LAB 3

Model and control of a half quadrotor system

Symbol	Description	Value / Unit
Variables		
V_m	Controlled input voltage	V
E_b	Back electromotive force (emf)	V
I_m	Motor current	A
Ω_m	Motor and propeller rotation speed	$rad.s^{-1}$
T_d	Resistant torque	$N.m$
T_m	Applied torque from the DC motor	$N.m$
DC motor constants		
R_m	Terminal resistance	$8.4\ \Omega$
k_t	Torque constant	$0.042\ N.m.A^{-1}$
k_m	Motor back-emf constant	$0.042\ N.m.A^{-1}$
J_m	Rotor inertia	$4.0 \times 10^{-6}\ kg.m^2$
L_m	Rotor inductance	$1.16\ mH$
Propeller constants		
k_1, k_2	Drag / air resistance coefficient	See Table 2
J_h	Propeller hub inertia	$3.04 \times 10^{-9} kg.m^2$
J_p	Propeller inertia	$7.2 \times 10^{-6} kg.m^2$

Question 1: model of the propulsive device.

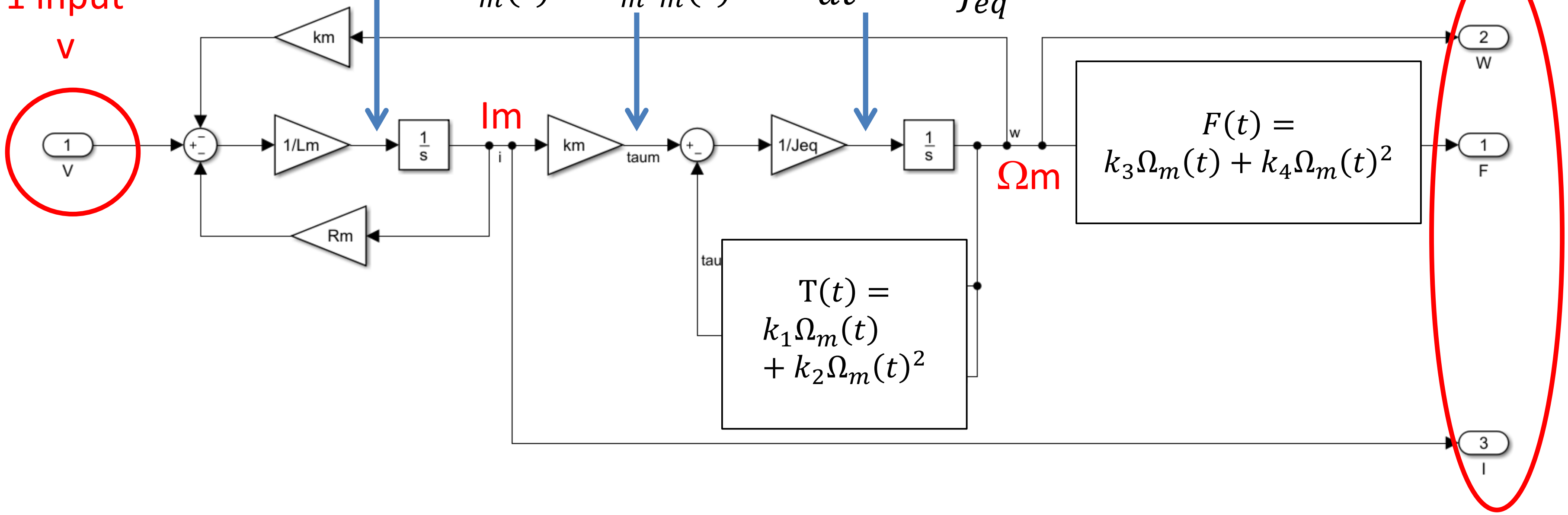
$$\frac{dI_m(t)}{dt} = \frac{1}{L_m}\left(V_m(t) - R_m I_m(t) - k_m \Omega_m(t)\right)$$

$$T_m(t) = k_m I_m(t)$$

$$\frac{d\Omega_m(t)}{dt} = \frac{1}{J_{eq}}\left(T_m(t) - T(t)\right)$$

1 input
v

3 outputs
F, w, I



Electric_sol.slx

Question 2: find the equilibrium point.

