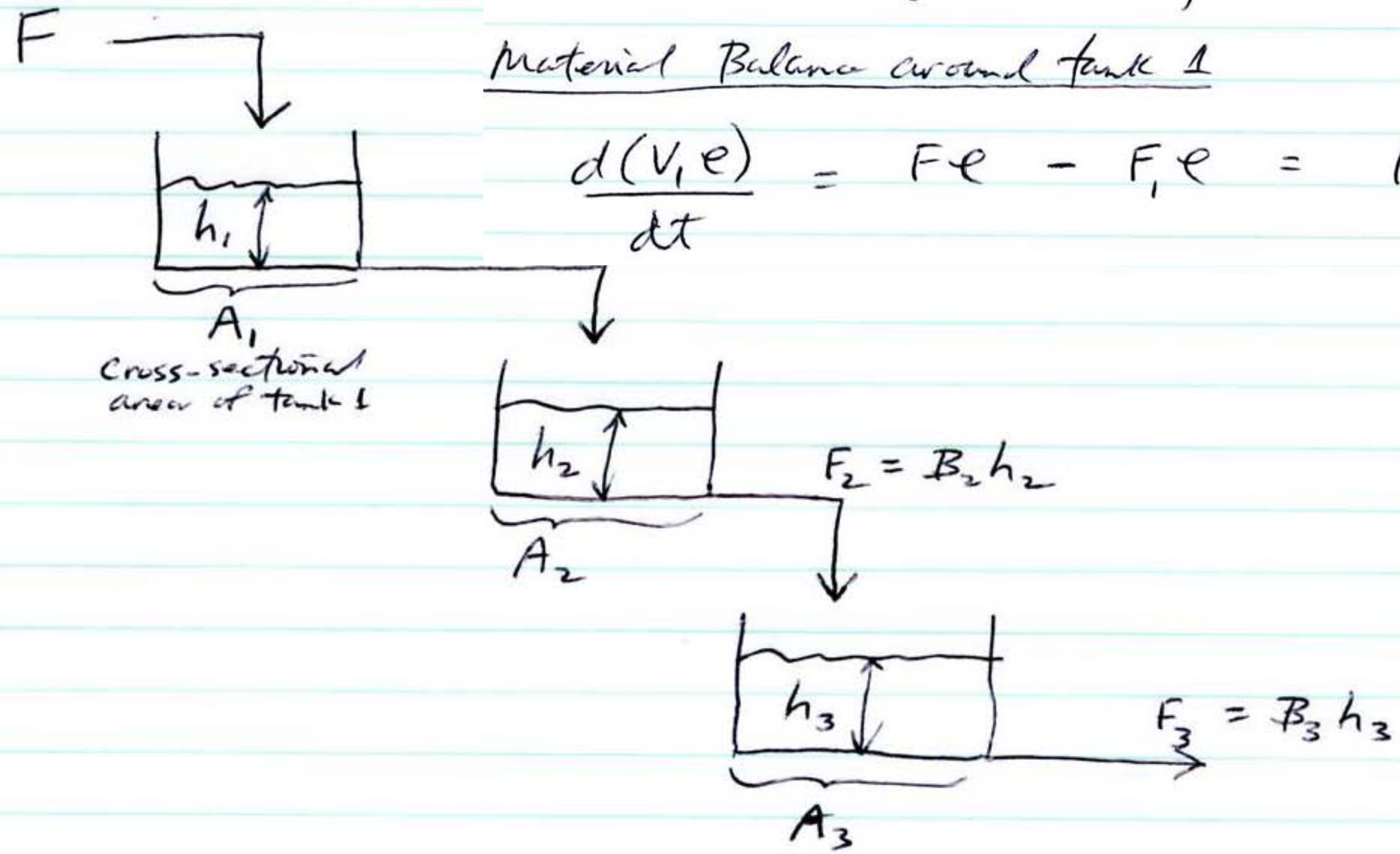
Student Workshop
3-Tank Model

- Derive state space model
- MATLAB Commands
  - Eigenvalues (eig)
  - State Space model (ss), step input (step)
- Integration using ode45: two methods
  - Single integration
  - In a “loop,” integrating from time step to time step
- Convert state space to transfer function (ss2tf)

Three tank example

Consider 3 tanks in series, where the flowrate out of each tank is proportional to the height of liquid in the tank

