RBT2_HW3
Aly Khater

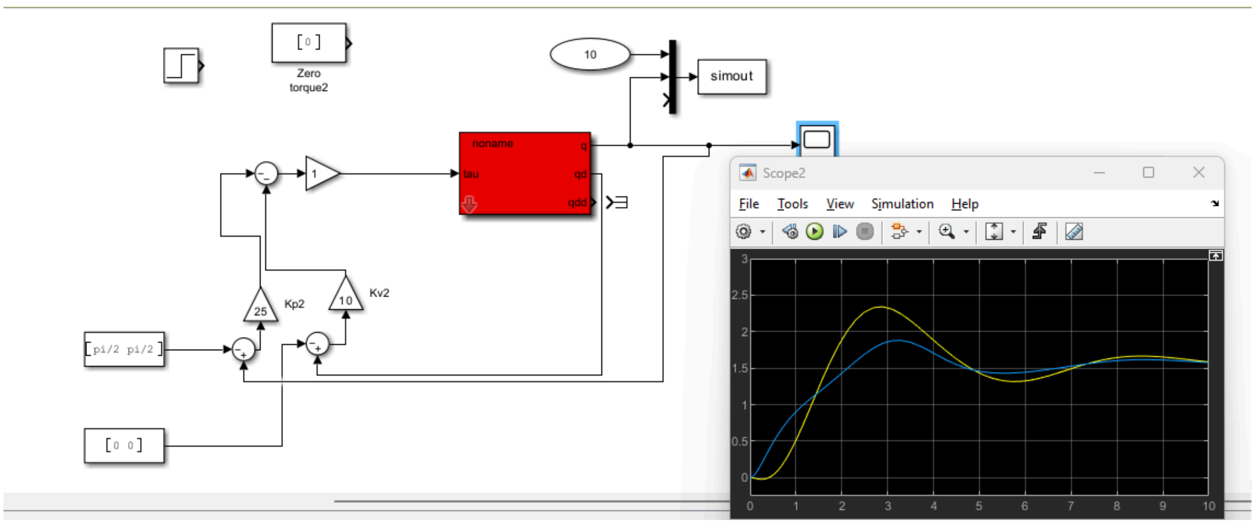
Problem 2

SCURRTWOLINK.m

```
%clear all

L{1} = link([0 1 0 0 0], 'standard');
L{2} = link([0 1 0 0 0], 'standard');
L{1}.m = 3;
L{2}.m = 3;
L{1}.r = [0 0 0];
L{2}.r = [0 0 0];
L{1}.I = [0 0 3 0 0 0];
L{2}.I = [0 0 3 0 0 0];
L{1}.Jm = 0;
L{2}.Jm = 0;
L{1}.G = 1;
L{2}.G = 1;
L{1}.B = 3;
L{2}.B = 3;

%useful poses
qz=[0 0];   %zero angle

SCURRtwolink = robot(L);
r = robot(L);
```

Problem 2 part i

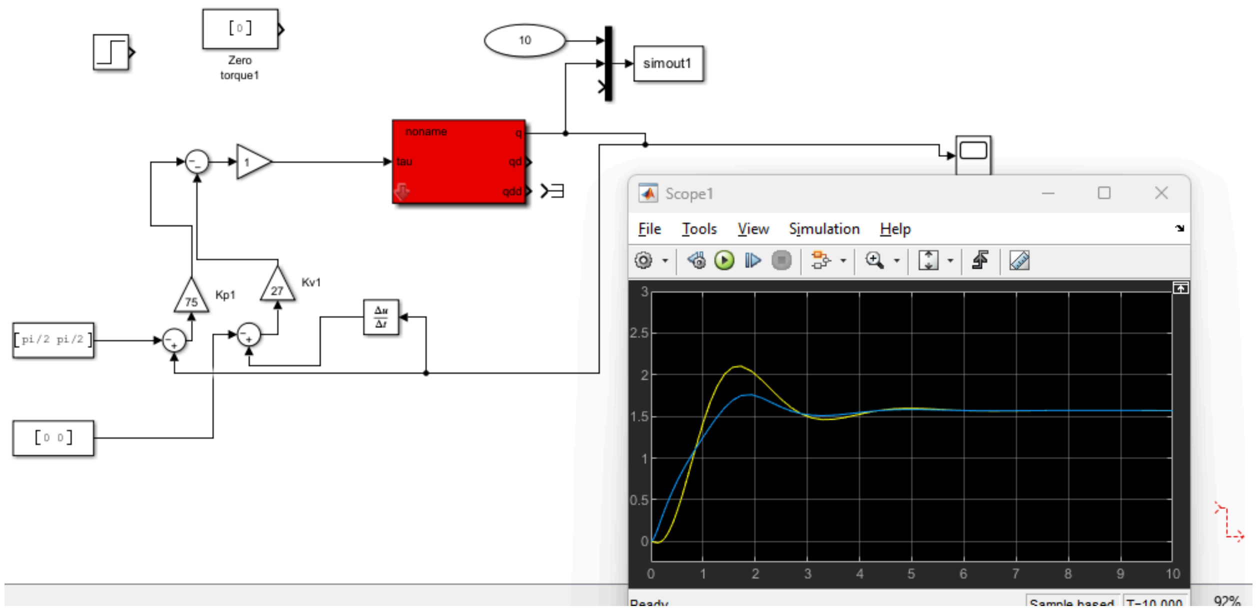
We will not get ideal results. We are assuming linearity.



