



Semester project
Identification & control of a 3DOF hovering
system

Mechanical Engineering, EPFL STI SGM

Author: David KHATANASSIAN

Supervision: Prof. Alireza KARIMI, Vaibhav GUPTA

January 15, 2023

Contents

1	Introduction	3
2	Physical set-up	4
2.1	Actuators	4
2.2	Sensors	5
2.3	Connections to the micro-controller	5
2.4	Physical limitations	6
3	Modeling	7
3.1	Pitch and roll axis model	7
3.2	Yaw axis model	8
3.3	State-space model	8
3.4	Stability of the system	9
4	Stabilization	10
4.1	Decoupling	10
4.1.1	Pitch and roll axis SISO systems	10
4.1.2	Yaw axis SISO system	10
4.1.3	Decoupled SISOs	11
4.2	Parametric model from step responses	12
4.2.1	Pitch step response	12
4.2.2	Roll step response	13
4.2.3	Yaw step response	14
4.3	Stabilizing controllers	14
4.3.1	Structure	14
4.3.2	Choice of sampling period	16
4.3.3	Pitch stabilizing controller	17
4.3.4	Roll stabilizing controller	18
4.3.5	Yaw stabilizing controller	18
4.3.6	Validation	19
4.3.7	Practical aspects	19
5	System identification	21
5.1	Input signal	21
5.2	Fourier analysis with periodic signal	22
5.3	MIMO system identification	22
5.4	SISO systems identification	25
6	Robust controller design	28
6.1	H_∞ data-driven controller	28
6.1.1	Structure	28
6.1.2	Pitch and roll weighting filters design	28

6.1.3	Yaw weighting filters design	30
6.1.4	Optimization process and results	30
6.1.5	Validation	31
6.2	Q-parametrization	32
6.2.1	Validation	33
7	Conclusion	34

1 Introduction

This report outlines the identification and control procedure of the 3DOF Hover system manufactured by Quanser. It is structured in the way the project was conducted. It begins with the presentation of the different pieces of hardware and their electrical connections. Then, the system is modeled and its stability is analyzed. For reasons detailed later on, the original MIMO system is decoupled into 3 SISO systems, and initial stabilizing controllers are implemented on the latter. A frequency-domain closed-loop system identification is performed by means of PRBS signals. Both the decoupled and original plant are identified. As a last step, data-driven H_∞ and robust RST controllers are designed and implemented on the decoupled system.

This project also required the realization of a craft PCB, and a sizable amount of software design, which is not highlighted in this report.

2 Physical set-up

The 3DOF Hover system is composed of 4 actuators and 3 sensors. A myRIO-1900 device reads the outputs of the sensors and controls the actuators. The input commands need to go through a VoltPAQ-X4 power amplifier beforehand. A LabVIEW user interface is used to interact with the myRIO device.

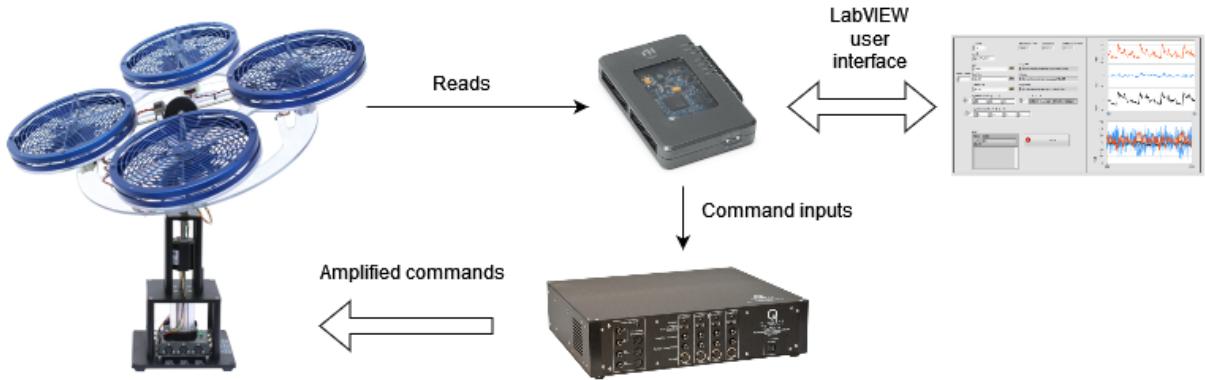


Figure 1: Set-up overview

2.1 Actuators

The actuators are AMETEK Pittman 9234S004-R1 DC motors. They have a nominal voltage of 12 VDC. The operating point is chosen to be $u_0 = 6$ VDC. The laboratory guide delivered by Quanser [1] specifies: *"due to their low resistance, switching between positive and negative voltage can cause permanent damage to the power amplifier"*. Hence, they will only rotate in the direction depicted on fig. 2:

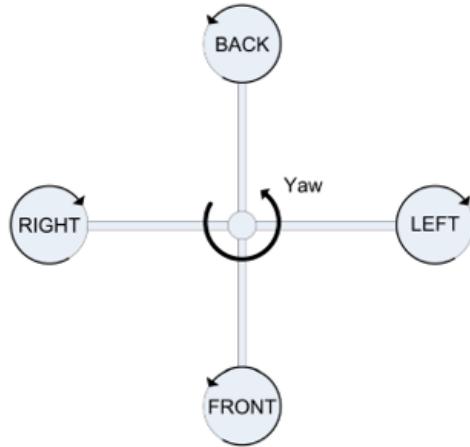


Figure 2: Top view of the system: direction of rotation

Propellers are mounted on the motors, allowing to generate thrust and torque. Around the operating point, the motor/propeller force-thrust constant is $K_f = 0.119$ N/V and the

motor/propeller torque-thrust constant is $K_t = 0.0036 \text{ Nm/V}$. The difference in thrust leads to a movement in pitch and roll, whereas a difference in torque leads to a movement in yaw. This justify the counter-rotating propellers, cancelling out each other's nominal torque. The motors are denoted as follows on the physical system: M_0 : Front, M_1 : Back, M_2 : Left, M_3 : Right.

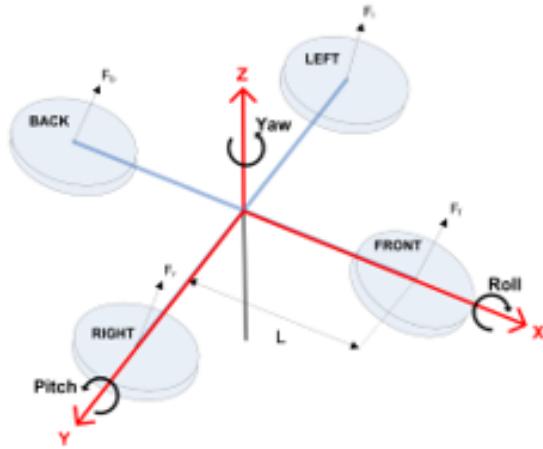


Figure 3: Simple free-body diagram of 3 DOF Hover

2.2 Sensors

The sensors are optical rotary quadrature encoders: E3-2048-250-NE-H-D-6 for pitch and roll, Orbex 512-1200 for yaw. One encoder consists of two channels, each of which has 2048 pulses per revolution (PPR). This means 8192 counts per revolution (CPR) in the resulting signal. Therefore, the resolution is of $\frac{360}{8192} \simeq 0.0439$ degree. As no index is present, they can only provide a relative position. For each axis, a counterclockwise (CCW) rotation leads to a positive increment. The encoders are denoted as follows on the physical system: E_0 : pitch, E_1 : roll, E_2 : yaw.

2.3 Connections to the micro-controller

To minimize the risk of troubleshooting necessity, a reliable connection solution is preferred. The ports A and B of the myRIO device are used with ribbon cables and IDC ports. The latter combined with a dedicated PCB offers a user-friendly plug-and-play solution. The PCB connects the pair of motors (M_0, M_1) and (M_3, M_2) to the device through the analogical output (2:AO0, 4:AO1) of port A and B, respectively. The channel A and B of the roll encoder are connected to the dedicated quadrature encoder input (18:ENC.A/22:ENC.B) of connector A. Analogously, the pitch encoder is connected to port B. The yaw encoder is connected to the digital inputs (13:DIO1/15:DIO2) of port A¹. The encoders are supplied by the device. The final set-up is shown in fig. 4.

¹ As these are not specifically dedicated to a quadrature encoder, a software adaptation is required. As temporary measure, a red/black to-be-removed pair of wire has been added, which permits to connect E_2 to the digital inputs (11:ENC0.A/13:ENC0.B) of port C, respectively.