Let ρ = the liquid density



Material Balance around tank 1

$$\frac{d(V_1 \rho)}{dt} = F\rho - F_1\rho = F\rho - \beta_1 \rho h_1$$

ℓ and A_1 are constant, so we write

$$A_1 \ell \frac{dh_1}{dt} = F\ell - B_1 \ell h_1$$

and we cancel the ℓ terms

$$A_1 \frac{dh_1}{dt} = F - B_1 h_1$$

which we prefer to write as

$$\frac{dh_1}{dt} = -\frac{B_1}{A_1} h_1 + \frac{F}{A_1} \qquad (1)$$

Similarly, the balances around tanks 2 & 3 can be written

$$\frac{dh_2}{dt} = \frac{B_1}{A_2} h_1 - \frac{B_2}{A_2} h_2 \qquad (2)$$

$$\frac{dh_3}{dt} = \frac{B_2}{A_3} h_2 - \frac{B_3}{A_3} h_3 \qquad (3)$$

$$\frac{dh_1}{dt} = -\frac{B_1}{A_1} h_1 + \frac{F}{A_1}$$

Similarly, the balances around tanks 2 & 3 can be written

$$\frac{dh_2}{dt} = \frac{B_1}{A_2} h_1 - \frac{B_2}{A_2} h_2 \qquad (2)$$

$$\frac{dh_3}{dt} = \frac{B_2}{A_3} h_2 - \frac{B_3}{A_3} h_3 \qquad (3)$$

And it is clear that the <u>states</u> are h_1, h_2 & h_3.

& the <u>parameters</u> are $A_1, A_2, A_3, B_1, B_2, B_3$

The manipulated input is F

In this example we will assume that the measured output is the third tank height, h_3

OK, since I like "clean" parameter values for my numerical examples, use

$$\beta_1 = \beta_2 = \beta_3 = 1 \ \frac{m^3/min}{m}$$

and $A_1 = A_2 = A_3 = 1 \ m^2$

Also, let the steady-state flowrate, $F_s = 1 \ m^3/min$

From (1), @ steady-state $h_{1s} = \frac{F_s}{\beta_1} = \frac{1 \ m^3/min}{1 \ m^3/min/m} = 1 m$

Similarly, from (2) & (3) @ s.s.

$$h_{2s} = h_{3s} = 1 \ m$$

Notice, then, that the modeling eqns, (1)-(3), are

$$\begin{bmatrix} \frac{dh_1}{dt} \\ \frac{dh_2}{dt} \\ \frac{dh_3}{dt} \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} F \qquad (4)$$

which is a linear, constant coefficient model, but the states and input are not in deviation variable form.

Let $\underline{x} = \begin{bmatrix} h_1 - h_{1s} \\ h_2 - h_{2s} \\ h_3 - h_{3s} \end{bmatrix}$ and $u = F - F_s$

and show that (since $\frac{dh_1}{dt} = \frac{d(h_1 - h_{1s})}{dt}$ etc.)

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \underbrace{\begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}}_{\underline{\underline{A}}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{\underline{\underline{B}}} u$$

and, assuming that the third tank height is the measured output

$$y = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{\underline{\underline{C}}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \underbrace{0}_{\underline{\underline{D}}} u$$

MATLAB problems & solutions

Consider the following state space model for 3 tanks in series

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

where

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \qquad D = 0$$

and the time unit is minutes.

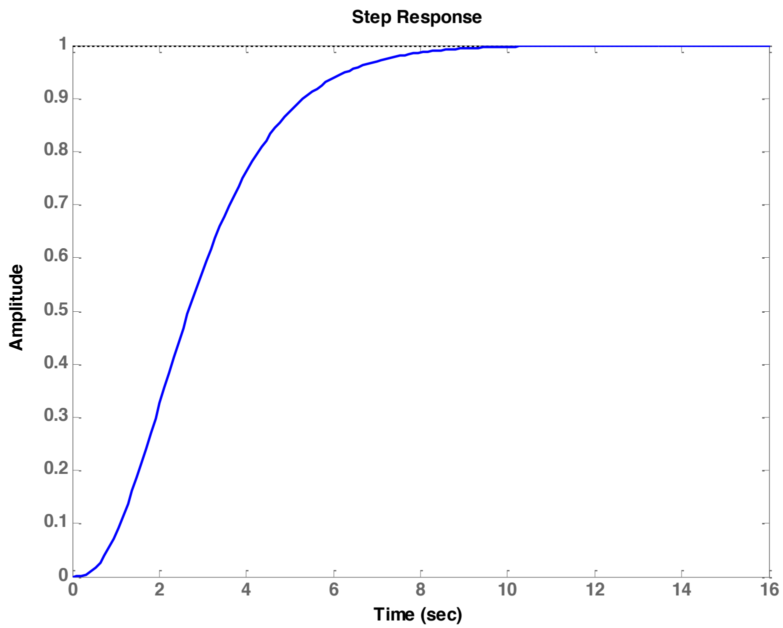
1. Find the eigenvalues of A. Using the eig function in MATLAB,

```
 a = [-1 0 0;1 -1 0;0 1 -1]
 b = [1;0;0]
 c = [0 0 1]
 d = 0
%
 eig(a)
```

