

# UCL Mechanical Engineering 2021/2022

## MECH0023 Coursework

RFLH9

December 9, 2021

### Contents

<b>1</b>	<b>Satellite model</b>	<b>1</b>
1.1	Description of model and assumptions . . . . .	1
1.2	Minimum thickness of beam . . . . .	3
1.3	Amplitude of displacement for steady-state oscillation of the sensor during rocket function	6
1.4	Limitations of analysis . . . . .	7
1.5	Manoeuvre . . . . .	7
1.5.1	Description of obtaining sensor response to the thrust force . . . . .	7
1.5.2	Displacement response . . . . .	8
<b>2</b>	<b>Linear modelling</b>	<b>9</b>
2.1	Description of non-linearity between $y(t)$ & $\theta(t)$ . . . . .	9
2.2	Description of non-linear relationship in a real system . . . . .	10
<b>3</b>	<b>Linear model performance and stability</b>	<b>10</b>
3.1	Unstable process + description of actuator switching on . . . . .	10
3.2	Root locus and effect of proportional control . . . . .	10
3.3	Negative velocity feedback controller . . . . .	11
3.3.1	Plot of root locus . . . . .	11
3.3.2	Areas where damping ratio is higher than 0.6 + damped natural frequency is higher than $7 \text{ rad s}^{-1}$ . . . . .	12

3.3.3	Range of $k$ where performance specifications are achieved . . . . .	13
<b>4</b>	<b>Controller techniques</b>	<b>14</b>
4.1	Description of closed loop position control system . . . . .	14
4.2	Specify disturbances . . . . .	14
<b>5</b>	<b>Inverted pendulum</b>	<b>15</b>
5.1	Identification and description of assumptions and limitations for the model of a rotary inverted pendulum . . . . .	15
5.2	Moving rocket to final position . . . . .	15
<b>6</b>	<b>Multi-Degree-of-Freedom systems</b>	<b>16</b>
6.1	Example of a system with multiple degrees-of-freedom . . . . .	16

## List of Figures

1	Satellite oscillation. . . . .	1
2	Supporting beam diagram. . . . .	2
3	Model of system. . . . .	2
4	Diagram to show beam dimensions. . . . .	3
5	Plot to show 1.22. . . . .	5
6	Response of sensor. . . . .	9
7	Root locus of system. . . . .	11
8	Root locus of system with negative velocity feedback controller. . . . .	12
9	Regions on root locus diagram where $\zeta \geq 0.6$ and $\omega_d \geq 7$ . . . . .	13
10	Minimum $k$ value. . . . .	14
11	Maximum $k$ value. . . . .	14
12	Stages of rocket aligning with base pad. . . . .	16
13	Physical model of suspension system. . . . .	16
14	Model of MDOF system. . . . .	17

# 1 Satellite model

## 1.1 Description of model and assumptions

We are given some information about some parameters of the system. They are:

- Beam cross-section is rectangular.
- Beam width is 5 cm.
- Beam material is high-grade aluminium alloy with modulus of elasticity 70 GPa.
- Beam length is 1.1 m.
- Sensor weight is 1.2 kg.

Hence, we can begin to start modelling our system. Our first assumption is that the satellite is oscillating only in the  $x$ -axis as show in Figure 1.

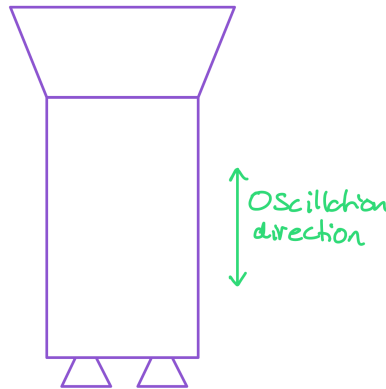


Figure 1: Satellite oscillation.

We can now simplify our supporting beam to a fixed cantilever beam with a mass at the free end. The vibrating base seen in Figure 2 is in reference to the oscillations that would be transmitted through the satellite frame to the base of the cantilever beam.

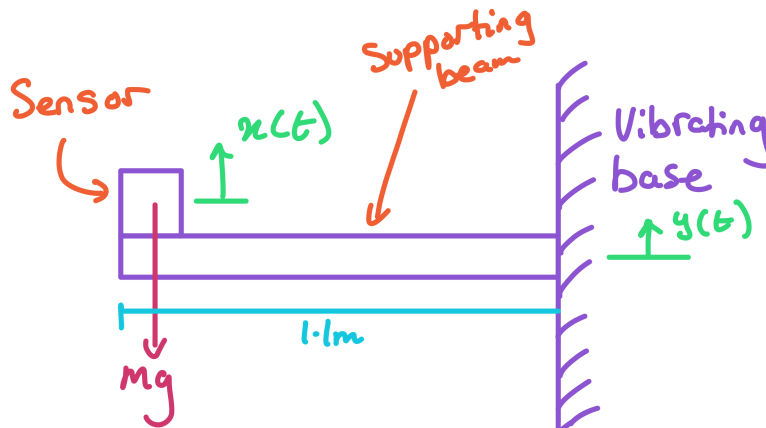


Figure 2: Supporting beam diagram.

Finally, we can model the sensor system as a mass attached to a base via a spring and a damper. As our satellite's environment will drastically change during the course of its journey to space (atmosphere  $\rightarrow$  vacuum of space), we will see a change in the way viscous damping (atmospheric drag) will affect our system. We also have frictional losses from the cantilever support and material damping in our system. We can simplify our model by considering the damping force to be constant throughout the operational timeframe of the satellite.

