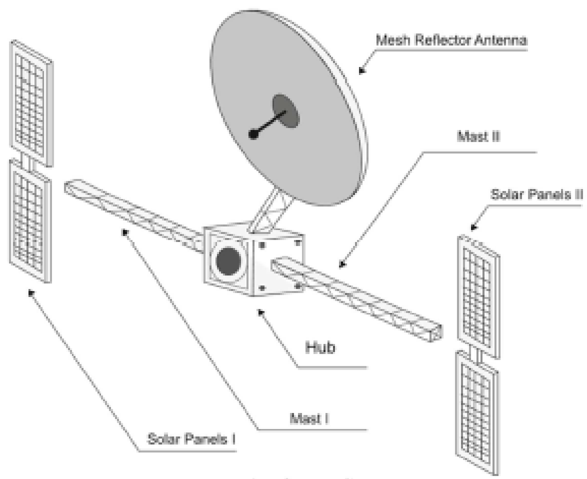
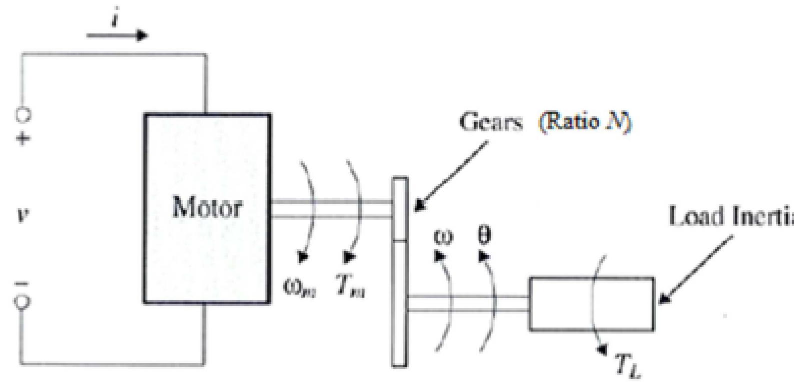
LAB 2

Satellite antenna with flexible solar panels





data for the gear motor

$N = 300$ ratio of reduction

$J_m = 0.001\ m^2kg$ DC motor inertia

$k_e = k_\Gamma = 0.2$ en V/rad/s ou en Nm/A torque constant

$R_e = 2\ \Omega$

$L_e = 2$ mH

data for the load

$J_l = 100\ m^2kg$ load inertia

K_l= 1e4 N/m load stiffness N/m

b = 10 Viscous coefficient

Write the electrical equation of the motor and the mechanical equation of the gear motor connected to the load.

Electrical law:

Kirchoff second law applied to the motor:

$$L\frac{di}{dt} + Ri = v - emf$$

with emf : electromotrice force

$$emf = k_e \omega_m$$

Mechanical law:

Newton second law applied to the motor:

$$J_m \dot{\omega}_m = T_m - T_e$$

Newton second law applied to the load:

$$J\dot{\omega} = NT_e - T_L$$

with $\omega_m = N\omega$ and $T_m = k_m I$

Case $T_L = 0$ - Compute the transfer function $\frac{\theta}{v}$

Apply Laplace transform

$$(Ls + R)i = v - emf = v - k_e\omega_m \quad \Rightarrow \quad i = \frac{v - k_e\omega_m}{(Ls+R)}$$

$$J_m\dot{\omega}_m = T_m - T_e \Rightarrow J_m s\omega_m = T_m - T_e \Rightarrow T_e = T_m - J_m s\omega_m$$

$$J\dot{\omega} = NT_e - T_L \Rightarrow Js\omega = NT_e - T_L \Rightarrow Js\omega = NT_e - T_L = NT_e = N(T_m - J_m s\omega_m)$$

$$Js\omega = N(\mathrm{k_m I} - J_m s\omega_m) = N(\mathrm{k_m I} - J_m sN\omega_{\square}) \Rightarrow J_T s\omega = \mathrm{Nk_m I}$$

$$J_T s\omega = N\mathrm{k_m}\frac{v - k_e\omega_m}{(Ls+R)} = N\mathrm{k_m}\frac{v - k_e N\omega_{\square}}{(Ls+R)}$$

$$\Rightarrow \omega(s) = \frac{Nk_m v(s)}{\left(J_T L_e s^2 + J_T R_e s + k_e k_m N^2\right)} \Rightarrow \theta(s) = \frac{Nk_m v(s)}{\left(J_T L_e s^2 + J_T R_e s + k_e k_m N^2\right)s}$$

|

Poles of the transfer function/ Bode diagrams and Margins

method 1: define the numerator and denominator and use the function *roots*
method 2: define a system (function *tf*) and use the function *zpkdata*