$$L_m \frac{dI_m(t)}{dt} = V_m(t) - R_m I_m(t) - k_m \Omega_m(t)$$

$$J_{eq} \frac{d\Omega_m(t)}{dt} = k_m I_m(t) - \mathrm{T(t)}$$

$$\mathrm{T}(t) = k_1 \Omega_m(t) + k_2 \Omega_m(t)^2$$

small variations

$$\begin{cases} V_m(t) &= & V + v(t) \\ I_m(t) &= & I + i(t) \\ \Omega_m(t) &= & \Omega + \omega(t) \\ \mathrm{F}_m(t) &= & \mathrm{F} + f(t) \end{cases}$$

V = V, I = I_o?, Ω=w_0?

Linearization at first order $\Omega_m(t)^2$=$(w_0 + \omega)^2$= $w_0{}^2 + 2\mathrm{w}w_0 + \cancel{w^2} \approx w_0{}^2 + 2\mathrm{w}w_0$

Equilibrium point $\Rightarrow \begin{cases} V_0 - R_m I_o - k_m w_0 = 0 \\ k_m I_0 - T_0 = k_1 w_0 + k_2 w_0^2 - k_m I_0 = 0 \end{cases}$

w_0=224
I_0 = 0.69

Question 3: Linearization.

$$\frac{di(t)}{dt} = \frac{1}{L_m}\left(v_m(t) - R_m i(t) - k_m \omega(t)\right)$$

$$\frac{d\omega(t)}{dt} = \frac{1}{J_{eq}}\left(k_m i(t) - T(t)\right) = \frac{1}{J_{eq}}\left(k_m i(t) - (k_1 + 2k_2 w_0)\omega(t)\right)$$

$$\mathrm{T}(t) = k_1\omega(t) + 2k_2 w_0 \omega(t)$$

$$F(t) = k_3\omega(t) + 2k_4 w_0 \omega(t)$$

$$\frac{\omega}{v} = \frac{k_m}{J_{eq} L_m s^2 + \left[R_m J_{eq} + k_1 L_m + 2 k_2 \omega_0 L_m\right] s + k_m^2 + R_m k_1 + 2 k_2 \omega_0 R_m}$$

$$\frac{F}{\omega} = k_3 + 2 k_4 \cdot \omega_0$$

Transfer function

```
v0 = 10;
w0 = 224;

num  = (k3+2*k4*w0)*km
den = [Jeq*Lm    (Rm*Jeq+k1*Lm+2*k2*w0*Lm)   (km*km+Rm*k1+2*k2*w0*Rm)]
Gtflin = tf(num,den)
roots(den)
figure(1); bode(Gtflin); grid on;hold on;
```

State-space representation

```
A = [(-k1-2*k2*w0)/Jeq km/Jeq   ;  -km/Lm  -Rm/Lm];
B = [0 ; 1/Lm];
C = [k3+2*k4*w0 0];
D = 0;
GTsslin = ss(A,B,C,D);
bode(GTsslin)
eig(A)
```