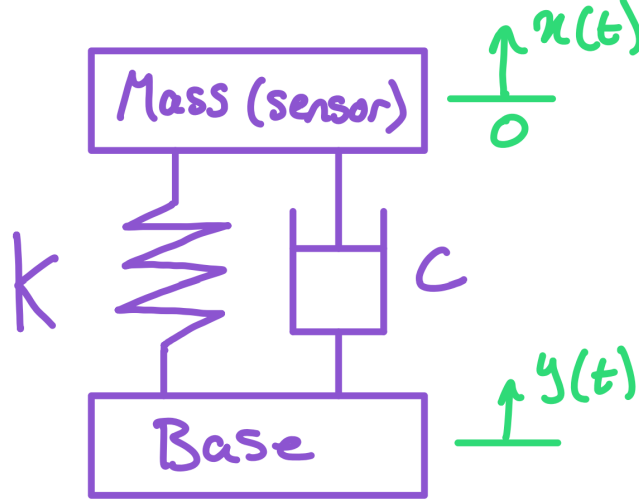


Figure 3: Model of system.

## 1.2 Minimum thickness of beam

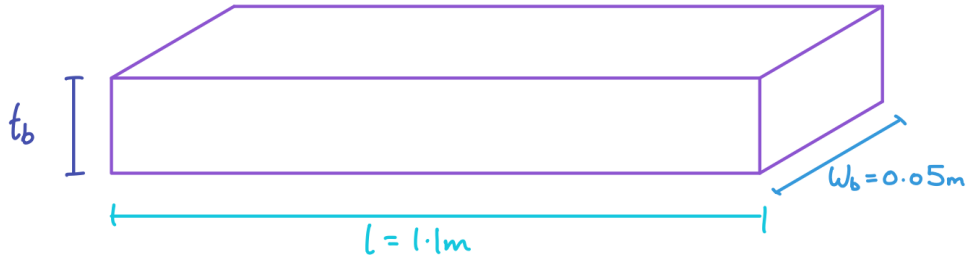


Figure 4: Diagram to show beam dimensions.

The relative motion of the mass from the base can be set as:

$$z = x - y \quad (1.1)$$

Next we can look at the equation of motion for the system:

$$m\ddot{x} + c(\dot{x} - \dot{y}) + k(x - y) = 0 \quad (1.2)$$

Substituting 1.1 into 1.2:

$$m\ddot{z} + m\ddot{y} + c\dot{z} + kz = 0 \quad (1.3)$$

We are given that  $\zeta = 0.003$ , hence our system is underdamped. The standard solution for 1.3 for underdamped conditions is:

$$z(t) = Ae^{-\zeta\omega_n t} \sin(\omega_d t + \phi) \quad (1.4)$$

where:

$$A = \frac{\sqrt{(\dot{z}_0^2 + \zeta\omega_n z_0)^2 + (\omega_d z_0)^2}}{\omega_d} \quad (1.5)$$

$$\phi = \arctan\left(\frac{z_0\omega_d}{\dot{z}_0 + \zeta\omega_n z_0}\right) \quad (1.6)$$

Boundary conditions:

$$\dot{z}_0 = 0.04 \text{ m s}^{-1} \quad (1.7)$$

$$z_0 = 0 \text{ m} \quad (1.8)$$

$$z(t = 900 \text{ s}) < 0.0001 \text{ m} \quad (1.9)$$

Inputting boundary conditions into 1.5 and 1.6:

$$A = \frac{\sqrt{(0.04^2 + 0.0003 \cdot \omega_n \cdot 0)^2 + (\omega_d \cdot 0)^2}}{\omega_d} \quad (1.10)$$

$$A = \frac{0.04}{\omega_d} \quad (1.11)$$

$$\phi = \arctan\left(\frac{0 \cdot \omega_d}{0.04 + 0.0003 \cdot \omega_n \cdot 0}\right) \quad (1.12)$$

$$\phi = 0 \quad (1.13)$$

We know that:

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \quad (1.14)$$

Substituting 1.11, 1.13, 1.14 into 1.4:

$$z(t) = \frac{0.04e^{-\zeta\omega_n t}}{\omega_n \sqrt{1 - \zeta^2}} \sin\left(\omega_n \sqrt{1 - \zeta^2} t\right) \quad (1.15)$$

Next, we must determine  $\omega_n$ , we can find the natural frequency of the cantilever beam with simulated load at free end from our work in the MECH0013 module. The deflection at the free end  $\delta_{fe}$  is given by:

$$\delta_{fe} = \frac{mgl^3}{3EI} \quad (1.16)$$

We also know:

$$k = \frac{mg}{\delta_{fe}} = \frac{3EI}{l^3} \quad (1.17)$$

Moment of inertia:

$$I = \frac{w_b t_b^3}{12} \quad (1.18)$$

Therefore:

$$k = \frac{Ew_b t_b^3}{4l^3} \quad (1.19)$$

$k$  is related to the natural frequency through:

$$\omega_n = \sqrt{\frac{k}{m}} \quad (1.20)$$

Looking at the amplitude term of 1.15, we know this must be less than 0.0001 m at  $t = 900$  s:

$$\frac{0.04e^{-\zeta\omega_n t}}{\omega_n \sqrt{1-\zeta^2}} < 0.0001 \quad (1.21)$$

Solving:

$$\frac{0.04}{\sqrt{\frac{Ew_b t_b^3}{4l^3 m}} \sqrt{1-\zeta^2}} \cdot e^{-\zeta \sqrt{\frac{Ew_b t_b^3}{4l^3 m}} t} < 0.0001 \quad (1.22)$$

$$\frac{0.04}{\sqrt{\frac{70 \cdot 10^9 \cdot 0.05 \cdot t_b^3}{4 \cdot 1.1^3 \cdot 1.2}} \sqrt{1-0.003^2}} \cdot e^{\left(-0.003 \sqrt{\frac{70 \cdot 10^9 \cdot 0.05 \cdot t_b^3}{4 \cdot 1.1^3 \cdot 1.2}} \cdot 900\right)} < 0.0001 \quad (1.23)$$