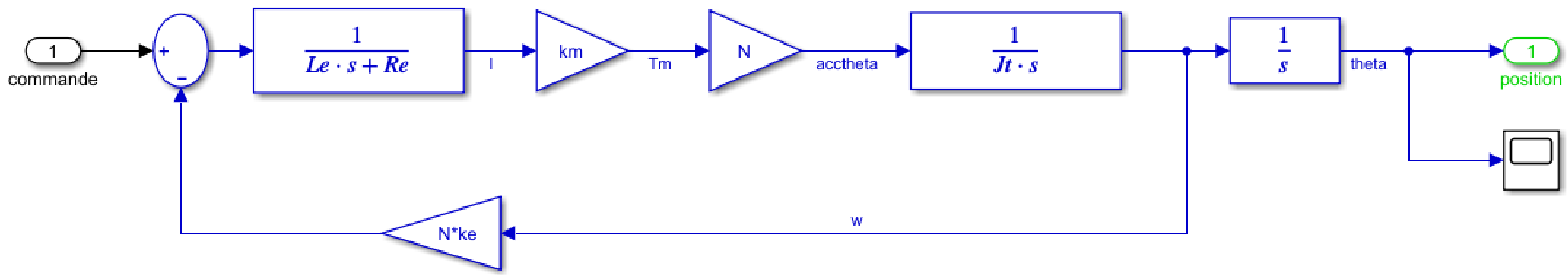
With Matlab, plot the Bode diagrams (function *bode*), the Nichols diagram (function *nichols*), and the root locus (function *rlocus*). Try also the LTIview Graphical Interface (function *ltiview*)

```
% model by transfer function (without load)
num = N*km;
den = [Jt*Le Jt*Re N^2*km*ke 0];
roots(den)
sys = tf(num,den);
[z,p,k]=zpkdata(sys,'v')

% results
figure(1);
bode(sys);grid on;hold on;
figure(2);
rlocus(sys)
ltiview(sys)

% margin
[Gm,Pm,Wcg,Wcp] = margin(sys)
MdB = 20*log10(Gm)
```

With Simulink, make the bock diagram of the system (use inport and outport blocks to define input and output). Give a name. Here: open_loop_satellite_load



With Matlab

sys1 = linmod('open_loop_satellite_load');
open_loop_satellite = ss(sys1.a,sys1.b,sys1.c,sys1.d)
figure(1);
bode(open_loop_satellite);grid on;hold on;