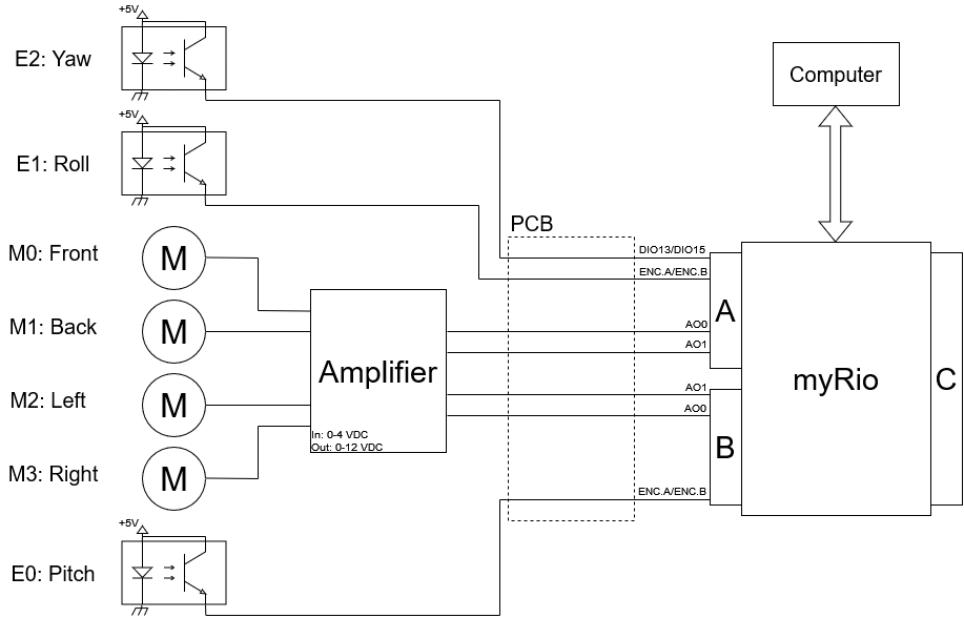


Figure 4: Set-up schematic

The PCB depicted on fig. 4 in dashed line is preseted here:

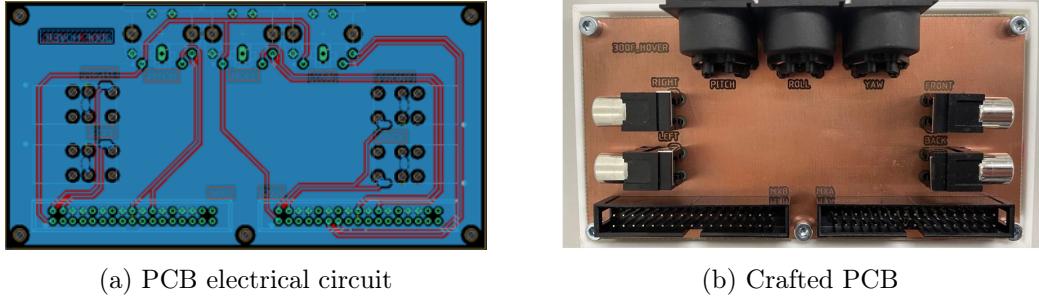


Figure 5: Final PCB

2.4 Physical limitations

- The motors can accept a nominal voltage of 12V. Even though they could accept peak voltages, it was chosen to avoid them as a first step. As the analogical output (2:AO0, 4:AO1) can deliver up to 5 VDC, which will be triple by the amplifier, the voltage output is limited to 4 VDC in `_sendCommand.vi`.
- The rotation for pitch and roll is limited within $\pm 37.5^\circ$. No software limitation is implemented, this is left for improvement.
- The rotation is unlimited for yaw.

3 Modeling

The initial system is a MIMO system. Its 4 inputs are the tensions applied to the motors: $u^T = [V_F, V_B, V_L, V_R]$, and the 3 outputs are the angle position of the platform: $y^T = [\theta_p, \theta_r, \theta_y]$. The reference position $y_0^T = \mathbf{0}_{[1 \times 3]}$ corresponds to the position of the platform before the beginning of an experiment with all motors off.

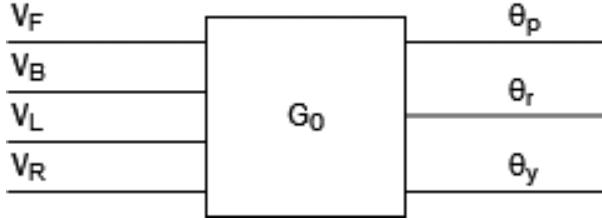


Figure 6: Plant's bloc diagram

3.1 Pitch and roll axis model

The behavior of the system about the pitch and roll axis can be modeled as a pendulum with friction:

$$\ddot{\theta}(t) + \frac{c}{m} \dot{\theta}(t) + \frac{g}{l} \theta(t) = \frac{1}{ml^2} M(t) \quad (1)$$

where m is the punctual mass of the system, l its distance to the center of rotation, c the friction coefficient and g the acceleration due to gravity. It is considered to be linear within the operating range of the platform, i.e. $\pm 37.5^\circ$.