The result was inputted into MATLAB and a numerical answer for  $t_b$  was obtained. The MATLAB code used is below.

```

1  clc
2  clear
3  close all
4
5  %def variables
6  m = 1.2;
7  E = 70e9;
8  l = 1.1;
9  wb = 0.05;
10 zeta = 0.003;
11 z0 = 0;
12 zdot0 = 0.04;
13 t = 900;
14
15 %symbolise thickness var
16 syms tb
17
18 %define eqn
19 eqn = ((zdot0)/((((E*wb*(tb^3))/(4*(l^3)*m))^(1/2))*((1 - (zeta^2))^(1/2))
20         ))*(exp(-zeta*(((E*wb*(tb^3))/(4*(l^3)*m))^(1/2))*900)) == 0.0001;
21
22 %graph eqns
23 eqnLeft = ((zdot0)/((((E*wb*(tb^3))/(4*(l^3)*m))^(1/2))*((1 - (zeta^2))
24         ^ (1/2))))*(exp(-zeta*(((E*wb*(tb^3))/(4*(l^3)*m))^(1/2))*900));
25 eqnRight = 0.0001;
26 fplot([eqnLeft eqnRight])
27 xlim([-0.005 0.005])%limits refined through visual inspection of plots
28 ylim([5e-5 15e-5])
29
30 %test eqn
31 %eqn2 = ((0.04)/((sqrt(((70*10^9)*0.05*(tb^3))/(4*(1.1^3)*1.2)))*(sqrt(1
32         - (0.003^2)))))*(exp(-0.003*(sqrt(((70*10^9)*0.05*(tb^3))/(4*(1.1^3)
33         *1.2)))*900)) == 0.0001;
34
35 %numerically solve with given starting value obtained from graphical

```

```

32 %inspection
33 v = vpasolve(eqn,tb,0.001);

```

This generated the plot shown in Figure 5

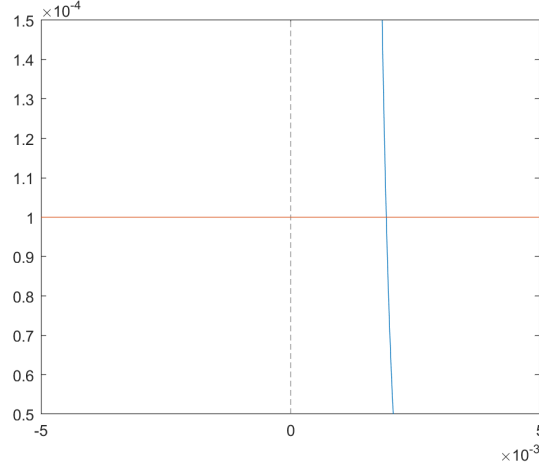


Figure 5: Plot to show 1.22.

Using MATLAB's `vpasolve` function, we can numerically solve our equation. This function can take an `init_param` variable. This allows MATLAB to solve for values close to the variable. By visually observing the plot, we can see that our solution is in the vicinity of 0.001. Hence, this was chosen as the initial parameter. This yielded:

$$t_b \approx 0.00192 \text{ m} \quad (1.24)$$

$$t_b \approx 1.92 \text{ mm} \quad (1.25)$$

### 1.3 Amplitude of displacement for steady-state oscillation of the sensor during rocket function

This is a case of base excitation. The equation to describe the motion of our base excitation takes the standard form:

$$y(t) = Y \sin(\omega t) \quad (1.26)$$

Therefore:

$$m\ddot{x} + c(\dot{x} - \dot{y}) + k(x - y) = 0 \quad (1.27)$$

$$m\ddot{x} + c\ddot{x} + kx = c\dot{y} + ky \quad (1.28)$$

Substituting 1.26:

$$m\ddot{x} + c\dot{x} + kx = c\omega Y \cos(\omega t) + kY \sin(\omega t) \quad (1.29)$$

$$= A \sin(\omega t - \alpha) \quad (1.30)$$

where

$$A = Y \sqrt{k^2 + (c\omega)^2} \quad (1.31)$$

$$\alpha = \arctan\left(-\frac{c\omega}{k}\right) \quad (1.32)$$

Our solution is then:

$$x_p(t) = \frac{Y\sqrt{k^2 + (c\omega)^2}}{\sqrt{(k - m\omega^2)^2 + (c\omega)^2}} \cdot \sin(\omega t - \alpha - \phi_1) \quad (1.33)$$

where

$$\phi_1 = \arctan\left(\frac{c\omega}{k - m\omega^2}\right) \quad (1.34)$$

We can reduce our equation further to:

$$x_p = X \sin(\omega t - \phi) \quad (1.35)$$

where

$$\frac{X}{Y} = \left[ \frac{1 + (2\zeta r)^2}{(1 - r^2)^2 + (2\zeta r)^2} \right]^{\frac{1}{2}} \quad (1.36)$$

The frequency of base excitation from the power spectrum graph is approximately:

$$f \approx 240 \text{ Hz} \quad (1.37)$$

Reading the gain from our power spectrum graph, we can find  $Y$ :

$$10^{\frac{-60}{20}} = \frac{Y\sqrt{2}}{2} \quad (1.38)$$

$$Y = 0.001\sqrt{2} \quad (1.39)$$

Using MATLAB, the various parameters were filled and the equation was solved to find  $X$ :

```

1  clc
2  clear
3  close all
4
5  %def vars
6  E = 70e9;
7  b = 0.05;
8  t = 0.0019;
9  m = 1.2;
10 l = 1.1;
11 f = 240;
12 zeta = 0.003;
13 Y = 0.001*sqrt(2);
14
15 k = (E*b*(t^3))/(4*l^3);
16 wn = sqrt(k/m);
17 w = f*2*pi;
18 c = 2*m*zeta*wn;
19
20 r = w/wn;
21 X = Y*((1 + (2*zeta*r)^2)/(((1-(r^2))^2 + (2*zeta*r)^2))^(1/2));

```

This yielded:

$$X = 1.12 \times 10^{-8} \text{ m} \quad (1.40)$$



## 1.4 Limitations of analysis

One limitation of our model may be the sample size selected. If we do not utilise a high enough sampling rate, we may not fully capture the extent of our FRF. Our signal is also noisy, which could affect our data. We could also check a higher range of frequencies to make sure that we do not have peaks in those frequency ranges (due to the potential of a MDOF system).