Linmod: command to get a state-space representation from a Simulink diagram

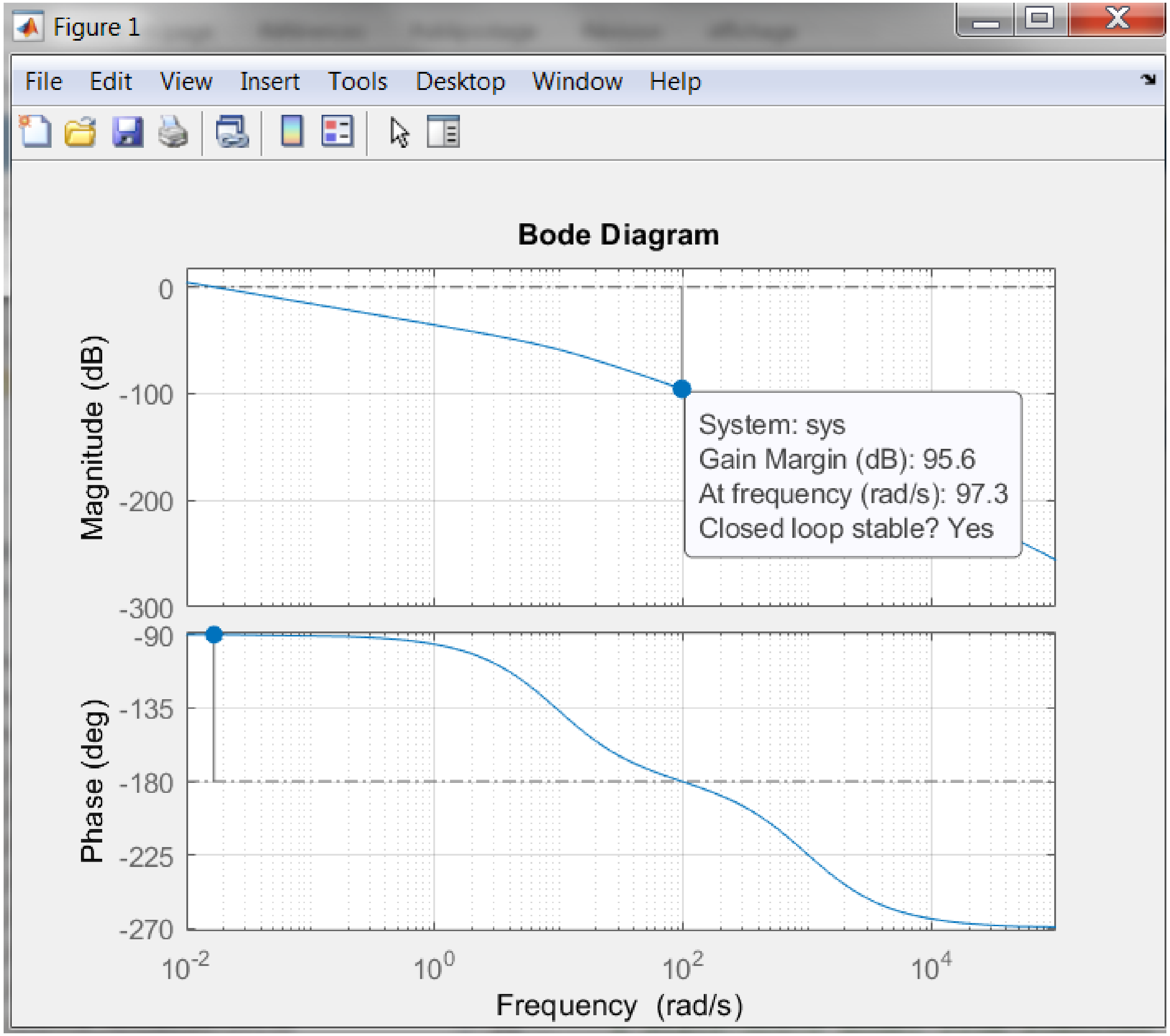
eig(sys1.a) => 3 eigenvalues = poles

```
         0
   -9.5652
 -990.4348
```

damp(sys1.a) => poles + damping + pulsation (frequency in rad)

Pole	Damping	Frequency (rad/TimeUnit)	Time Constant (TimeUnit)
0.00e+00	-1.00e+00	0.00e+00	Inf
-9.57e+00	1.00e+00	9.57e+00	1.05e-01
-9.90e+02	1.00e+00	9.90e+02	1.01e-03