we find that the eigenvalues of the continuous A matrix are -1, -1, -1 min^{-1}.

2. Form a continuous-time state space model and use the step command to generate the response to a unit step input change.

```
lintank = ss(a,b,c,d)
step(lintank)
```



Note that the plot uses seconds as the default time, even though our model is based on minutes.

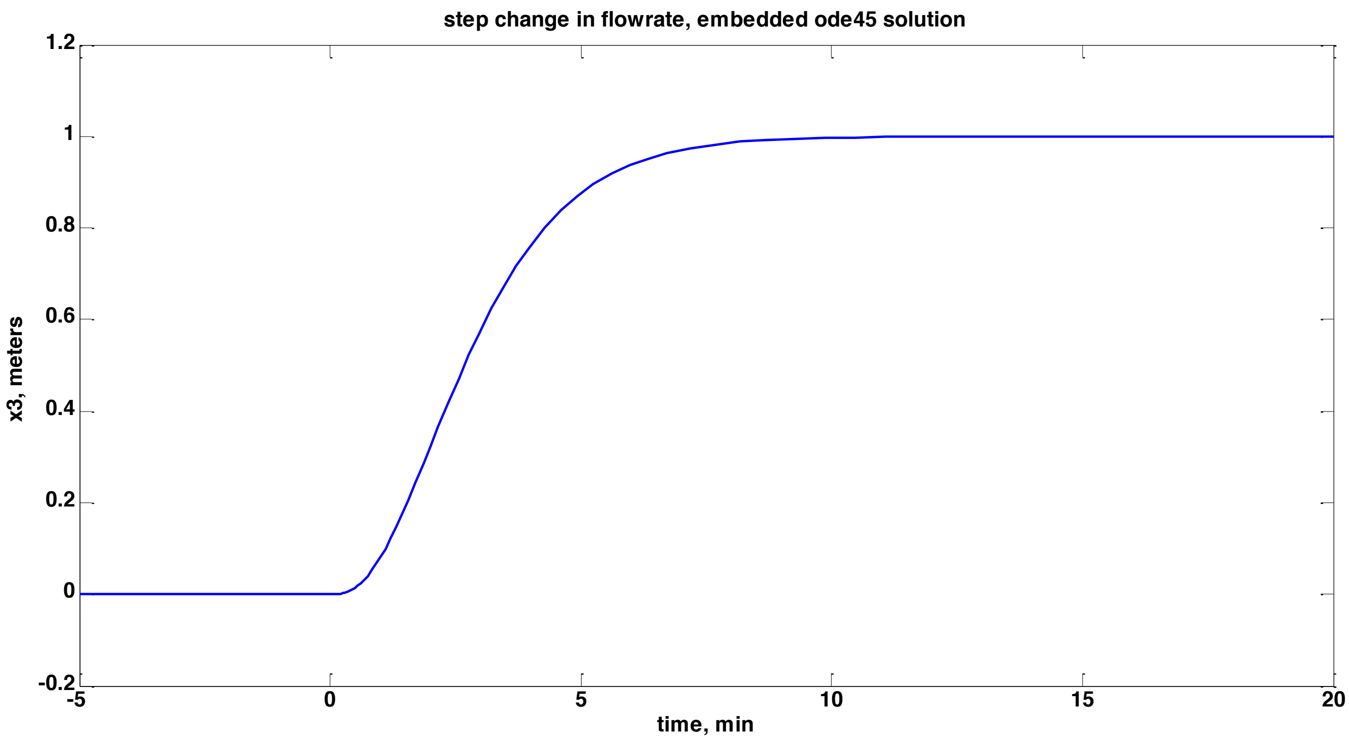
3. Write a function file to integrate the state space model using ode45. Assuming zero initial conditions and an input of u = 0 applied from time = -5 minutes to time = 0, followed by a step change in the input to u = 1 applied from time = 0 minutes to time = 20 minutes, solve for the states (and output) as a function of time. Prepare a plot and compare with the step response shown in problem 2.

```
 function xdot = examp3tnk(t,x)
%
% 3-tank example, with step change at t = 0. Model in linear, deviation variable form
%
 if t<0
    u = 0;
 else
    u = 1; % step change at t = 0
 end
   xdot(1) = -x(1) + u;
   xdot(2) = x(1) - x(2);
   xdot(3) = x(2) - x(3);
% make a row vector into a column vector
   xdot = xdot';
```