## 1.5 Manoeuvre

### 1.5.1 Description of obtaining sensor response to the thrust force

As the thrust acts on the satellite frame, we can use force transmissibility to find the force on the sensor. We can then use convolution to find the response at the sensor.

### 1.5.2 Displacement response

The force transmitted to the base of a SDOF system would be:

$$m\ddot{x} + c\dot{x} + kx = F \cos \omega t \quad (1.41)$$

$$F_T = kx + c\dot{x} \quad (1.42)$$

$$\left| \frac{F_T}{F} \right| = \left[ \frac{1 + (2\zeta r)^2}{(1 - r^2)^2 + (2\zeta r)^2} \right]^{\frac{1}{2}} \quad (1.43)$$

The total response of the system would be the sum of all impulse responses:

$$x(t) = \sum F(\tau) \Delta \tau h(t - \tau) \quad (1.44)$$

Using the limit  $\Delta \tau \rightarrow 0$ , we can then use convolution:

$$x(t) = \int_0^t (F(\tau) h(t - \tau)) d\tau \quad (1.45)$$

Substituting impulse response function:

$$x(t) = \frac{1}{m\omega_d} \int_0^t \left( F(\tau) e^{-\zeta\omega_n(t-\tau)} \sin(\omega_d(t-\tau)) \right) d\tau \quad (1.46)$$

The following MATLAB code was used to produce the following plots:

```
1  clc
2  clear
3  close all
4
5  %def vars
6  E = 70e9;
7  b = 0.05;
8  t = 0.0019;
9  m = 1.2;
10 l = 1.1;
11 f = 240;
```

```

12 zeta = 0.003;
13 Y = 0.001*sqrt(2);
14
15 k = (E*b*(t^3))/(4*l^3);
16 wn = sqrt(k/m);
17 wd = wn*sqrt(1-zeta^2);
18 w = f*2*pi;
19 c = 2*m*zeta*wn;
20 r = w/wn;
21 F = 16;
22
23 FT = F*((1 + (2*zeta*r)^2)/((1-r^2)^2 + (2*zeta*r)^2))^(1/2);
24
25 syms t tau
26
27 eqn = exp(-zeta*wn*(t-tau))*sin(wd*(t-tau));
28
29 int1 = FT/(wd*m)*int(tau*eqn, tau, 0, t);
30 int2 = FT/(wd*m)*int(tau*eqn, tau, 0, t) + FT/(wd*m)*int(eqn, tau, 1, t);
31 int3 = FT/(wd*m)*int(tau*eqn, tau, 0, t) + FT/(wd*m)*int(eqn, tau, 1, 3)
      + FT/(wd*m)*int(0*eqn, tau, 3, t);
32
33 hold on
34 fplot(int1, [0,1])
35 fplot(int2, [1,3])
36 fplot(int3, [3,15])
37 grid on
38 xlabel('Time/s')
39 ylabel('Displacement/m')

```

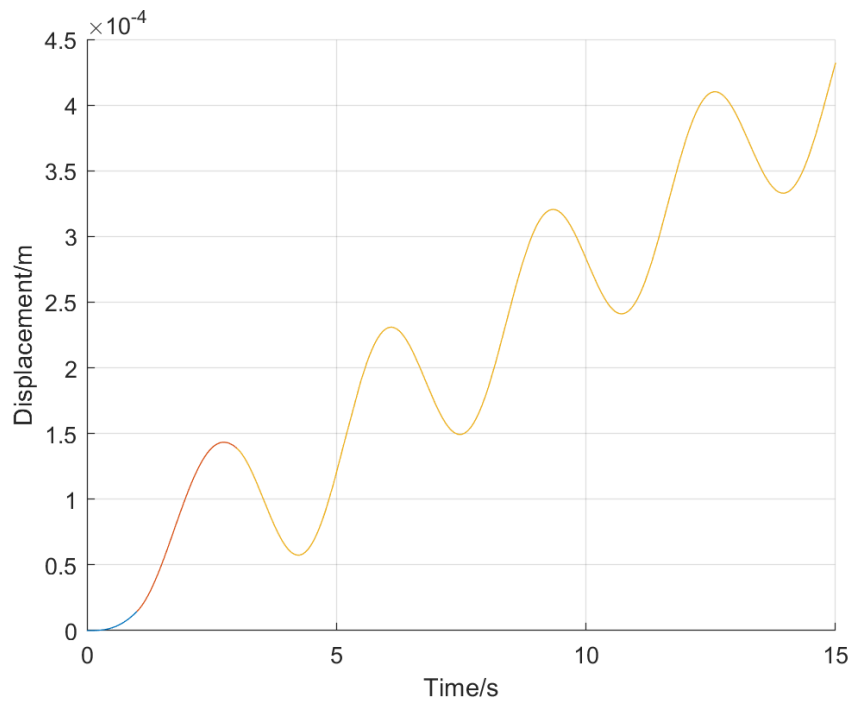


Figure 6: Response of sensor.

## 2 Linear modelling

### 2.1 Description of non-linearity between $y(t)$ & $\theta(t)$

The following function:

$$\frac{dy(t)}{dt} = 150 \cos(\theta(t)) - 47 \quad (2.1)$$

has a non-linear relationship due to  $\theta(t)$  being within a cosine function. There is also an offset value of 47 which makes our relationship non-linear as well.