Bode diagram



Gain margin = 95.6dB

Phase margin = 90°

Root locus

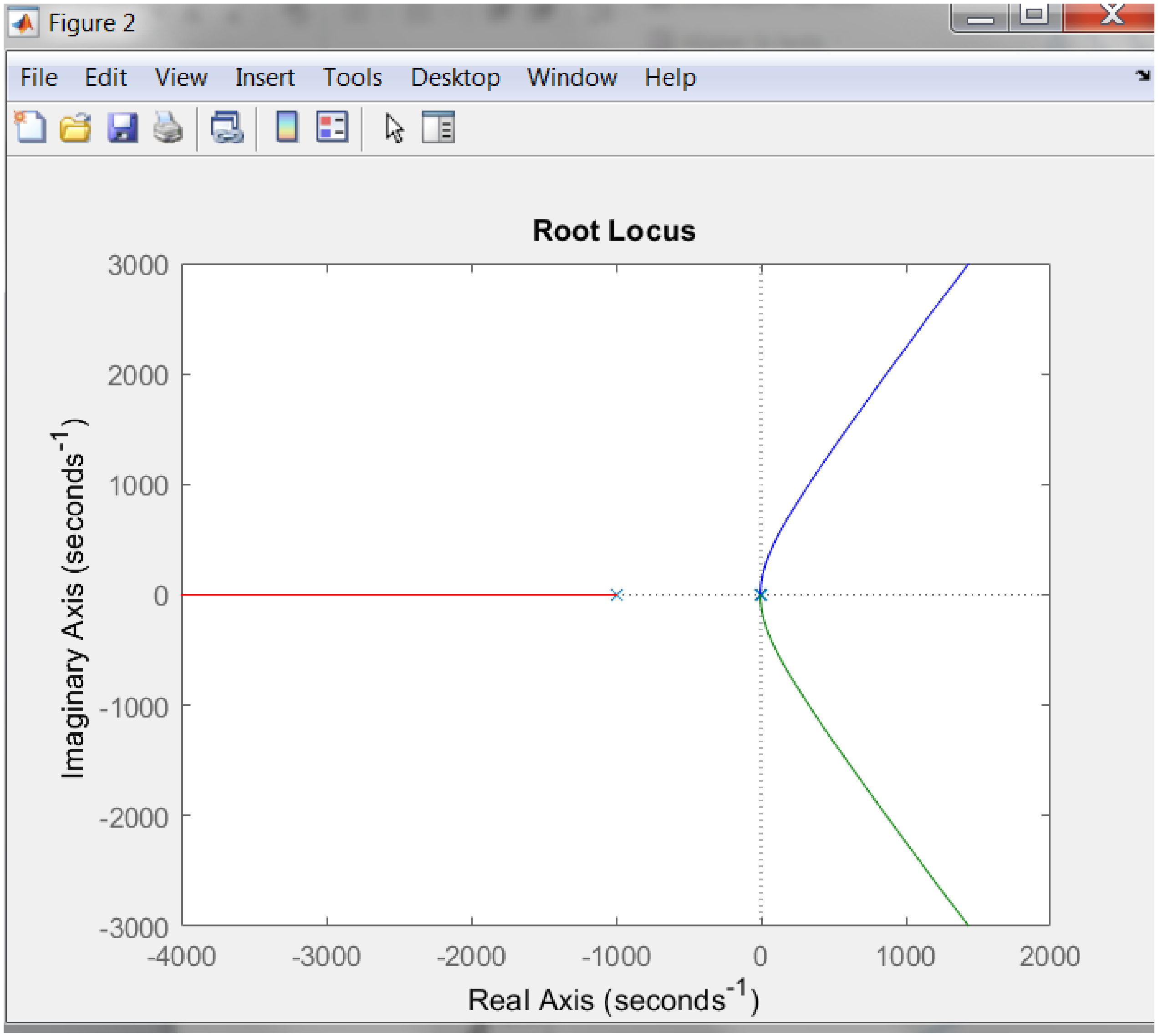
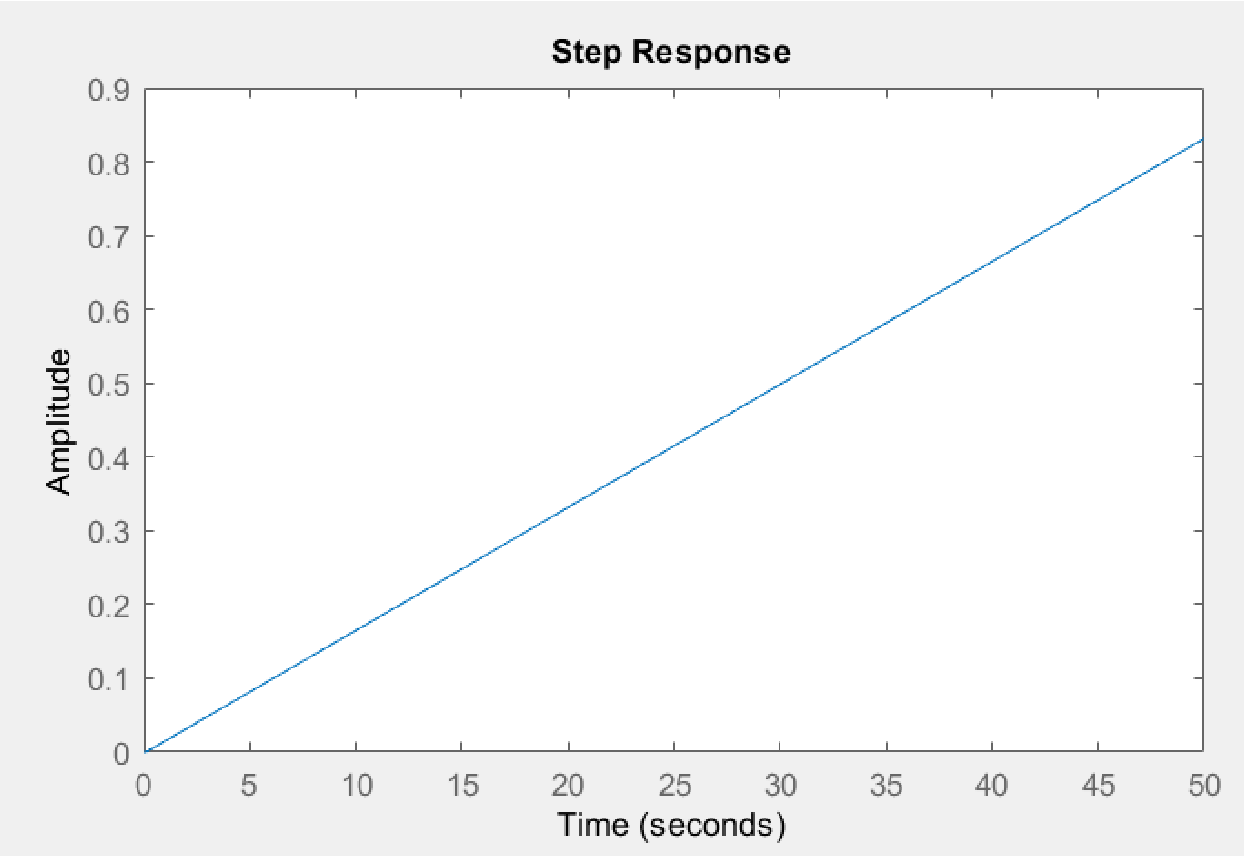