According to fig. 3, the motors acting on the pitch axis are the front and back motors. If $V_F > V_B$, the system starts rotating CCW about the pitch axis. The torque M_p is given by the following equation:

$$M_p = K_f L \cdot (V_F - V_B) \quad (2)$$

Putting (1) and (2) together:

$$\ddot{\theta}_p(t) + \alpha_p \dot{\theta}_p(t) + \beta_p \theta_p(t) = \frac{K_f L}{J_p} (V_F - V_B) \quad (3)$$

where $J_p = ml^2$, $\alpha_p = \frac{c}{m}$, $\beta_p = \frac{g}{l}$. K_f is the thrust-force constant, and L is the distance from the center of rotation to the center of a propeller.

Similarly, for the roll axis, the system rotates CCW if $V_L > V_R$. The torque applied to the roll axis is:

$$M_r = K_f L \cdot (V_L - V_R) \quad (4)$$

Leading to:

$$\ddot{\theta}_r(t) + \alpha_r \dot{\theta}_r(t) + \beta_r \theta_r(t) = \frac{K_f L}{J_r} (V_L - V_R) \quad (5)$$

3.2 Yaw axis model

About the yaw axis, the system can be modeled as a rotating disc subjects to frictions:

$$\ddot{\theta}(t) + \alpha_y \dot{\theta}(t) = \frac{1}{J_y} M(t) \quad (6)$$

The system will start to rotate if the torque τ generated by the propellers is not balanced. As the front-back pair of motors rotates CCW and the left-right pair rotates CC, the system will rotate CCW if $\tau_{LR} > \tau_{FB}$:

$$M_y = K_t \cdot (V_L + V_R) - K_t \cdot (V_F + V_B) \quad (7)$$

where K_t is the torque-force constant. Hence:

$$\ddot{\theta}_y(t) + \alpha_y \dot{\theta}_y(t) = \frac{K_t}{J_y} (-V_F - V_B + V_L + V_R) \quad (8)$$

3.3 State-space model

The state space representation is given by:

$$\dot{x} = Ax + Bu$$

and

$$y = Cx + Du.$$

The state vector is defined as

$$x^T = [\theta_p \quad \dot{\theta}_p \quad \theta_r \quad \dot{\theta}_r \quad \theta_y \quad \dot{\theta}_y]$$

the output vector as

$$y^T = [\theta_p \quad \theta_r \quad \theta_y] \quad (9)$$

and the control vector as

$$u^T = [V_F \quad V_B \quad V_L \quad V_R]. \quad (10)$$

Using (3), (5), and (8), the state space matrices are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\beta_p & -\alpha_p & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\beta_r & -\alpha_r & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -\alpha_y \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{K_f L}{J_p} & -\frac{K_f L}{J_p} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{K_f L}{J_r} & -\frac{K_f L}{J_r} \\ 0 & 0 & 0 & 0 \\ -\frac{K_t}{J_y} & -\frac{K_t}{J_y} & \frac{K_t}{J_y} & \frac{K_t}{J_y} \end{bmatrix}$$

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

$$D = \begin{bmatrix} \mathbf{0} \end{bmatrix}_{4 \times 4}$$

3.4 Stability of the system

The matrix A is bloc-diagonal

$$A = \begin{bmatrix} A_p & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & A_r & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & A_y \end{bmatrix}$$

the eigenvalues of A_p and A_r are given by

$$\lambda_{1,2} = -\frac{\alpha}{2} \pm \frac{1}{2}(\alpha^2 - 4\beta)^{1/2}$$

We can see that $\text{Re}(\lambda_{1,2}) \in \text{LHP} \iff \alpha = \frac{c}{m} > 0, \beta = \frac{g}{l} > 0$. As $c > 0, m > 0, l > 0$, the system is stable about the pitch and roll axis. The eigenvalues of A_y are

$$\lambda_{y1,2} = \frac{1}{2}(\alpha_y \pm \alpha_y) = (0, -\alpha_y),$$

which means that the system is unstable about the yaw axis.

4 Stabilization

To perform system identification, the system must be stable. However, the latter is unstable about the yaw axis. The presence of a pole on the imaginary axis translates the presence of an integrator. In theory, to integrate a zero average input signal results in an zero output. However, unmodeled dynamics cause the system to drift. The yaw axis must be stabilized. The stabilization of the pitch and roll axis is not required to perform system identification, but could be appreciated as the damping of the roll axis is close to 0. Therefore, some unmodeled coupling dynamics could excite resonance frequencies, leading the system to get out of its linear behavior, or worse. To overcome this issue, the input signal amplitude should be reduced, which might imply a poorer quality of the identification. In any case, the computation of a datadriven controller requires a stabilizing controller.