This function file is called by ode45 (see next page)

In the command window, we can enter

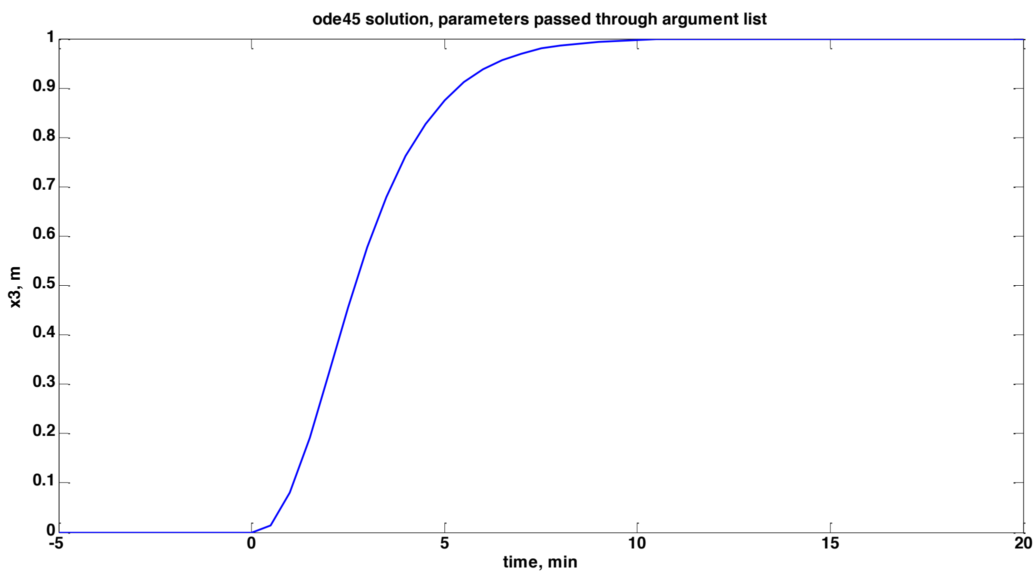
```
  tbeg = -5;        % initial simulation time, minutes
  tend = 20;        % final simulation time, minutes
  x0   = [0;0;0];   % initial conditions, column vector
%
  [t,x] = ode45('examp3tnk',[tbeg tend],x0);
%   returns a time vector and an array of states
%   the states are oriented "column-wise"
  figure(2)
  plot(t,x(:,3));   % plots the third state
  xlabel('time, min')
  ylabel('x3, meters')
  title('step change in flowrate, embedded ode45 solution')
```



4. Write a script file (m-file) with a for-loop with ode45 used to integrate the function file from time = -5 minutes to time = 20 minutes, *with a sample time of 0.5 minutes*. Again, allow the input to change from 0 to 1 at t = 0 minutes, and compare your results with problems 2 and 3.