Figure 2
File Edit View Insert Tools Desktop Window Help
Root Locus
3000
2000
1000
0
-1000
-2000
-3000
-4000
-3000
-2000
-1000
0
1000
2000
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})

Time response - no load

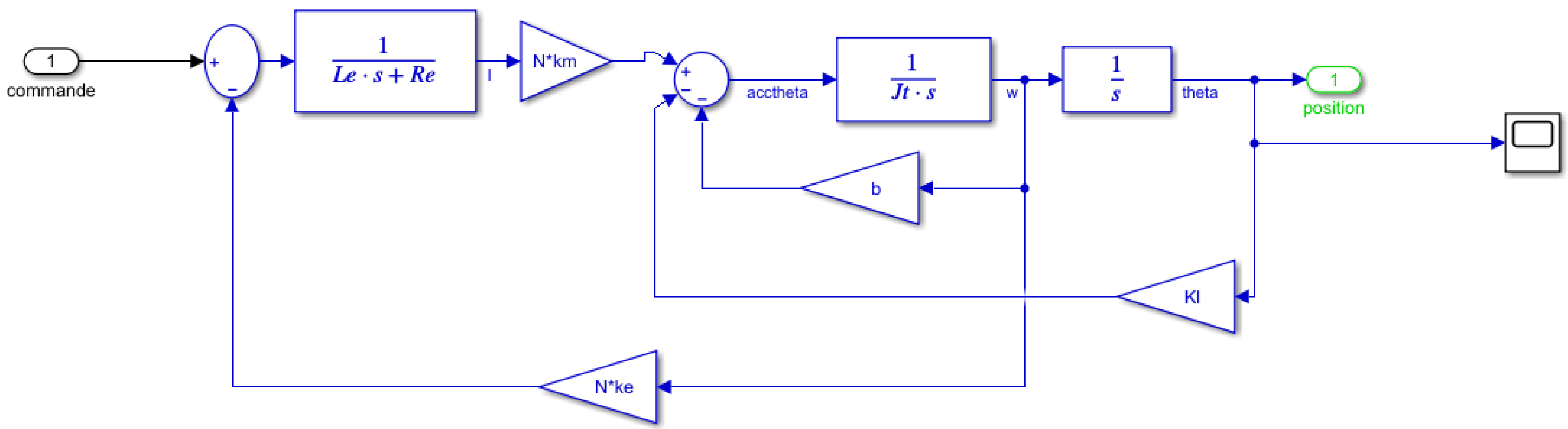
- Look at the effect of the integrator!



With load

<u>$Case\ T_L = b\dot{\theta} + K_l\theta$</u>

$$\theta(s) = \frac{Nk_m v(s)}{(J_T L_e s^3 + (J_T R_e + L_e b)s^2 + (k_e k_m N^2 + R_e b + L_e K_l)s + R_e K_l)}$$