For those reasons, a stabilizing controller is computed for each axis.

4.1 Decoupling

From the initial MIMO system, 3 SISO systems are obtained. This permits an easier design of the stabilizing controllers.

4.1.1 Pitch and roll axis SISO systems

For the pitch axis, we introduce $k_p := \frac{K_f L}{J_p}$ in (3) and $\Delta V_p := (V_F - V_B)$

$$\ddot{\theta}_p(t) + \alpha_p \dot{\theta}_p(t) + \beta_p \theta_p(t) = k_p(V_F - V_B) = k_p \Delta V_p$$

taking the Laplace transform , the following transfer function (tf) is obtained

$$\frac{\theta_p(s)}{\Delta V_p(s)} = \frac{k_p}{s^2 + \alpha_p s + \beta_p} \quad (11)$$

Similarly, the tf between θ_r and $\Delta V_r := (V_L - V_R)$ is given by

$$\frac{\theta_r(s)}{\Delta V_r(s)} = \frac{k_r}{s^2 + \alpha_r s + \beta_r} \quad (12)$$

4.1.2 Yaw axis SISO system

For the yaw axis, we introduce $k_y := \frac{K_y}{J_y}$ in (8)

$$\frac{1}{k_y} \ddot{\theta}_y(t) + \frac{\alpha_y}{k_y} \dot{\theta}_y(t) = (-V_F - V_B + V_L + V_R) = (-V_{FB} + V_{LR}) = \Delta V_y$$

leading to the following tf

$$\frac{\theta_y(s)}{\Delta V_y(s)} = \frac{k_y}{s^2 + \alpha_y s} = \frac{k_y}{s(s + \alpha_y)} \quad (13)$$

4.1.3 Decoupled SISOs

Assuming that the coupling terms are very small in comparison to (11), (12), and (13), the open-loop tf of the system is given by:

$$y = G\hat{u}$$

with the previously defined output vector y , the pseudo control input vector

$$\hat{u}^T = [\Delta V_p \quad \Delta V_r \quad \Delta V_y] \quad (14)$$

and the decoupled system matrix

$$G = \begin{bmatrix} G_p & 0 & 0 \\ 0 & G_r & 0 \\ 0 & 0 & G_y \end{bmatrix} = \begin{bmatrix} \frac{k_p}{s^2 + \alpha_p s + \beta_p} & 0 & 0 \\ 0 & \frac{k_r}{s^2 + \alpha_r s + \beta_r} & 0 \\ 0 & 0 & \frac{k_y}{s(s + \alpha_y)} \end{bmatrix} \quad (15)$$

The vector \hat{u} corresponds to applying the decoupling matrix

$$D = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 1 & 1 \end{bmatrix} \quad (16)$$

to the control input vector u

$$\hat{u} = Du$$

However, the provided control input vector being \hat{u} , the true input to the system u is internally computed² (c.f. fig. 7):

$$u = D^{-1}\hat{u}$$

with the pseudo-inverse matrix

$$D^{-1} = \begin{bmatrix} 0.5 & 0 & -0.25 \\ -0.5 & 0 & -0.25 \\ 0 & 0.5 & 0.25 \\ 0 & -0.5 & 0.25 \end{bmatrix} \quad (17)$$

the system can be expressed as

$$y = G\hat{u} = G_0 D^{-1} \hat{u}$$

with the initial MIMO system G_0 and the decoupled system $G = G_0 D^{-1}$. The open-loop bloc diagram is shown below