### 2.2 Description of non-linear relationship in a real system

An example of a non-linear relationship is car velocity and air resistance. Our input is the velocity of the car and our output is the drag force. We can describe this relationship with the following equation.

$$F_D = \frac{1}{2} \rho v^2 C_D A \quad (2.2)$$

$$F_D \propto v^2 \quad (2.3)$$

The relationship between the drag force and velocity is proportionally squared. This means that as velocity increases, our drag force will increase by four times. Non-linearity can also come in the form as headwind or tailwind. Mathematically, this would be represented as a constant term ( $B$ ) added to our equation. For example:

$$F_D = \frac{1}{2} \rho v^2 C_D A + B \quad (2.4)$$

## 3 Linear model performance and stability

### 3.1 Unstable process + description of actuator switching on

The transfer function,  $G_p(s)$  is:

$$G_p(s) = \frac{Y(s)}{V(s)} = \frac{1}{(s^2 - 81)} \frac{k}{(0.05s + 1)} \quad (3.1)$$

By looking at the position of the roots ( $s = 9$ ,  $s = -9$ ,  $s = -20$ ), we can check whether our system is unstable. We have a root at  $s = 9$ , which is on the positive side of the s-plane. This means that our whole system is unstable.

Our poles lie on the real axis on the positive side of the s-plane. Hence, when we switch our voltage to 5 V, we will find that our rudder angle will exponentially rise. We will not see any oscillation as our response will have no sine term. The rudder angle will continually increase until it reaches its mechanical limit and potentially damage itself. The rudder angle may go in either direction (positive or negative/left or right) until it reaches this limit.

### 3.2 Root locus and effect of proportional control

Using the following MATLAB code, the root locus of the system was plotted in Figure 7.

```

1  clc
2  clear
3  close all
4
5  %define variables
6  k = 1;
7  %transfer function
8  sys = tf([k],[0.05 1 -4.05 -81]);
9  %plot root locus
10 rlocus(sys)
11 xlim([-40 40])
12 ylim([-40 40])

```

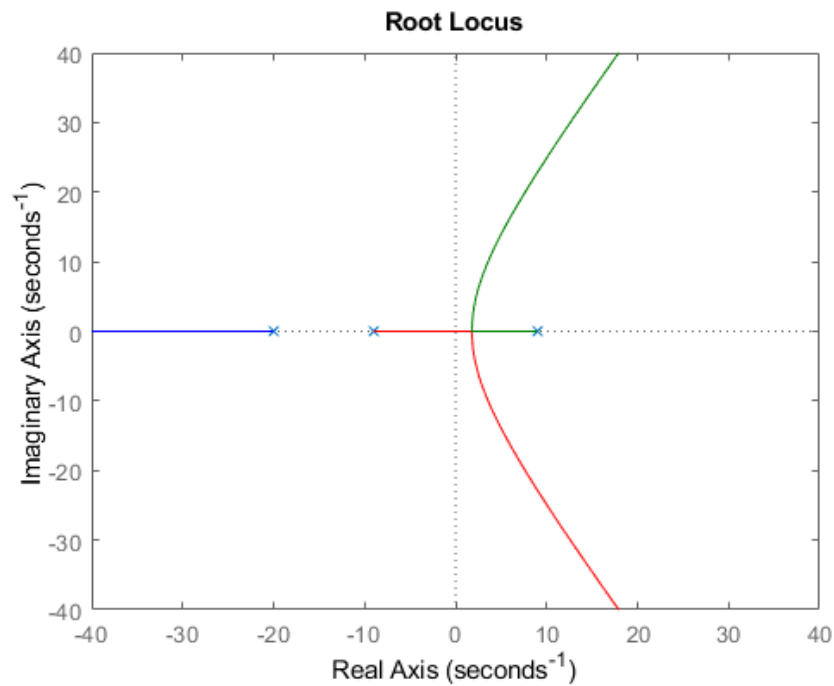


Figure 7: Root locus of system.

This root locus diagram shows us that the system would be unstable in closed-loop operation for all possible values of gain  $K$ , hence rendering proportional control ineffective for stabilising the system.

### 3.3 Negative velocity feedback controller

#### 3.3.1 Plot of root locus

Our new transfer function is:

$$H_p(s) = \frac{k(s+8)}{0.05s^3 + s^2 - 4.05s - 81} \quad (3.2)$$

Using the following MATLAB code, the root locus of the system was plotted in Figure 8.

```

1  clc
2  clear

```

```

3 close all
4
5 %define variables
6 k = 1;
7 %transfer function
8 sys = tf([k 8*k],[0.05 1 -4.05 -81]);
9 %plot root locus
10 rlocus(sys)
11 xlim([-25 15])
12 ylim([-30 30])

```

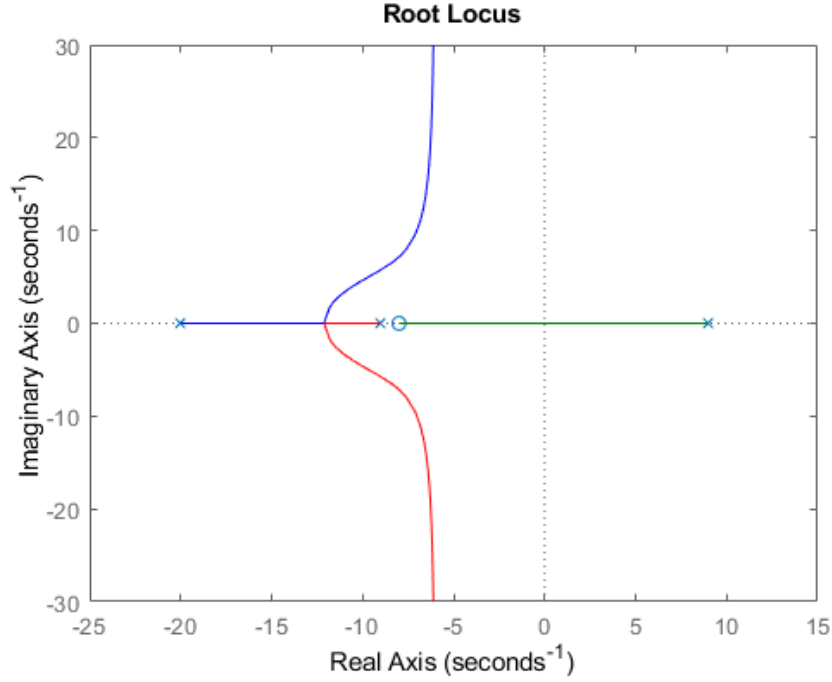


Figure 8: Root locus of system with negative velocity feedback controller.