I = 3
B = 3

Kp = 25
Kv = 10

Problem 2 part 3



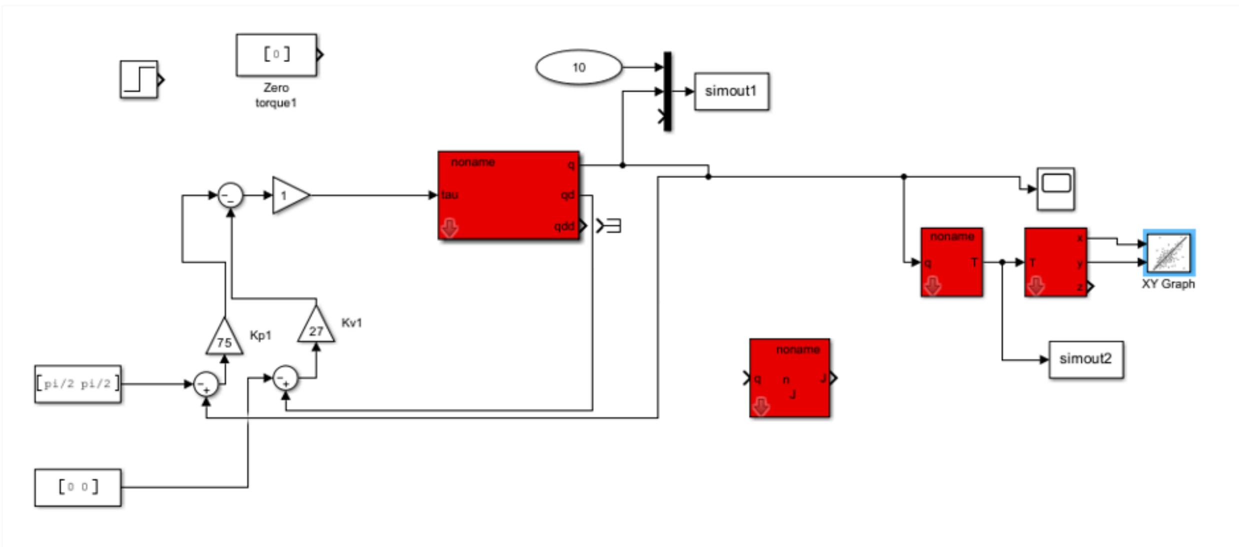
I = 3
B = 3
Kp = 75
Kv = 27

Worse, mainly due to linearity.