Question 3 bis: create and convert Matlab Linear Systems Models

Equilibrium point

```
v0 = 10;
w0 = 224;
```

Transfer function

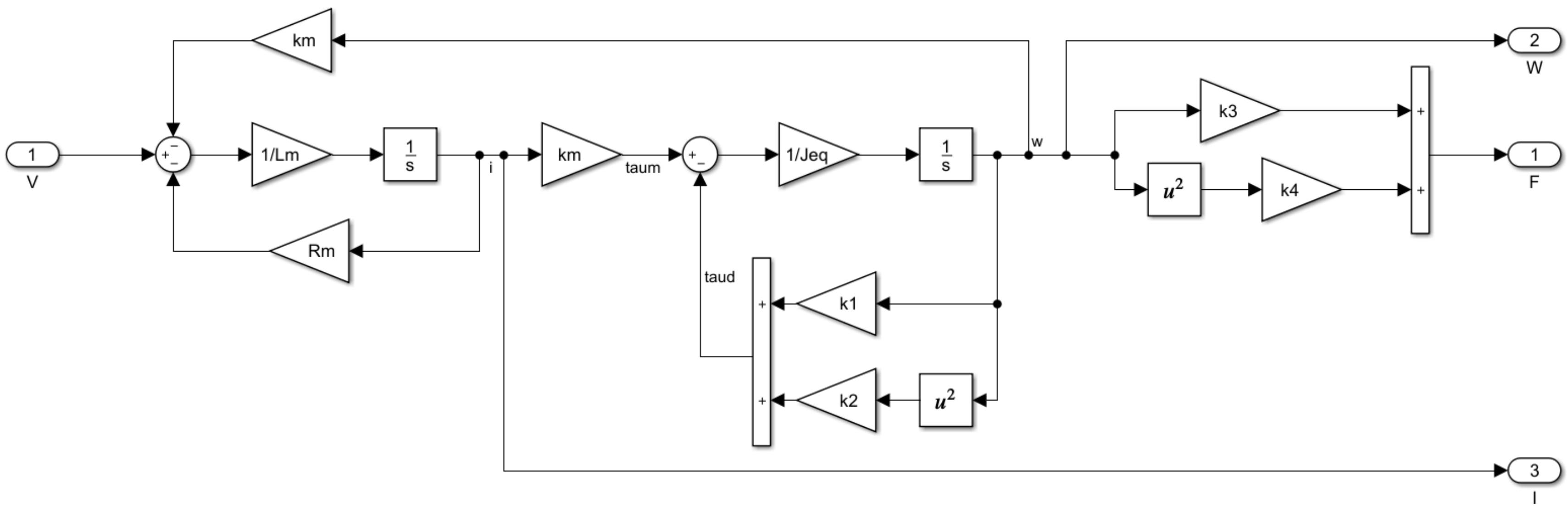
```
num  = (k3+2*k4*w0)*km;
den = [Jeq*Lm (Rm*Jeq+k1*Lm+2*k2*w0*Lm) (km*km+Rm*k1+2*k2*w0*Rm)];
Gtflin = tf(num,den)
roots(den)
figure(1); bode(Gtflin); grid on;hold on;
```

State-space representation

```
A = [(-k1-2*k2*w0)/Jeq km/Jeq   ;  -km/Lm  -Rm/Lm];
B = [0 ; 1/Lm];
C = [k3+2*k4*w0 0];
D = 0;
GTsslin = ss(A,B,C,D);
bode(GTsslin)
eig(A)
```

Question 4: create the Simulink nonlinear model

Electric_sol.slx



Question 5: find the equilibrium point

Set initial values (or not: in that case use [ ])

```
%x0=[];u0=[];y0=[];
x0=[];u0=10;y0=[]; %Initial input at 10
```

Select the index of states and/or states which are to be freezed:

```
ix=[];
iu=1; %The input n°1 is set constant initial value was set at 10
iy=[];
```

Find the equilibrium point:

```
[xtrim,utrim,ytrim,dxtrim] = trim('Electrical_sol',x0,u0,y0,ix,iu,iy)
```

```
utrim =

     10
```

```
xtrim =

  224.4221
    0.0684
```

Question 6: find the linearized model

Find the linearized model at the equilibrium point. (function linmod)

```
[A,B,C,D] = linmod('Electrical_sol', xtrim, utrim)
```

Create a state space model and a transfer function model

```
Gss=ss(A,B,C,D);
Gss=Gss(1,1);
Gtf=tf(Gss)
damp(Gtf)
```