² this is done in `convertInput.m` and `_convertInput.vi`

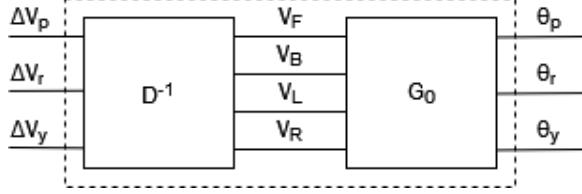


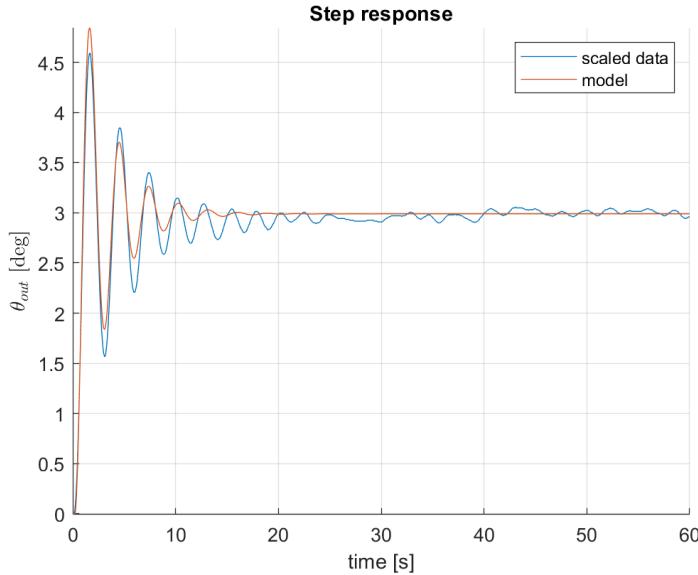
Figure 7: Decoupled plant's bloc diagram

4.2 Parametric model from step responses

Each system being simple enough, a first approximation is made by fitting a model to their step responses. As mentioned in section 3, the initial position of the platform is set at the beginning of an experiment when the motors are off. As their respective thrust and torque are not perfectly balanced, setting the motors to $u_0 = 6V$ causes a step disturbance in pitch and roll. The yaw axis does not start to move as a torque of at least $\Delta V_y = 4V$ is required to overcome the initial friction. A fast sampling period $T_s = 5ms$ is chosen for the open-loop (OL) acquisition. The step input is applied 25 seconds after motors start. This is the required time for the system to reach its new steady state.

4.2.1 Pitch step response

A step input of $\Delta V_p = 5V$ is applied, the scaled step response is shown bellow.

Figure 8: θ_p scaled step response, the step is applied after transient time of 25[s]

The step response is approximated with the following 2nd degree order model:

$$G_p(s) = \frac{k_p}{s^2 + \alpha_p s + \beta_p} \equiv \frac{\gamma \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

A delay $\tau = 0.2$ is observed, and hence added to the initial model. The parameters $\gamma = 2.99$, $\xi = 0.15$, and $w_n = 2.21$ provide an acceptable first approximation.

$$G_p(s) = e^{-\tau s} \frac{\gamma \omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} = e^{-0.2s} \frac{14.6}{s^2 + 0.66s + 4.88}$$

The extra oscillations are mostly due to the power wire that goes from the support to the platform, which is aligned with the pitch axis. It acts as a spring that disturbs the movement. Furthermore, when the power cable becomes compressed, the tension is partially released by actuating the roll axis. Hence, a cross-coupling is expected, but ignored as a first step.



Figure 9: Power cable

4.2.2 Roll step response

The same 2nd degree order model is used for the roll axis. Being less damped, a smaller step input $\Delta V_r = 2V$ is applied, the scaled step response is shown below.

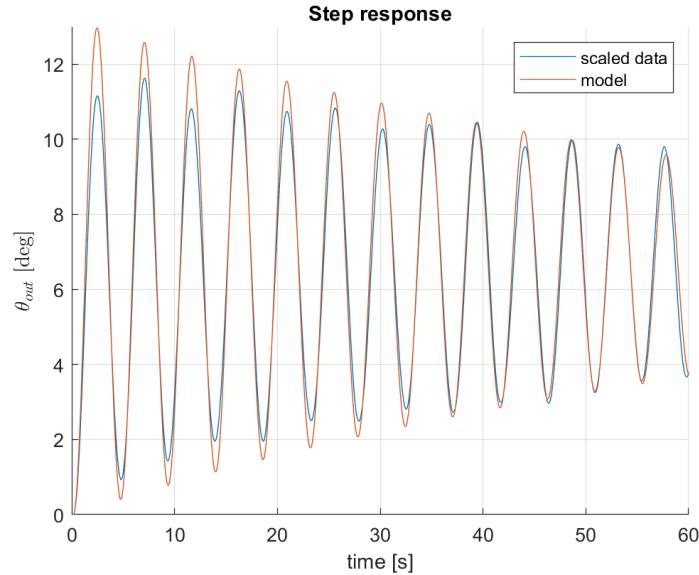


Figure 10: θ_r scaled step response, after transient time of 25[s]

A delay $\tau = 0.15$ is observed. The parameters $\gamma = 6.59$, $\xi = 0.01$, and $w_n = 1.36$ provide an acceptable first approximation.

$$G_r(s) = e^{-\tau s} \frac{\gamma \omega_n^2}{s^2 + 2\xi \omega_n s + \omega_n^2} = e^{-0.15s} \frac{12.22}{s^2 + 0.03s + 1.86}$$

4.2.3 Yaw step response

A step input of $\Delta V_y = 24V$ is applied. The derivative of the step response is used to identify the following 1st order model:

$$\frac{\dot{\theta}_y}{\Delta V_y} = \frac{k_y}{s + \alpha_y} \equiv \frac{\gamma}{\tau s + 1}$$

The parameters $\gamma = 17.6$, $\tau = 10.95$ provide an acceptable first approximation model.

$$G_y(s) = \frac{1}{s} \frac{\gamma}{\tau s + 1} = \frac{1}{s} \frac{17.6}{10.95s + 1}$$

The scaled step response is shown below. 0.63γ is depicted with a red dashed line.