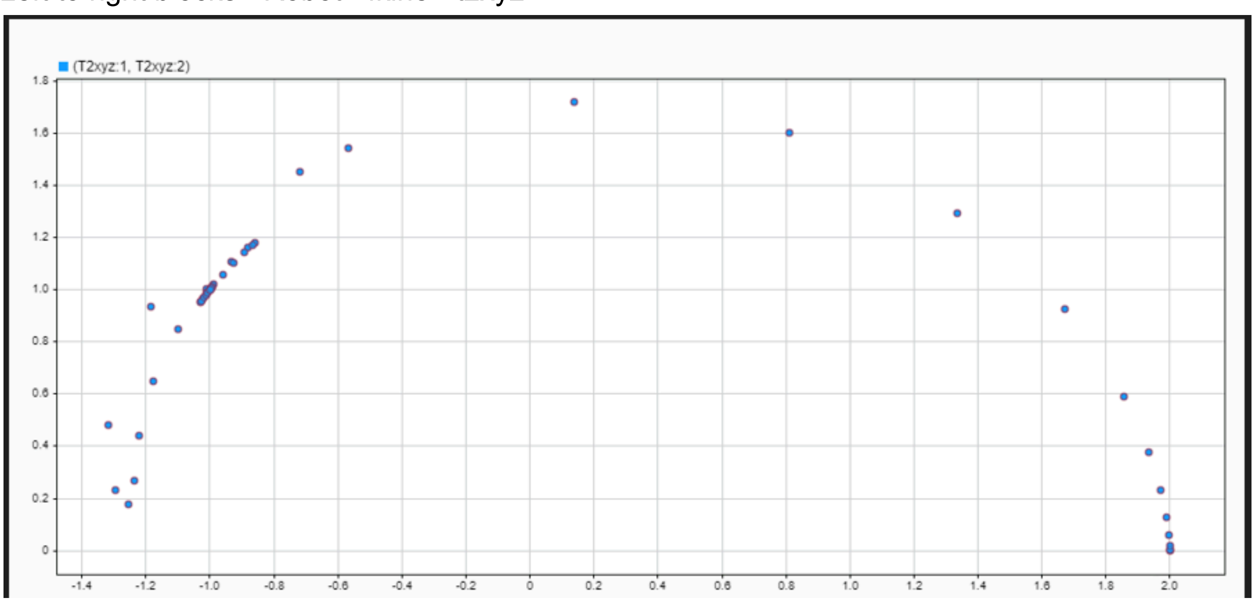
Problem 3

Problem 3a

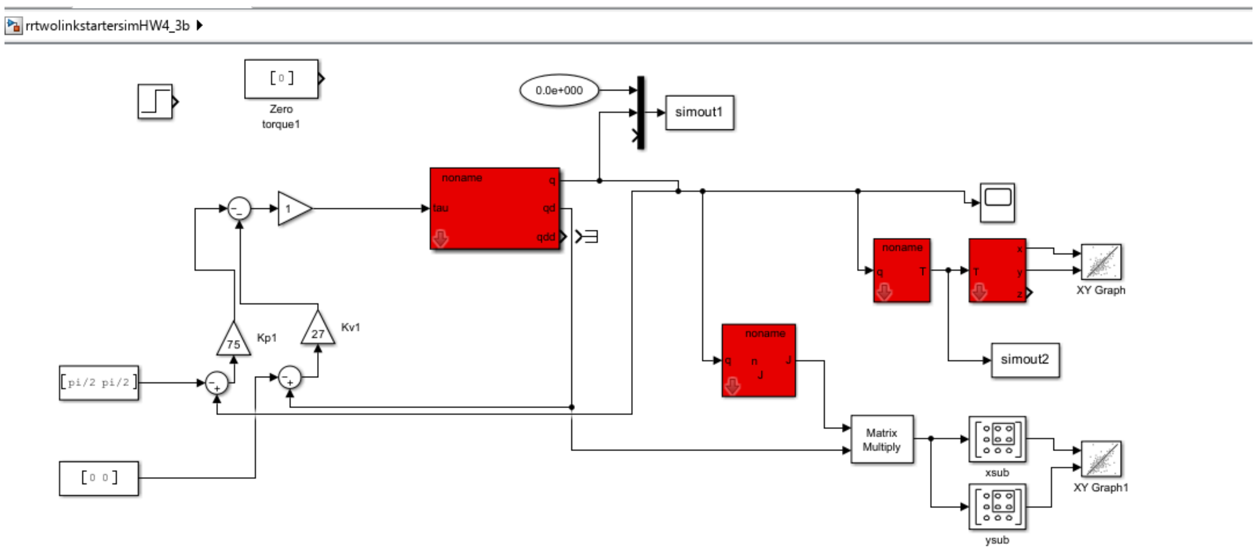
ignore J block in the center



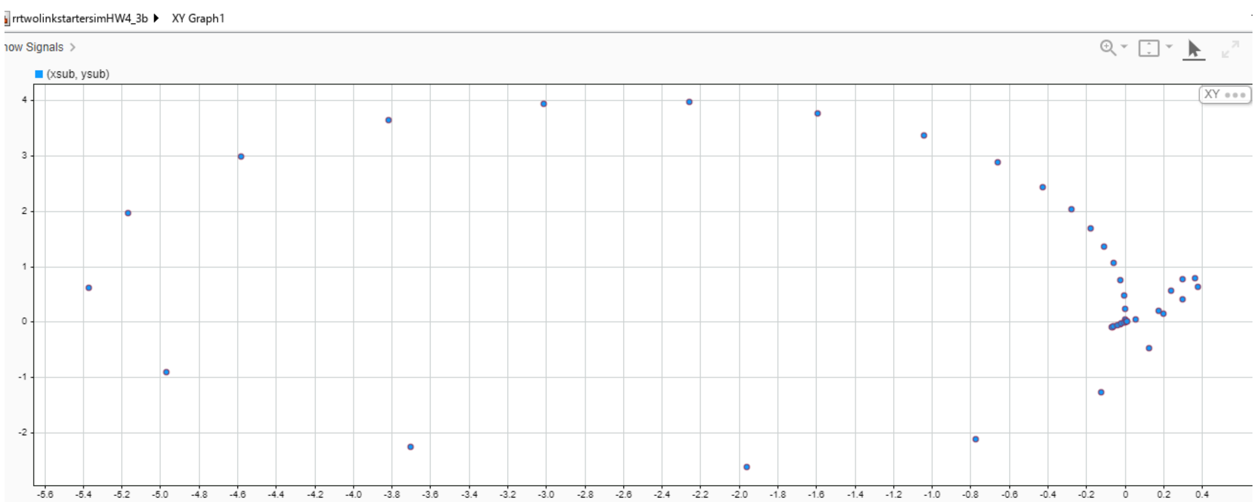
Left to right blocks - Robot - fkine - t2xyz