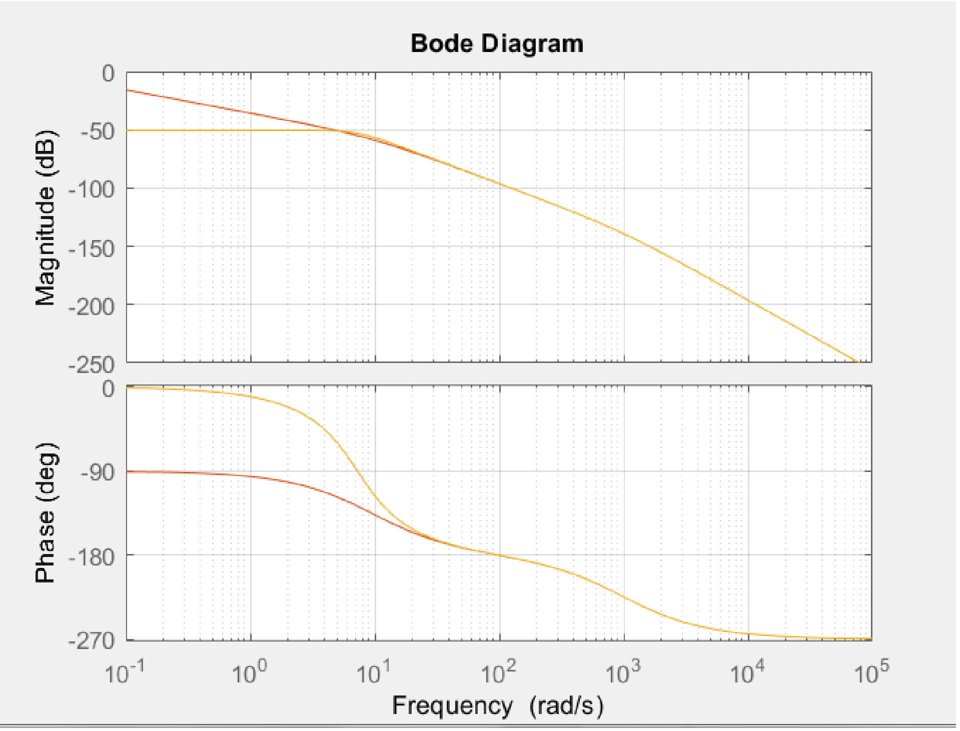
3 poles: 1 single pole and 1 double pole

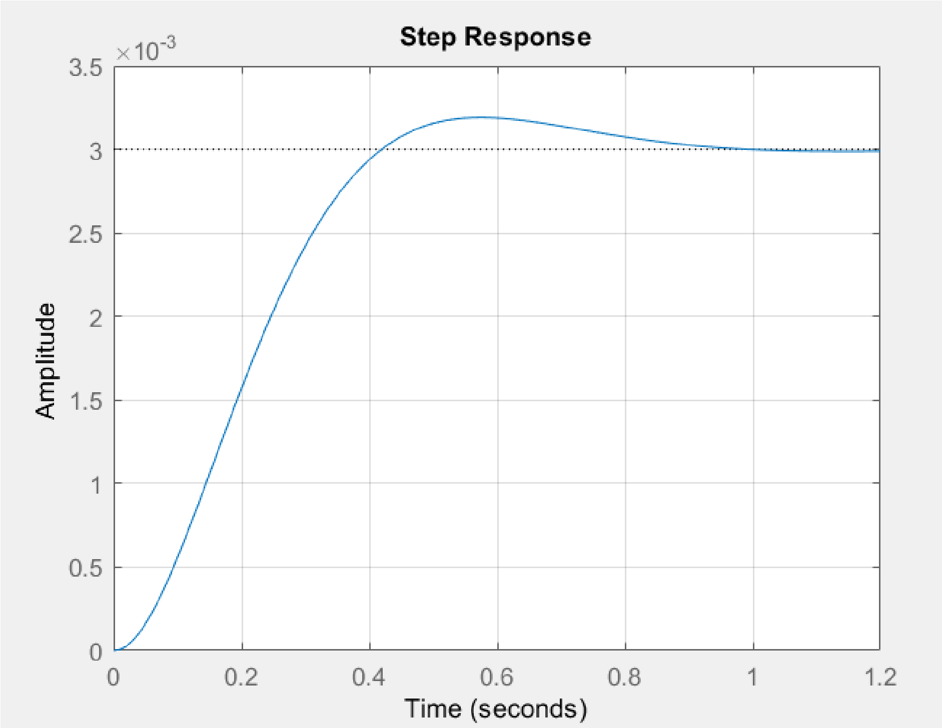
Pole	Damping
-4.81e+00 + 5.48e+00i	6.60e-01
-4.81e+00 - 5.48e+00i	6.60e-01
-9.90e+02	1.00e+00

Performance analysis – Effect of the load

Pole	Damping	Frequency (rad/seconds)
-4.81e+00 + 5.48e+00i	6.60e-01	7.29e+00
-4.81e+00 - 5.48e+00i	6.60e-01	7.29e+00
-9.90e+02	1.00e+00	9.90e+02

Gain margin = 95.6dB
Phase margin = Inf





State-space representation

The state-space representation is not unique!
You get one using the function linmod.