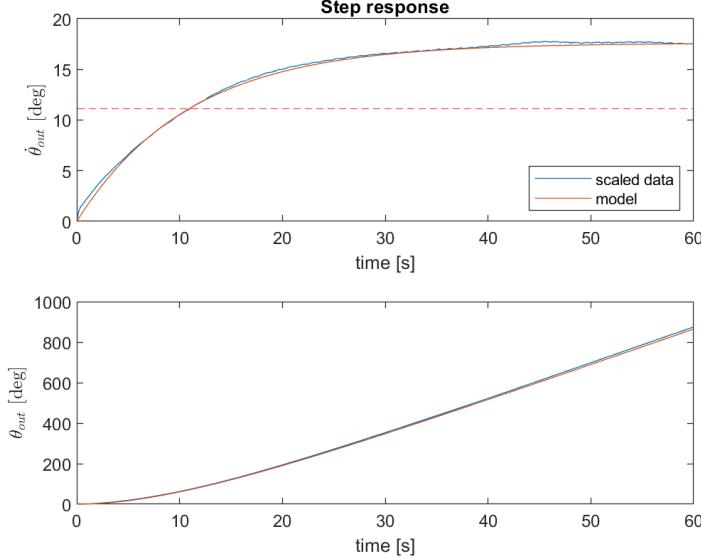


Figure 11: $\dot{\theta}_y$ and θ_y scaled step response, after transient time of 25[s]

4.3 Stabilizing controllers

4.3.1 Structure

The chosen structure is an RST structure. This choice is motivated by its ease of implementation and its general and more permissive structure [2].

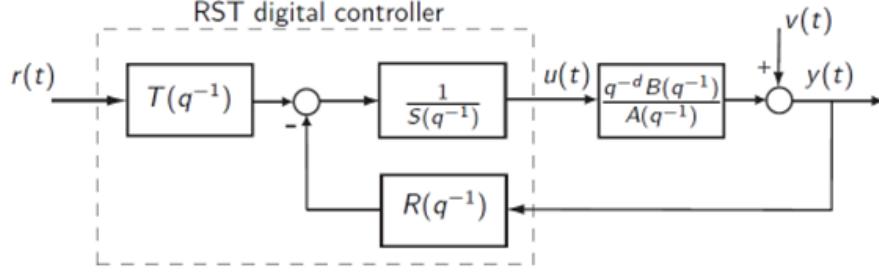


Figure 12: RST controller bloc diagram

Its two-degree of freedom form is given by:

$$T(q^{-1})r(t) = R(q^{-1})y(t) + S(q^{-1})u(t)$$

where q^{-1} is the backward shift operator and

$$R(q^{-1}) = r_0 + r_1q^{-1} + \dots + r_{n_R}q^{-n_R}$$

$$S(q^{-1}) = 1 + s_1q^{-1} + \dots + s_{n_S}q^{-n_S}$$

$$T(q^{-1}) = t_0 + t_1q^{-1} + \dots + t_{n_T}q^{-n_T}$$

In practice, the input is easily computed from

$$u(t) = -\sum_{i=0}^{n_R} r_i y(t-i) - \sum_{i=1}^{n_S} s_i u(t-i) + \sum_{i=0}^{n_T} t_i r(t-i) = \theta_{RST}^T \cdot \phi(t)$$

with the parameter vector

$$\theta_{RST}^T = [-r_0 \quad \dots \quad -r_{n_R} \quad -s_1 \quad \dots \quad -s_{n_S} \quad t_0 \quad \dots \quad t_{n_T}]$$

and the history vector

$$\phi^T(t) = [y(t) \quad \dots \quad y(t-n_R) \quad u(t-1) \quad \dots \quad u(t-n_S) \quad r(t) \quad \dots \quad r(t-n_T)].$$