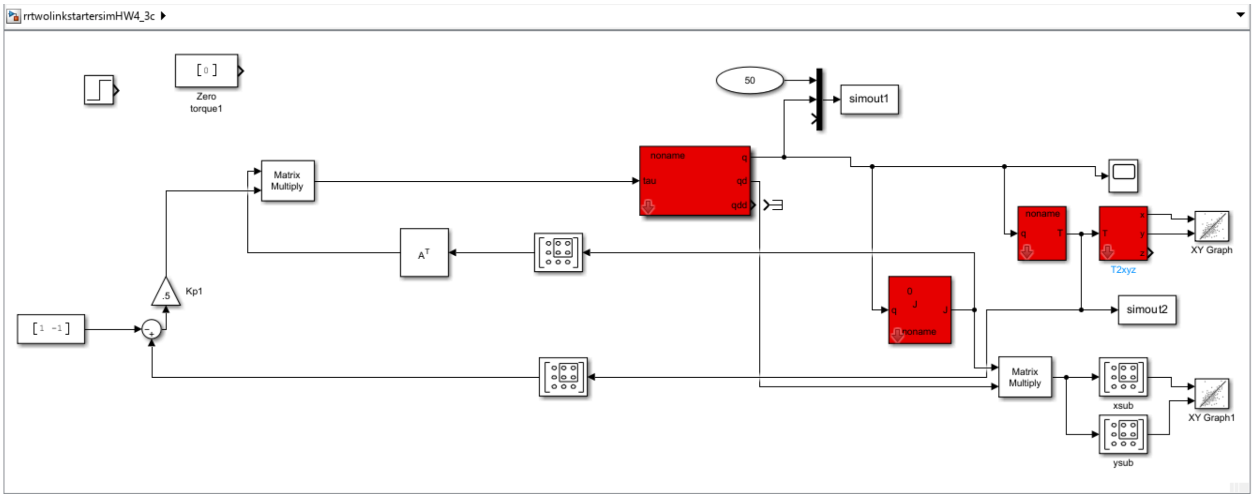
Problem 3b



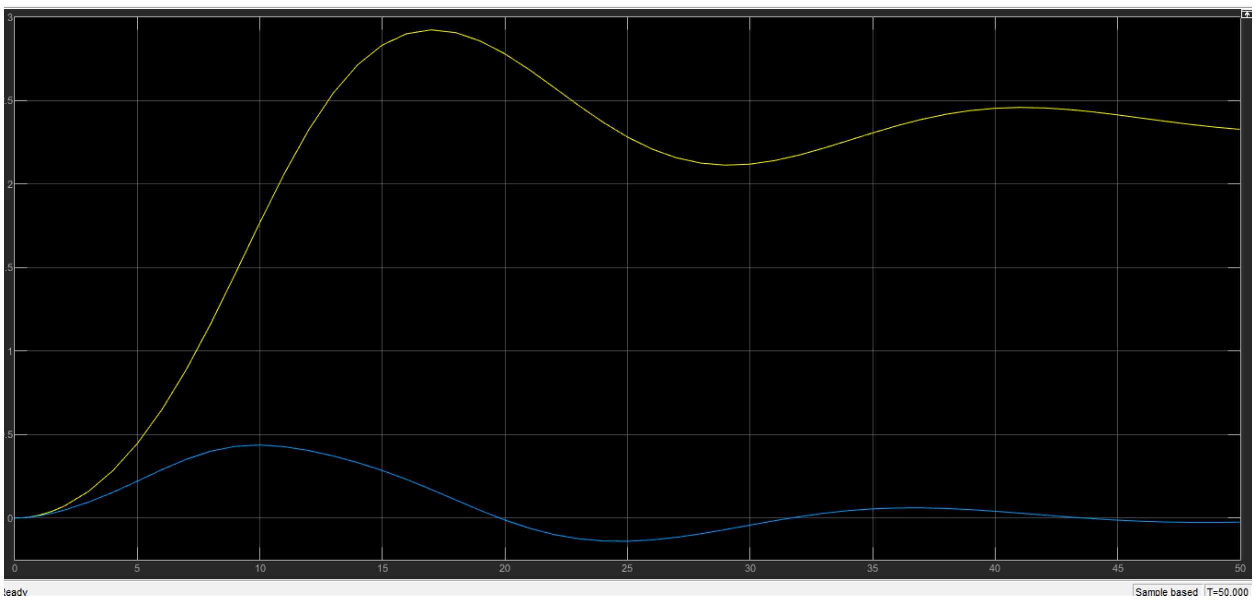
Left to right blocks - Robot - jacob0 - fkine - t2xyz