Transfer function between output 1 and input 1 => F/v

$$\frac{F}{V} = \frac{903.7}{s^2 + 7243\,s + 1.445e05}$$

Question 7: Analyze the system properties

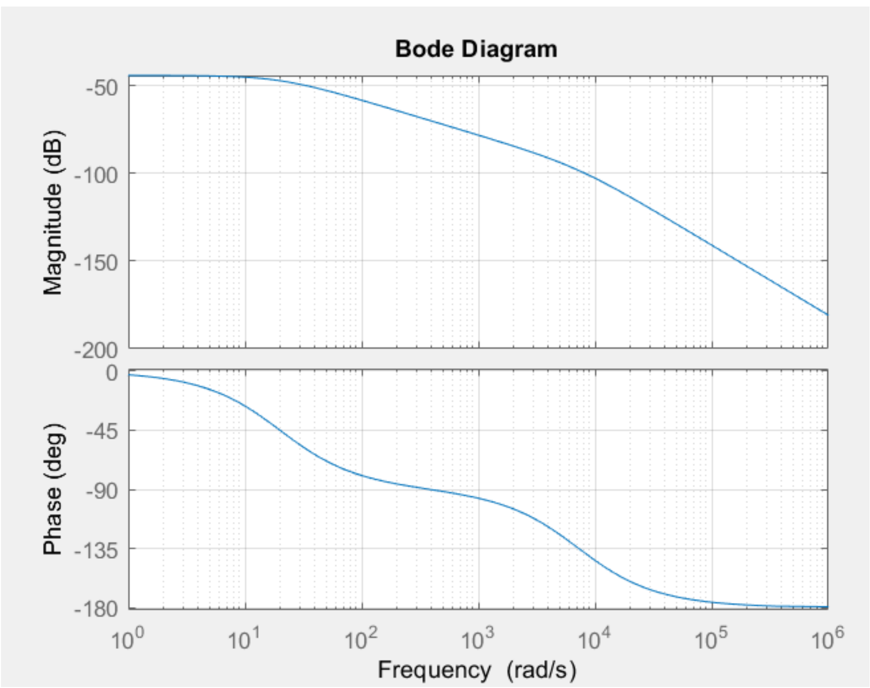
Slow dynamic
Dominant mode

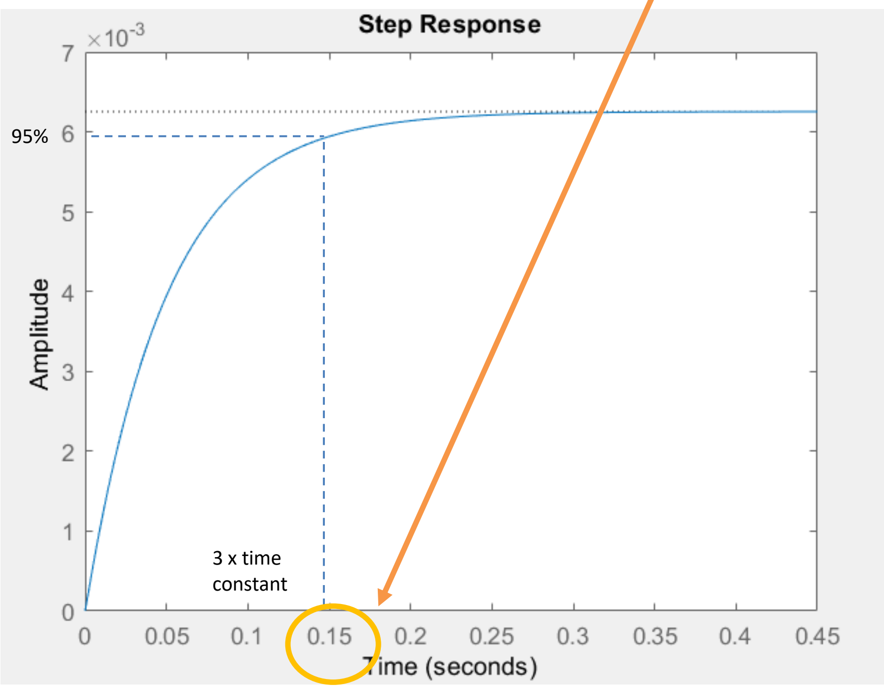
Second order transfer function

2 single poles with Re<0 => Stable system

2 different dynamics: 1 slow & 1 quick

Pole	Damping	Frequency (rad/seconds)	Time Constant (seconds)
-2.00e+01	1.00e+00	2.00e+01	5.00e-02
-7.22e+03	1.00e+00	7.22e+03	1.38e-04