### 3.3.2 Areas where damping ratio is higher than 0.6 + damped natural frequency is higher than $7 \text{ rad s}^{-1}$

We are given that our damping ratio must be at least 0.6 and that our damped natural frequency must be at least  $7 \text{ rad s}^{-1}$ :

$$\zeta \geq 0.6 \quad (3.3)$$

$$\omega_d \geq 7 \quad (3.4)$$

Solving to find the minimum undamped natural frequency:

$$\omega_n = \frac{\omega_d}{\sqrt{1 - \zeta^2}} \quad (3.5)$$

$$\omega_n = \frac{7}{\sqrt{1 - 0.6^2}} \quad (3.6)$$

$$\omega_n = 8.75 \quad (3.7)$$

Since the undamped and damped natural frequency are proportional, we know that:

$$\omega_n \geq 8.75 \quad (3.8)$$

Using the following MATLAB code, the regions which fulfil the above constraints were plotted in Figure 9.

```

1  clc
2  clear
3  close all
4
5  %define variables
6  k = 1;
7  %transfer function
8  sys = tf([k 8*k],[0.05 1 -4.05 -81]);
9  %plot root locus
10 figure1 = figure;
11 rlocus(sys)
12 xlim([-16 1])
13 ylim([-20 20])
14
15 %input zeta and wn, vectorise from minimum value to arbitrary value
16 zeta = linspace(0.6,0.9,4);
17 wd = 7;
18 wn = wd./sqrt(1 - zeta.^2);
19 sgrid(zeta,wn)

```

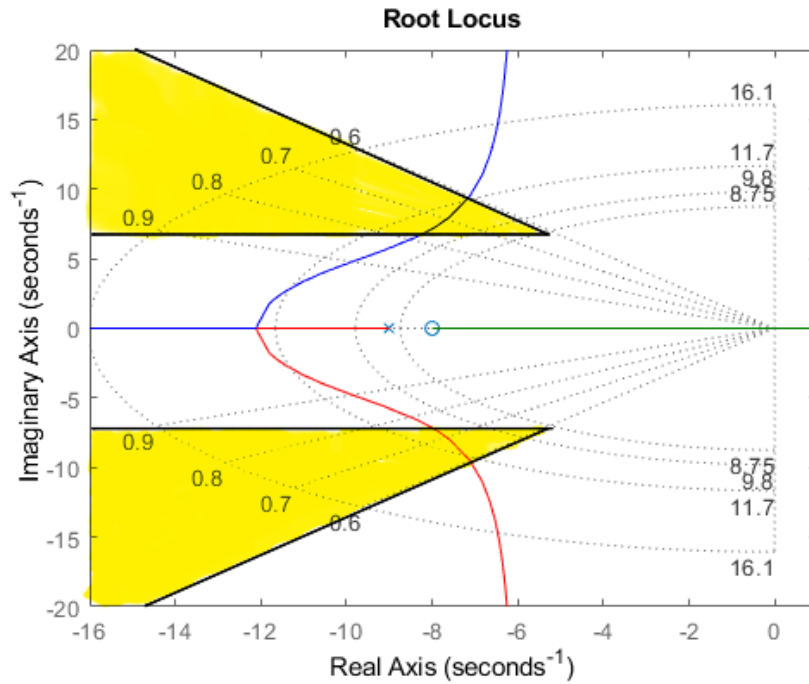


Figure 9: Regions on root locus diagram where  $\zeta \geq 0.6$  and  $\omega_d \geq 7$ .

### 3.3.3 Range of $k$ where performance specifications are achieved

By inspecting our MATLAB plots, we can find the gain values at any specified point.

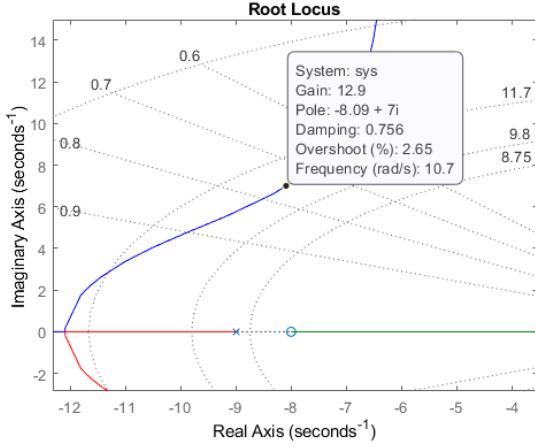


Figure 10: Minimum  $k$  value.

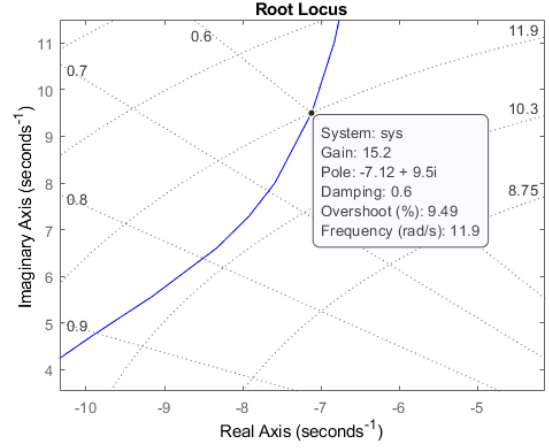


Figure 11: Maximum  $k$  value.