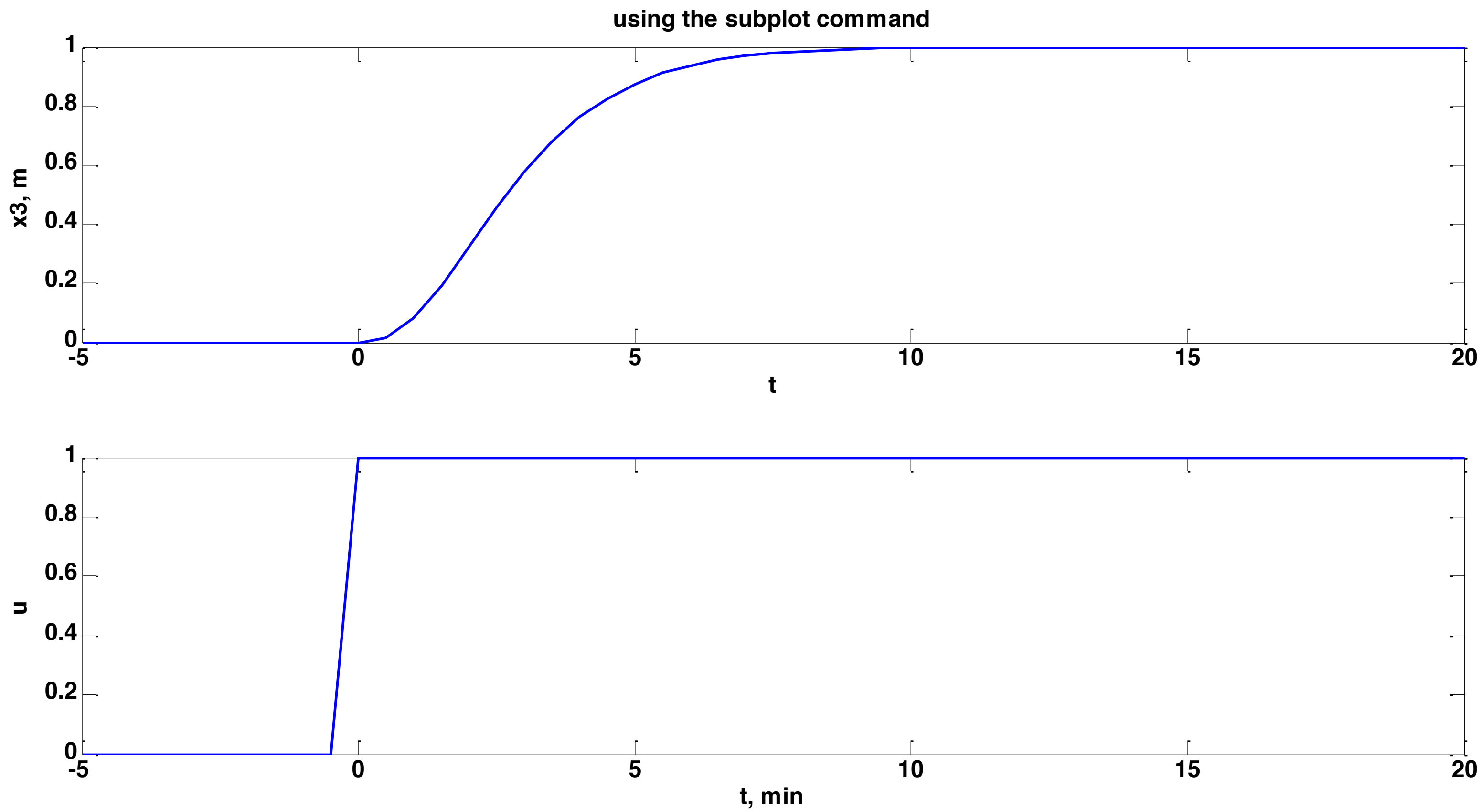
```
% same step change as problem 3, but pass the parameters and input into the function file
% set up a for loop and integrate over 0.5 minute time steps
  delt = 0.5;  % sample time = 0.5 minutes
%   start the simulation at t = -5
  tbeg = -5;
  tend = 20;
  x0   = zeros(3,1);
  time = tbeg:delt:tend;   % generate the time vector
%      for-loop to make an input step change at t = 0 minutes
  for k = 1:length(time)-1;
      if time(k) < 0;
          uvec(k) = 0;
      else
          uvec(k) = 1;
      end
  end
   xdis(:,1) = x0;  % state initial condition
  for k = 1:length(time)-1;
% generates tdummy (vector) and xdummy(array) for time points from time(k) to time(k+1)
    [tdummy,xdummy] = ode45('linodepar',[time(k) time(k+1)], xdis(:,k), [], a,b,uvec(k));
    ndum = length(tdummy);        % finds length of tdummy vector
    xdis(:,k+1) = xdummy(ndum,:)'; % xdummy is oriented with states in columns
  end
    uvec(k+1) = uvec(k);  % need length of uvec to be same as time vector
```

```
figure(5)
plot(time,xdis(3,:))
xlabel('time, min')
ylabel('x3, m')
title('ode45 solution, parameters passed through argument list')
```