You can get another one from the equations:

$$T_L = b\omega + K_L\theta$$

$$\dot{\omega} = \frac{N}{J_T}k_m i - \frac{1}{J_T}T_l = \frac{N}{J_T}k_m i - \frac{1}{J_T}(K_L\theta + b\omega)$$

$$L\frac{di}{dt} + Ri = v - Nk_e\omega \Rightarrow \frac{di}{dt} = -\frac{R}{L}i + \frac{1}{L}v - \frac{Nk_e}{L}\omega$$

$$\frac{d}{dt}\begin{bmatrix}\theta\\ \omega\\ i\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ -\frac{K_L}{J_T} & -\frac{b}{J_T} & \frac{Nk_m}{J_T}\\ 0 & -\frac{Nk_e}{L} & -\frac{R}{L}\end{bmatrix}\begin{bmatrix}\theta\\ \omega\\ i\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \frac{1}{L}\end{bmatrix}v$$

If θ is the output, $Y = \begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}\theta\\ \omega\\ i\end{bmatrix}$

Governability - observability

```
% X = [theta   omega  current ]
sys3.a = [ 0 1 0 ;-Kl/Jt -b/Jt N*km/Jt; 0 -ke*N/Le  -Re/Le]
sys3.b = [0 ; 0 ; 1/Le]
sys3.c = [1 0 0];
sys3.d = 0;
open_loop_satellite3 = ss(sys3.a,sys3.b,sys3.c,sys3.d)
figure(2);bode(open_loop_satellite3);grid on;
eig(sys3.a)
damp(sys3.a)

% governability - observability
O = rank(obsv(sys3.a ,sys3.c))
G = rank(ctrb(sys3.a ,sys3.b))
```

The rank of the observability matrix is equal to the the order of the system
⇨ the systemis observable.
The rank of the governability matrix is equal to the the order of the system
⇨ the systemis governable.

Time-domain analysis of the system

What kind of basic systems composes the system?
One first order system and one second order system

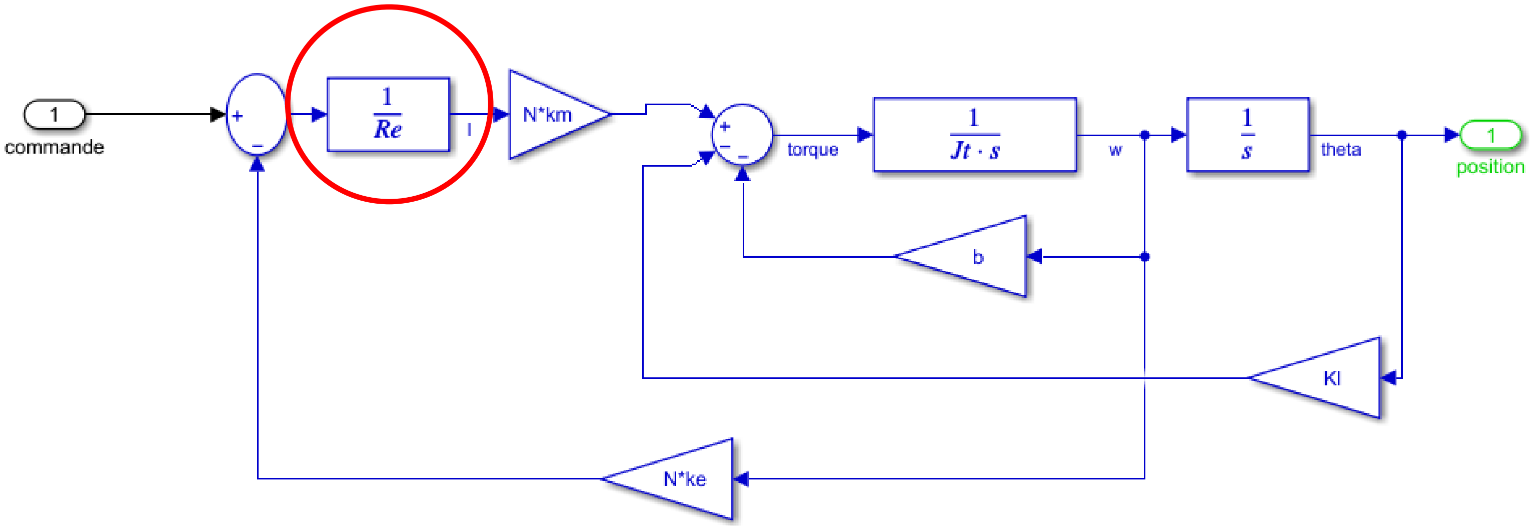
Use the natural frequencies or the cut-off frequencies of the system to compute the time response at 5% (for the second order system) or the time response for the first order system.
first order system => time response ~ 1ms
second order system => time response ~ 0.82s