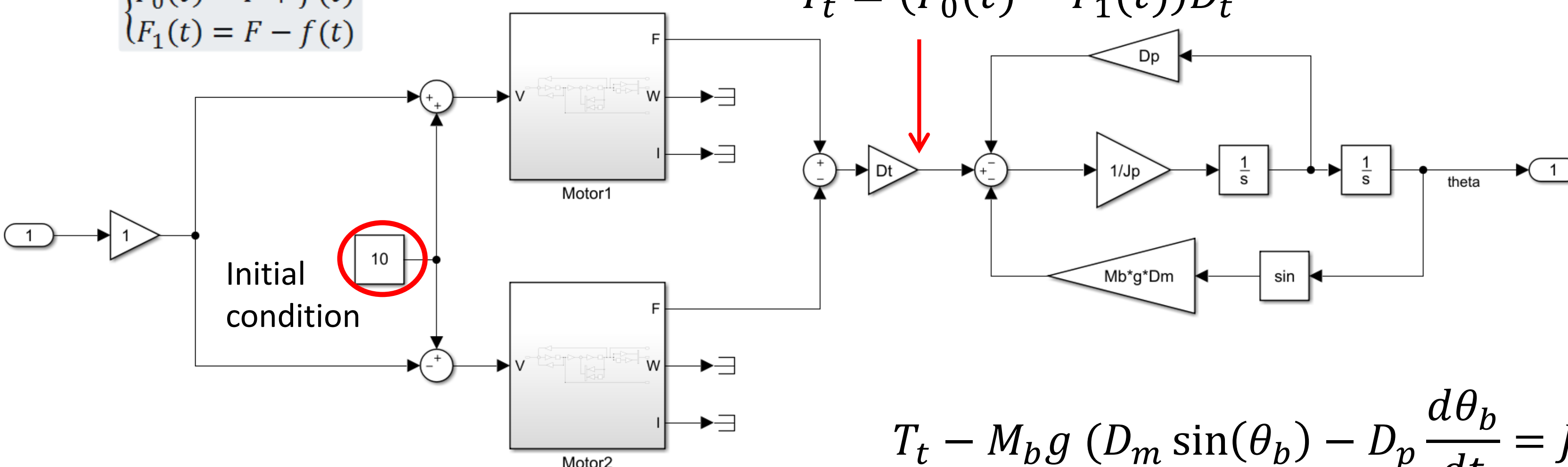




Question 8: the Simulink simulation model QUADROTOR CONFIGURATION

$$\begin{cases} F_0(t) = F + f(t) \\ F_1(t) = F - f(t) \end{cases}$$

$$T_t = (F_0(t) - F_1(t))D_t$$



$$T_t - M_b g \left(D_m \sin(\theta_b)\right) - D_p \frac{d\theta_b}{dt} = J_p \frac{d^2\theta_b(t)}{dt^2}$$

Question 9: find the equilibrium point

```
x0=[];u0=0;y0=[]; %Input is initialized at 0V
ix=[];
iu=1; %Input 1 is freezed
iy=[];
[xtrim,utrim,ytrim,dxtrim] = trim('Mechanical_sol',x0,u0,y0,ix,iu,iy)
```

The system is of order 6 (two states for each motors and two states more for the pendulum)