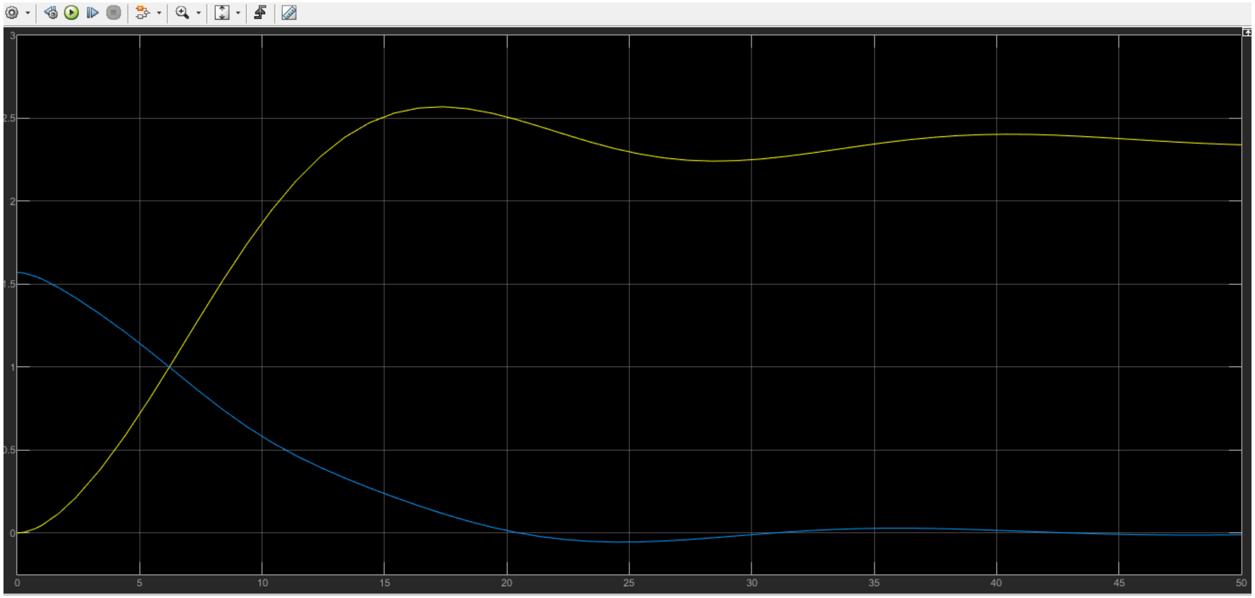
Problem 3c

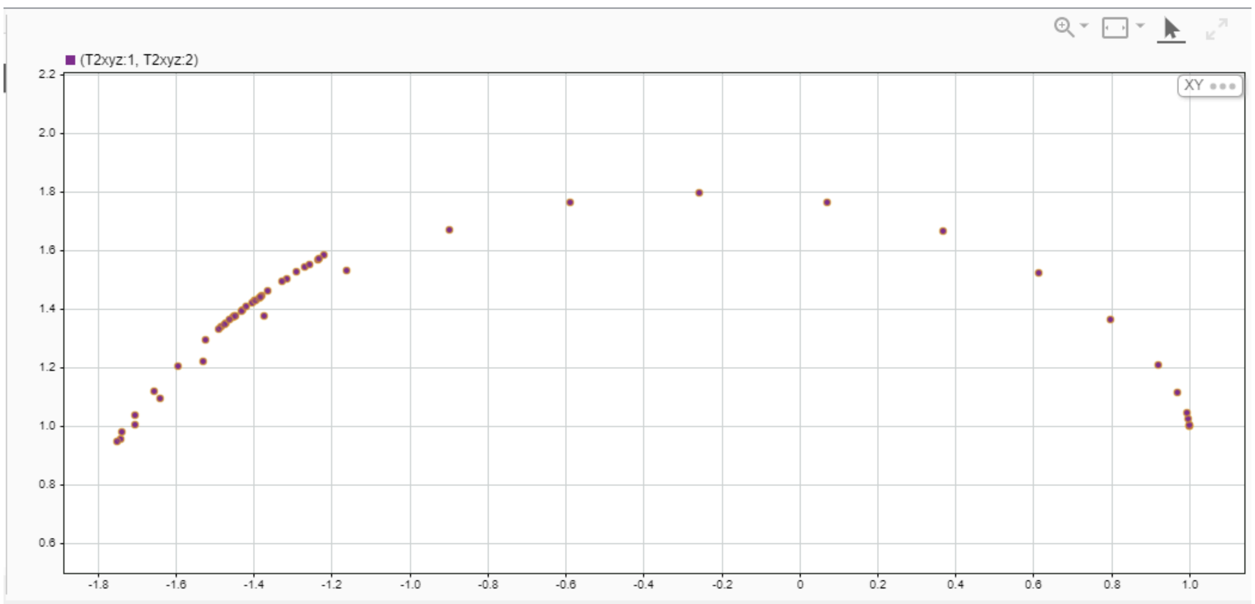


Left to right blocks - Robot - jacob0 - fkine - t2xyz



Joint position [0 pi/2]


0
5
10
15
20
25
30
35
40
45
50


(T2xyz:1, T2xyz:2)
XY
2.2
2.0
1.8
1.6
1.4
1.2
1.0
0.8
0.6
-1.8
-1.6
-1.4
-1.2
-1.0
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1.0

Problem 4 and 5

④ $$\tau = \overset{M}{(2\sqrt{\theta}+1)}\,\ddot{\theta} + \overset{C}{3\dot{\theta}^2} - \overset{G}{\sin\theta}$$

$$\alpha = 2\sqrt{\theta}+1$$
$$\beta = 3\dot{\theta}^2 - \sin\theta$$

$k = 10$ $\qquad \ddot{\theta} + 2\zeta w_n \dot{\theta} + w_n^2\theta$

$w_n^2 = 10$

$w_n = \sqrt{10}$

$k_v = 2\sqrt{10}$

$k_p = 10$

⑤ $$\tau_1 = m_1 l_1^2 \ddot{\theta}_1 + m_1 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2$$
$$\tau_2 = m_2 l_2^2 (\ddot{\theta}_1 + \ddot{\theta}_2) + v_2 \dot{\theta}_2$$

$$\alpha = \begin{bmatrix} m_1 l_1^2 & 0 \\ m_2 l_2^2 & m_2 l_2^2 \end{bmatrix}$$

$$\beta = \begin{pmatrix} m_1 l_1 l_2 \dot{\theta}_1 \dot{\theta}_2 \\ v_2 \dot{\theta}_2 \end{pmatrix}$$

$$k_p = \begin{pmatrix} k_{p1} & 0 \\ 0 & k_{p2} \end{pmatrix} = \begin{pmatrix} w_{n1}^2 & 0 \\ 0 & w_{n2}^2 \end{pmatrix}$$

$$k_v = \begin{pmatrix} 2\sqrt{k_{p1}} & 0 \\ 0 & 2\sqrt{k_{p2}} \end{pmatrix} = \begin{pmatrix} 2\zeta w_n & 0 \\ 0 & 2\zeta w_n \end{pmatrix}$$

They relate to critical damping constant since $k_v = 2\sqrt{k_p}$

Yes, it can be a serial manipulator

Problem 6

Linearization and decoupling

Problem 7

Dynamic parameters can be updated at a rate slower than the rate of the closed-loop servo. Functions can be computed by a background process. Parallel processing.