Compute the settling time response at 5% of the system using the step response plotted with Matlab.
time response ~ 0.82s

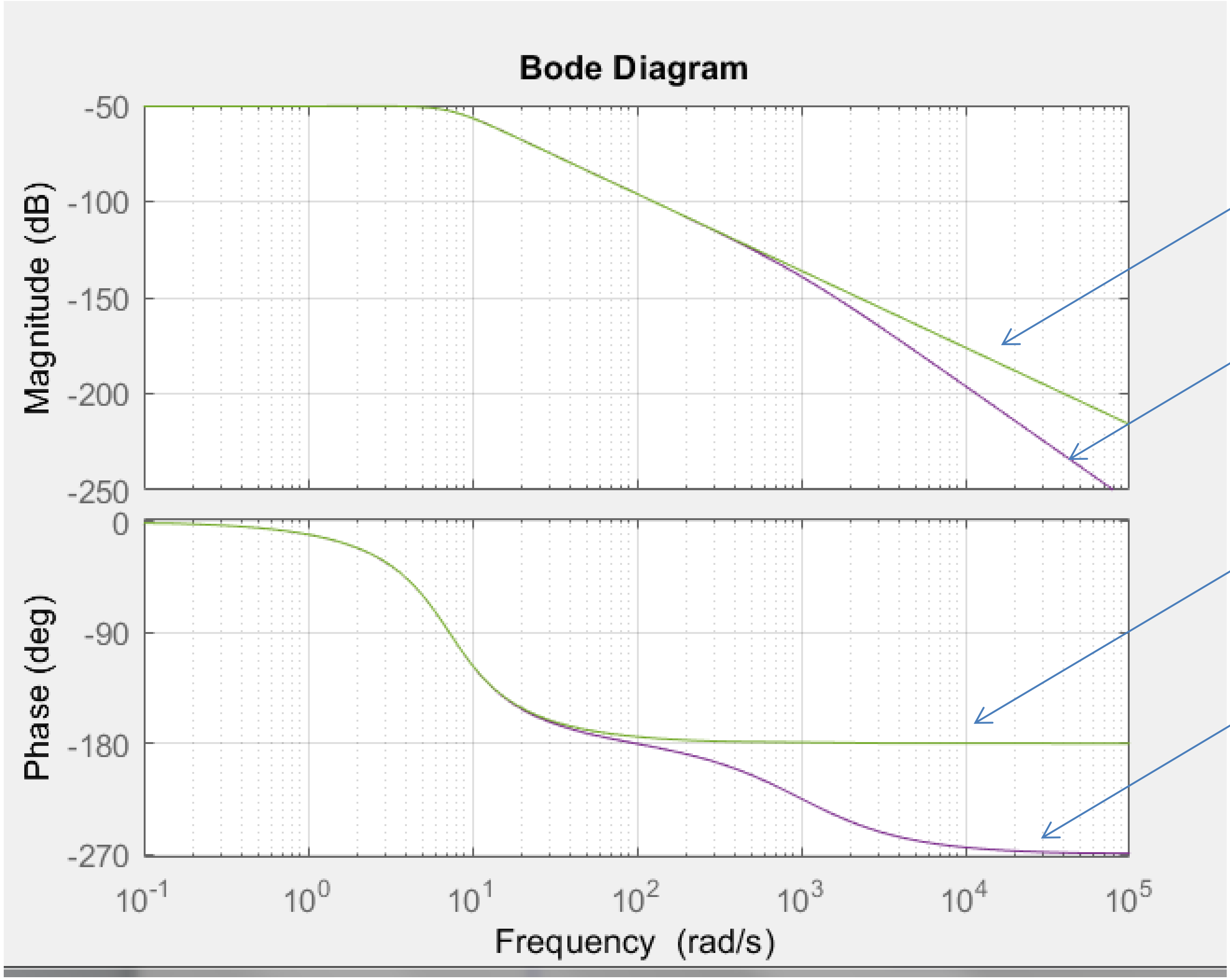
Conclusion : the dynamics of tjhe system is imposed by the second order system.
We can reduce the system.

Simulink diagram of the reduced system

Replace the block $\frac{1}{L_e s + R_e}$ by $\frac{1}{R_e}$ in the block diagram



Bode diagram of the reduced system % Bode diagram of the initial system



Reduced system
Order 2

Initial system
Order 3

Reduced system
Order 2

Initial system
Order 3

Step response of the reduced system % Step response of the initial system

Same response