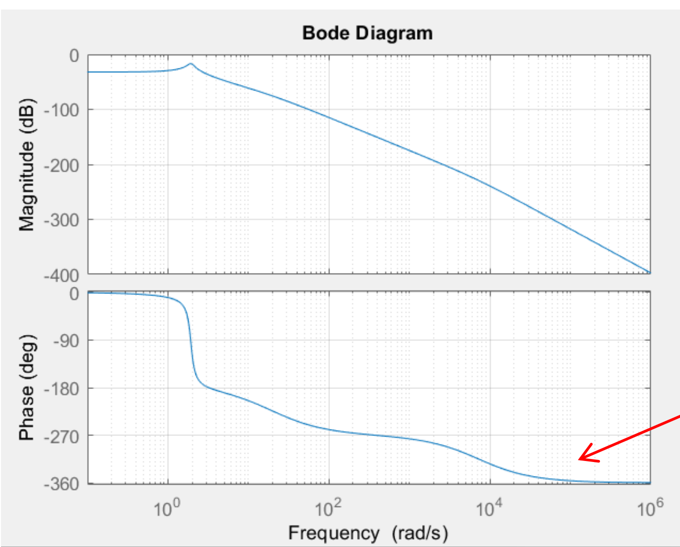
```
xtrim =

    0.0000
  224.4221
  224.4221
   -0.0000
    0.0684
    0.0684
```

Question 10: find the linearized model

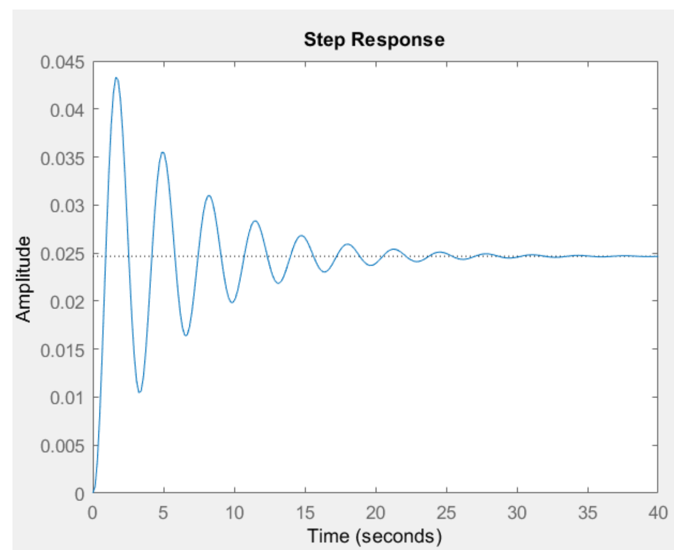
```
[A,B,C,D] = linmod('Mechanical_sol', xtrim, utrim)
Gmec=ss(A,B,C,D)
```

6 poles

Pole	Damping	Frequency (rad/TimeUnit)	Time Constant (TimeUnit)
-1.65e-01 + 1.92e+00i	8.55e-02	1.93e+00	6.06e+00
-1.65e-01 - 1.92e+00i	8.55e-02	1.93e+00	6.06e+00
-2.00e+01	1.00e+00	2.00e+01	5.00e-02
-7.22e+03	1.00e+00	7.22e+03	1.38e-04
-7.22e+03	1.00e+00	7.22e+03	1.38e-04
-2.00e+01	1.00e+00	2.00e+01	5.00e-02



Order 4 transfer function



Question 11: observability and governability

Compute controlability matrix

```
Co = ctrb(A,B);
```

Determine the number of uncontrollable states.

```
unco = length(A) - rank(Co)
```

unco = 4

Do the same for observability

```
Obs = obsv(A,C);
unobs = length(A) - rank(Obs)
```

unobs = 2

Question 12: model reduction

Use minreal to obtain the pole-zero cancelled transfer function

```
Gtf_reduced=minreal(Gmec)
[A,B,C,D]=ssdata(Gtf_reduced)
bode(Gtf_reduced)
damp(A)
```

```
        Pole                Damping       Frequency       Time Constant
                                        (rad/TimeUnit)     (TimeUnit)

 -7.22e+03                 1.00e+00       7.22e+03          1.38e-04
 -2.00e+01                 1.00e+00       2.00e+01          5.00e-02
 -1.65e-01 + 1.92e+00i     8.55e-02       1.93e+00          6.06e+00
 -1.65e-01 - 1.92e+00i     8.55e-02       1.93e+00          6.06e+00
```