Problem 8

$$M(\Theta)=\begin{bmatrix} l_2^2 m_2 + 2 l_1 l_2 m_2 c_2 + l_1^2 (m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 c_2 \\ l_2^2 m_2 + l_1 l_2 m_2 c_2 & l_2^2 m_2 \end{bmatrix}.$$

(6.60)

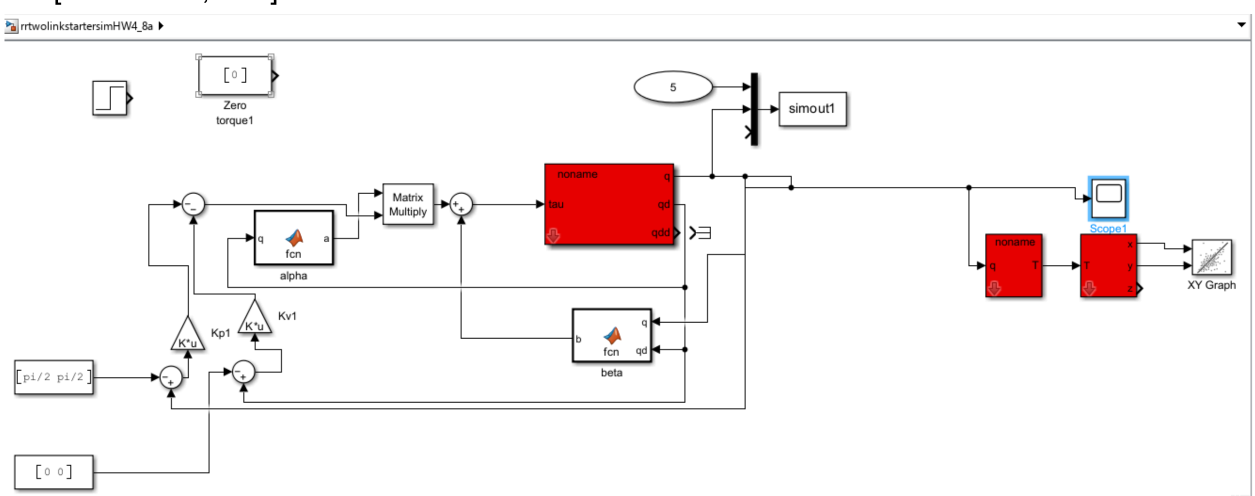
Any manipulator mass matrix is symmetric and positive definite, and is, therefore, always invertible.

The velocity term, $V(\Theta, \dot{\Theta})$, contains all those terms that have any dependence on joint velocity. Thus, we obtain

$$V(\Theta, \dot{\Theta})=\begin{bmatrix} -m_2 l_1 l_2 s_2 \dot{\theta}_2^2 - 2 m_2 l_1 l_2 s_2 \dot{\theta}_1 \dot{\theta}_2 \\ m_2 l_1 l_2 s_2 \dot{\theta}_1^2 \end{bmatrix}.$$

(6.61)

M = [3+6c2+6 , 3+3c2 ; 3+3c2, 3]
V = [-3S2 - 6S2, 3S2]