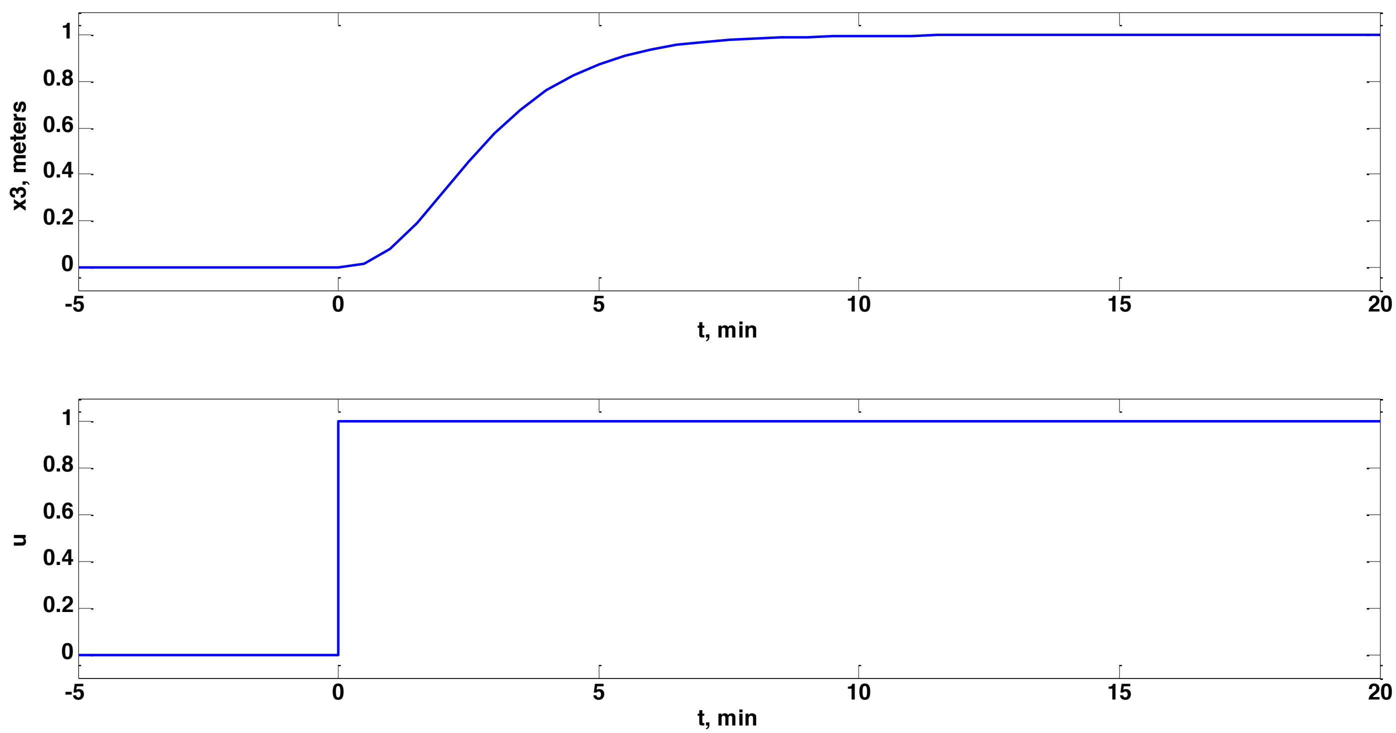


% use the subplot command to plot the third state (top) and input (bottom)

```
figure(7)
subplot(2,1,1)
plot(time,xdis(3,:))
xlabel('t')
ylabel('x3, m')
title('using the subplot command')
subplot(2,1,2)
plot(time,uvec)
xlabel('t, min')
ylabel('u')
```



```
%  notice that a discontinuous step looks better
%
  [tt,uu] = stairs(time,uvec);  % uses zero-order hold
%
  figure(8)
  subplot(2,1,1)
  plot(time,xdis(3,:))
  axis([-5 20 -0.1 1.1])    % axis limits for the state 3 plot
  xlabel('t, min')
  ylabel('x3, meters')
  subplot(2,1,2)
  plot(tt,uu)
  axis([-5 20 -0.1 1.1])    % axis limits for the input plot
  xlabel('t, min')
  ylabel('u')
```



5. Use the MATLAB ss2tf command to convert the state space model to a transfer function model. Find the poles of the transfer function. How do the poles of the transfer function compare with the eigenvalues of A?

```
 [num,den] = ss2tf(a,b,c,d)

num =
      0    0.0000    0.0000    1.0000

den =
    1     3     3     1

 roots(den)      % find the poles (inverse minutes)

ans =
 -1.0000 + 0.0000i
 -1.0000 - 0.0000i
 -1.0000
```

Answer: the poles of the transfer function are the same as the eigenvalues of the A matrix