Here, we find that the maximum gain value that we can have is 15.2 at pole value  $-7.12 + 9.5i$ , limited by our damping ratio. For gains below 15.2, our pole position moves towards the real axis and away from the imaginary axis. The minimum gain value we can have is limited by the damped natural frequency and at a pole value of  $-8.09 + 7i$ , our gain is 12.9. Therefore:

$$12.9 \leq k \leq 15.2 \quad (3.9)$$

## 4 Controller techniques

### 4.1 Description of closed loop position control system

The system I have chosen to describe is a drone autopilot system. The drone I have picked as part of this analysis is a DJI Mavic 3. This drone is used to take high quality aerial video footage, hence stability and control are important for this particular drone. In a simple case, let us try and make our drone hover in a fixed position above the ground (in a fixed reference frame). Our desired output response is to keep the drone's coordinates fixed. To further simplify our example, let us only consider the motion in the  $z$ -axis (altitude).

The actuator is the motors attached to the drone propellers. The physical position to be controlled is the altitude of the drone. According to the drone specifications, the drone has a hovering accuracy range of  $\pm 0.1$  m in the vertical (with Vision Positioning). Using GPS data, we have a lower hovering accuracy at  $\pm 0.5$  m (?). As this is a video recording drone, the ability to stay at a fixed altitude and to not 'drift' is important to prevent the footage from looking shaky/blurry/distracting.

### 4.2 Specify disturbances

As the ground which the drone flies above is rough, we will find that there are perturbations, peaks and valleys in the terrain. In order to determine the distance to the ground, we have to measure the instantaneous vertical distance from the drone to the ground.

Some disturbances our drone will experience is wind, changing its position in the air. This will manifest as noise as the position of the drone will always be changing. Control is needed to keep the drone in a fixed position.

The DJI Mavic 3 uses GPS to determine its distance from the ground (?). If the drone is using GPS to determine its distance from the ground, there will be measurement noise. This will come from the signal power varying (the signal gain of the radio signals used to communicate with GPS satellites), the method of analog-to-digital conversion will produce signal noise, and the design of the antennas will also produce signal noise. The position data from the GPS receivers then needs to be processed and denoised, in order for an accurate measurement to be made. The position data then needs to be referenced against pre-existing terrain data. This terrain data should have a resolution high enough to accurately determine the distance to the ground over any point, but also the data should be processed to be ‘smooth’. Details such as rocks/small boulders or trees do not necessarily have to be in the terrain data, as the drone also uses Vision Positioning to determine the distance to smaller objects at closer distances (?).

## 5 Inverted pendulum

### 5.1 Identification and description of assumptions and limitations for the model of a rotary inverted pendulum

For the inverted pendulum system, we are considering the angle to be small. Hence, we can use the small angle approximation:

$$\sin \theta \approx \theta \quad (5.1)$$

Next, we have constrained our pendulum to only move in a 2D-plane ( $x$ - and  $z$ - axes), simplifying our model. We can make further simplifications by neglecting frictional forces and drag acting on the pendulum during movement.

### 5.2 Moving rocket to final position

There are a few steps to steering the rocket to minimise the  $x$ -position.

1. The rocket is vertical but a distance  $x$  from the centre of the base.
2. Let us first gimbal our engine to produce a thrust which will move the base of the rocket to the right. This will happen if we gimbal the engine to the left.
3. Now the centre of mass of the rocket is to the left of the long axis and it will start to move.
4. Now we can gimbal our engine in-line with the long axis of the rocket, and our rocket will start to translate left.
5. When we have reached the  $x = 0$  position, we can gimbal our engine to the right, which will move the centre of mass back in line with long axis of the rocket and the engine, straightening out our rocket.
6. Rocket is stable and stationary above the pad.



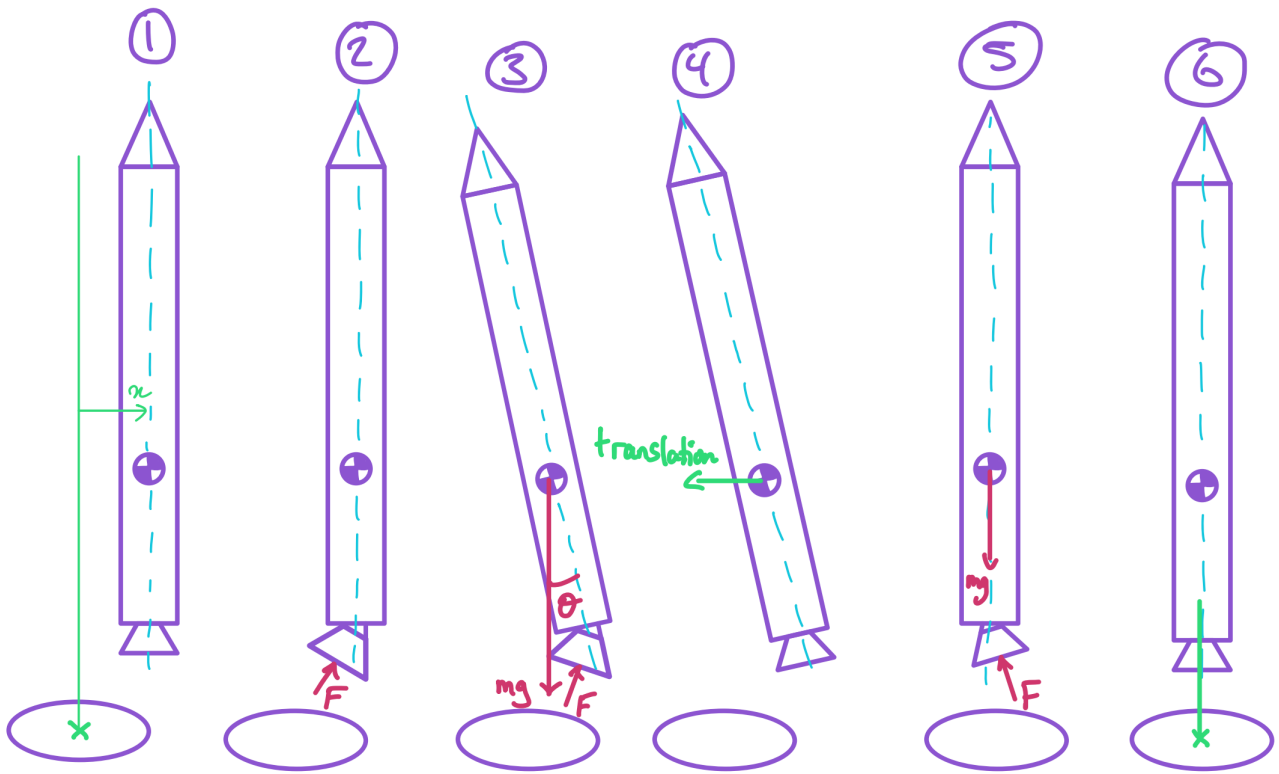


Figure 12: Stages of rocket aligning with base pad.