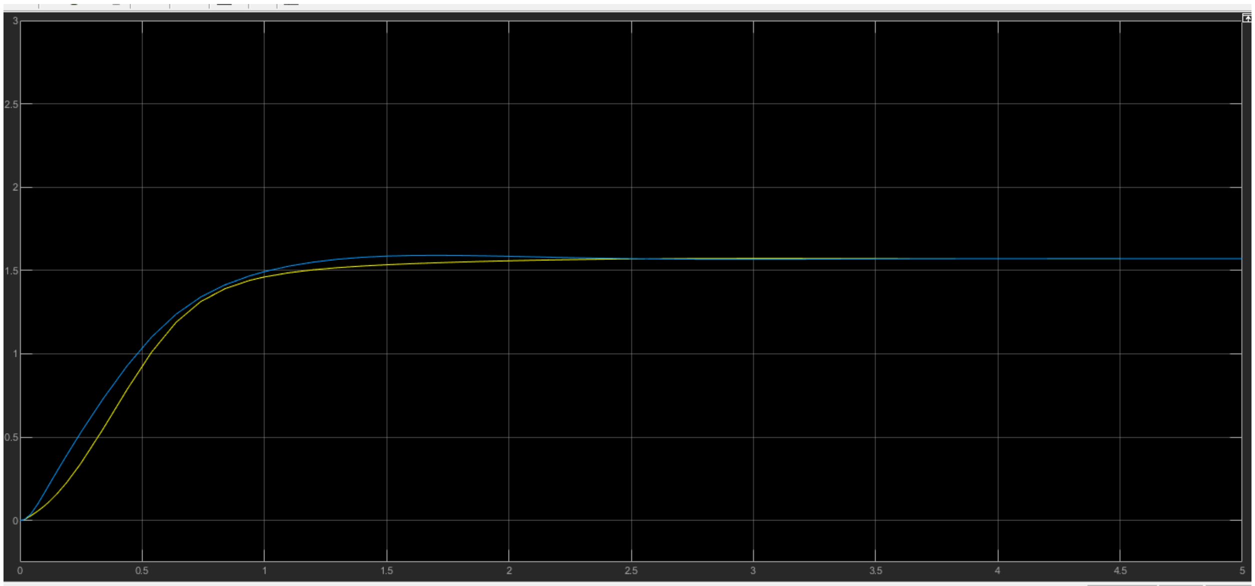


Left to right blocks - Robot - fkine - t2xyz rrtwolinkstartersimHW4_8a ▸ alpha

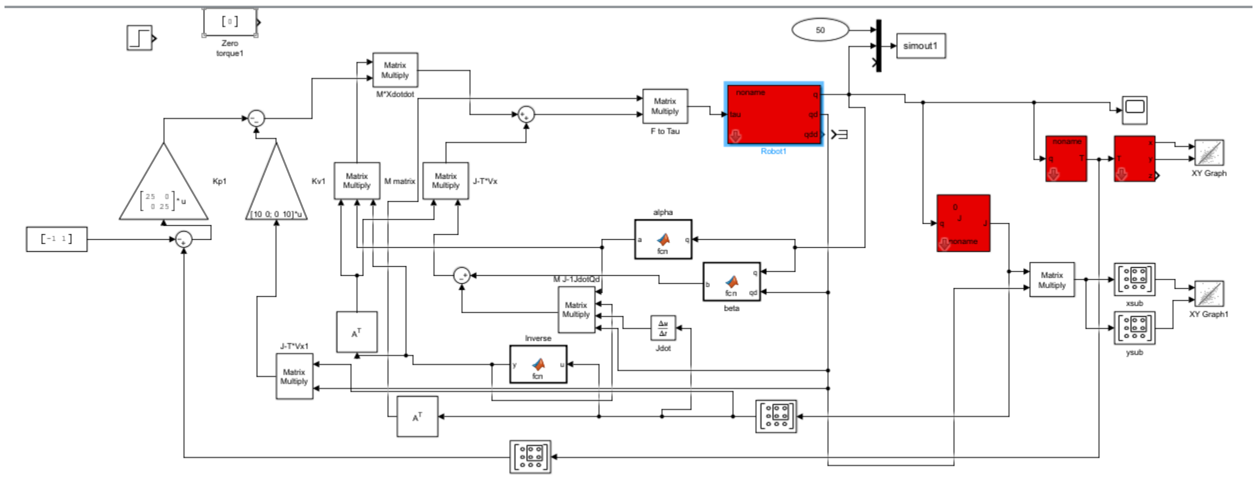
```
function a = fcn(q)
l1 = 1;
l2 = 1;
m1 = 3;
m2 = 3;
c2 = cos(q(2));

%y = [3+6*cos(q(2))+6, 3+3*cos(q(2)); 3+3*cos(q(2)), 3]*e;
a = [l2^2*m2+2*l1*l2*m2*c2+l1^2*(m1+m2), l2^2*m2+l1*l2*m2*c2;l2^2*m2+l1*l2*m2*c2,l2^2*m2];
end
```

rrtwolinkstartersimHW4_8a ▸ beta

```
function b = fcn(q,qd)
l1 = 1;
l2 = 1;
m2 = 3;
s2 = sin(q(2));
t1 = qd(1);
t2 = qd(2);

%y = [-3*(sin(q(2)) * qd(2)^2+2*3*sin(q(2))*qd(1)*qd(2)); 3*sin(q(2))*qd(1)^2];
a = [-m2*l1*l2*s2*t2^2-2*m2*l1*l2*s2*t1*t2;m2*l1*l2*s2*t1^2];
b = a + [3*t1;3*t2]
end
```



Problem 9a

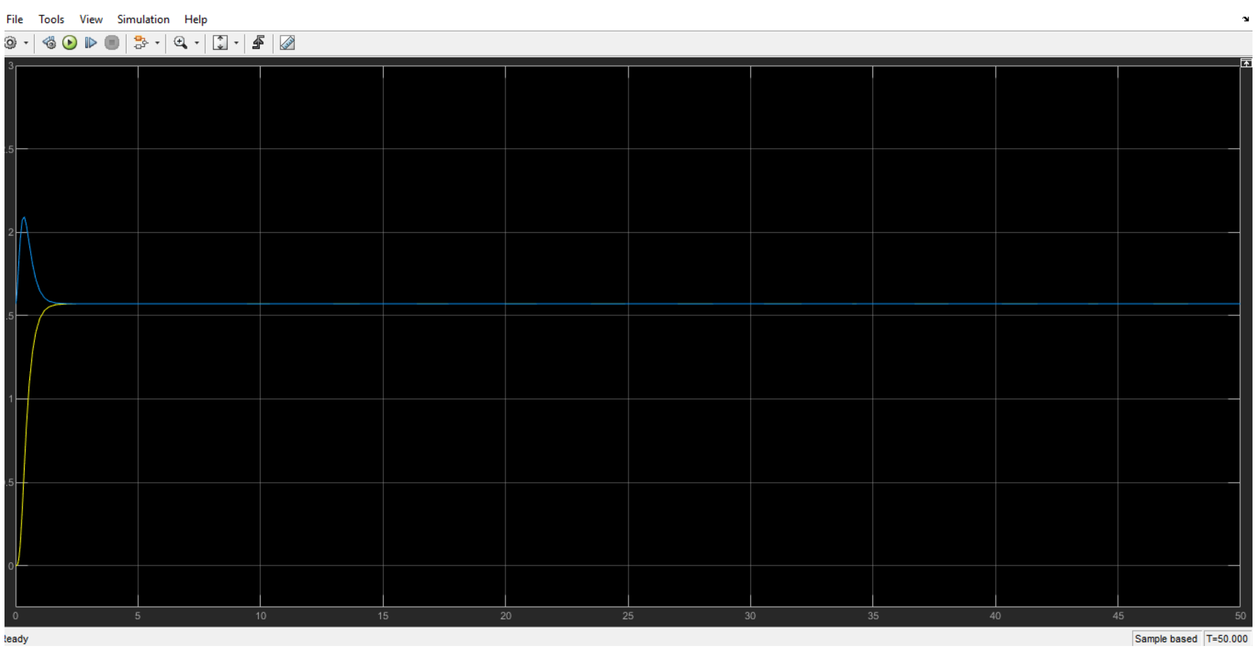


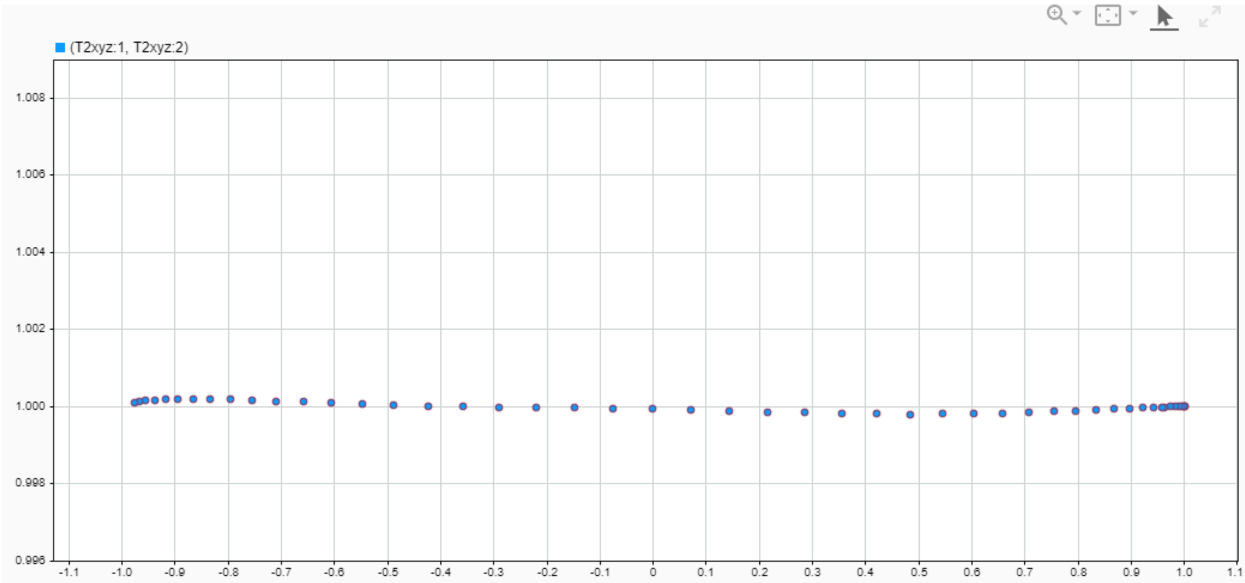
Left to right blocks - Robot - jacob0 - fkine - t2xyz

$$M_x(\Theta) = J^{-T}(\Theta)M(\Theta)J^{-1}(\Theta),$$

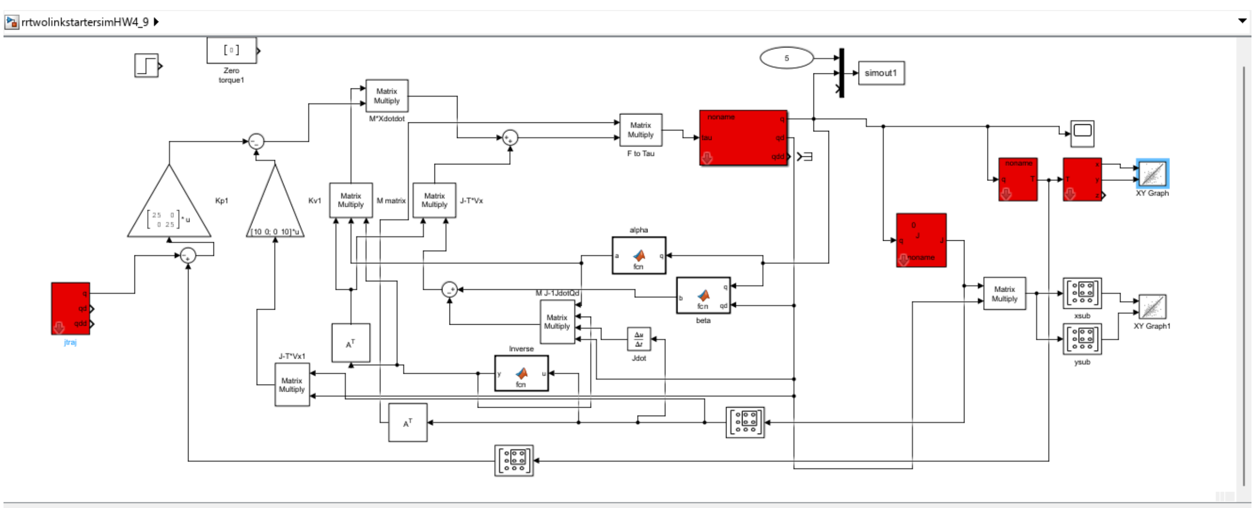
$$V_x(\Theta,\dot{\Theta}) = J^{-T}(\Theta)\left(V(\Theta,\dot{\Theta}) - M(\Theta)J^{-1}(\Theta)\dot{J}(\Theta)\dot{\Theta}\right),$$

$$G_x(\Theta) = J^{-T}(\Theta)G(\Theta).$$





Jtraj



Left to right blocks - jtraj - Robot - jacob0 - fkine - t2xyz twolinkstartersimHW4_9 ▶ Inverse

```
function y = fcn(u)

y = inv(u);
end
```