## 6 Multi-Degree-of-Freedom systems

### 6.1 Example of a system with multiple degrees-of-freedom

The system that I have chosen to describe is a hydraulic vehicle suspension system. This system's primary purpose is to reduce transmitted vibrations from the road surface to the car body and maximise tyre contact with the ground. It can achieve different characteristics by utilising oil or compressed air to change the stiffness and damping of the system. In one model, we can model the system as such:

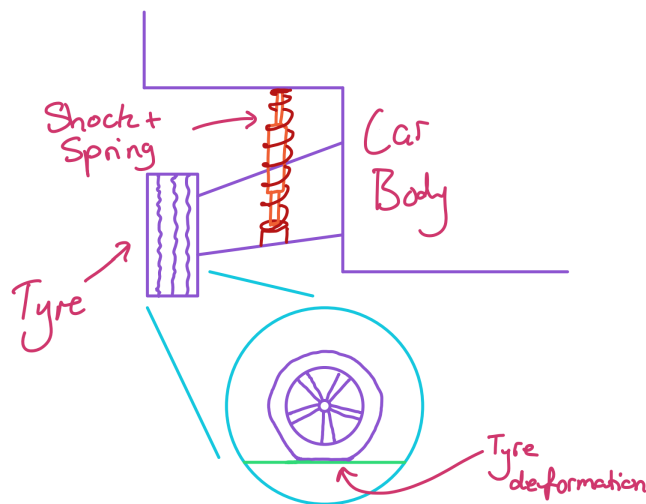


Figure 13: Physical model of suspension system.

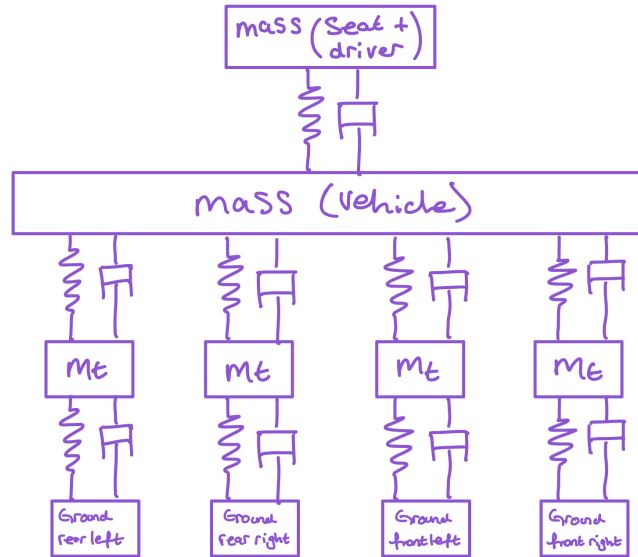


Figure 14: Model of MDOF system.

Our inputs are forces applied (and hence motion relative to the rest of the system) to the tyres from the ground. We know that tyres themselves have a certain stiffness and certain deformation characteristics, which can be modelled.

The output of our system is the overall displacement of the driver sitting in the car. We could model each individual component from the ground to the driver separately, but for simplicity's sake, the seat and driver have been modelled as one system (the seat is directly attached to the frame of the vehicle).

The forces transmitted from the ground can vary due to perturbations in the road surface, its smoothness and any obstacles in the path of the tyre. The ground may also not be homogeneous and each wheel may be receiving different displacements (for example on off-road terrain). Another external influence could be weather and damage to the components. Rainy/snowy weather would affect how the tyres interact with the ground, possibly preventing a good contact between tyre and ground. Damage to the components could be failed shocks/bushings (dampers) or corroded bolts. These would result in components being able to move in ways in which vibrations could be transmitted through the car in unwanted ways.

In vehicle suspension systems, the springs and shocks are not directly connected to the tyres, instead they are connected to the (lower) control arms. Similarly, the vehicle is a large object and the suspension struts are connected in the corners of the car, whereas the driver sits (roughly) in the middle of the vehicle. The flexure and stiffness of the car body may be negligible in this case, we may also consider the flexure in the control arms to be negligible as they are rigid components.

The main sensors would be a suspension spring displacement sensor and an accelerometer placed on the car body. This will give the onboard computer information about how the tyre is deforming and how the car body is reacting. There may be a supplementary sensor installed in the car seat to fine tune the response of the hydraulic suspension. Certain programmes can be installed on the computer which can tell the hydraulic suspension components to either stiffen or relax the damping within the shocks. This would have the effect of improving vehicle handling for a variety of scenarios. For example, for a high-performance scenario, a stiffer suspension system would allow for the tyres to have a better contact patch with the ground, allowing more power to be transmitted to the ground from the engine. On the other hand, a relaxed suspension system would smoothen out and reduce vibrations from the ground as experienced by the driver.

The controller to be used in this case would have to utilise some form of PID control. Proportional control would be used to control the vehicle vibrations (displacement). Integral control would be used to keep the vehicle stable after oscillatory action. This could be in the form of stabilising the ride height (distance between car body and ground) of the vehicle after a large displacement. Derivative control would be needed to prevent large overshoots, which would result in an overly ‘bouncy’ suspension response, with many oscillations after an initial displacement. Together, proportional, integral and derivative control would be necessary to stabilise the car body from the various forces exerted upon it by the road through the tyres.