Based on figure 12 the closed-loop (CL) tf between the reference signal $r(t)$ and the output $y(t)$ is :

$$Y_oR(q^{-1}) = \frac{q^{-d}T(q^{-1})B(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} \quad (18)$$

and the CL tf between $r(t)$ and $u(t)$ is

$$U_oR(q^{-1}) = \frac{q^{-d}T(q^{-1})A(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} \quad (19)$$

Similarly, the the CL tf between $v(t)$ and $y(t)$ is

$$YoV(q^{-1}) = \frac{q^{-d}S(q^{-1})A(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} \quad (20)$$

and the CL tf between $v(t)$ and $u(t)$ is

$$UoV(q^{-1}) = \frac{q^{-d}R(q^{-1})A(q^{-1})}{A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})} \quad (21)$$

Hence, the poles of the closed-loop system are the zeros of the polynomial $A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1})$ which can be placed using $R(q^{-1})$ and $S(q^{-1})$. $P(q^{-1})$ is defined as:

$$P(q^{-1}) = 1 + p_1q^{-1} + \dots + p_{n_P}q^{-n_P}$$

Thus, equation 22 can be solved to obtain $R(q^{-1})$ and $S(q^{-1})$ such that the closed-loop poles are the poles described by the polynomial $P(q^{-1})$.

$$A(q^{-1})S(q^{-1}) + q^{-d}B(q^{-1})R(q^{-1}) = P(q^{-1}) \quad (22)$$

4.3.2 Choice of sampling period

To obtain a discretized plant

$$G_d(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}$$

where

$$\begin{aligned} A(q^{-1}) &= 1 + a_1q^{-1} + \dots + a_{n_A}q^{-n_A} \\ B(q^{-1}) &= b_1q^{-1} + \dots + b_{n_B}q^{-n_B} \end{aligned}$$

a sampling period T_s has to be defined. It can be chosen from:

$$\frac{2\pi}{30\omega_b} < T_s < \frac{2\pi}{20\omega_b} \quad \text{or} \quad \frac{T_r}{10} < T_s < \frac{T_r}{5}$$

where ω_b is the bandwidth of the system and T_r its rise time [3]. For the pitch and roll, the bandwidth of G_p and G_r is used. For the yaw, the rise time of \dot{G}_y is used.

	$w_b[\text{rad/s}]$	rise time [s]	$T_s \text{ min [s]}$	$T_s \text{ max [s]}$
Pitch	3.38	—	0.06	0.09
Roll	2.12	—	0.10	0.15
Yaw	—	23.95	2.39	4.79

Table 1: Sampling period comparison

As all 3 systems are running at the same speed, $T_s = 50[\text{ms}]$ is chosen. This is fast enough to capture the dynamic of the fastest system.

4.3.3 Pitch stabilizing controller

As the main objective is a simple stabilizing controller, the requirements are i) to increase the damping, ii) to keep the input commands below saturation, and iii) to reject a step disturbance (not necessary, but preferable). The discretized model is

$$G_{pd}(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-4}(0.01803q^{-1} + 0.01783q^{-2})}{1 - 1.955q^{-1} + 0.9674q^{-2}}$$

$n_A = 2$, $n_B = 2$, $d = 4$. To reject the mentioned step disturbance an integrator $H_S(q^{-1}) = 1 - q^{-1}$ is introduced in $S(q^{-1})$. The loop is opened at nyquist frequency by adding $H_R(q^{-1}) = 1 + q^{-1}$ in $R(q^{-1})$:

$$R(q^{-1}) = H_R(q^{-1})R'(q^{-1}) \quad (23)$$

$$S(q^{-1}) = H_S(q^{-1})S'(q^{-1}) \quad (24)$$

$n_{H_S} = n_{H_R} = 1$. Injecting (23) and (24) in (22):

$$A(q^{-1})H_S(q^{-1})S'(q^{-1}) + q^{-d}B(q^{-1})H_R(q^{-1})R'(q^{-1}) = P(q^{-1})$$

by denoting $A'(q^{-1}) = A(q^{-1})H_S(q^{-1})$ and $B'(q^{-1}) = B(q^{-1})H_R(q^{-1})$ we get:

$$A'(q^{-1})S'(q^{-1}) + q^{-d}B'(q^{-1})R'(q^{-1}) = P(q^{-1}) \quad (25)$$

For the minimal order solution we should have:

$$n_{R'} = n_A + n_{H_S} - 1 = 2$$

$$n_{S'} = n_B + n_{H_R} + d - 1 = 6$$

$$n_P \leq n_A + n_{H_S} + n_B + n_{H_R} + d - 1 = 9$$

$P(q^{-1})$ can be written as

$$P(q^{-1}) = P_d(q^{-1})P_f(q^{-1})$$

where $P_d(q^{-1})$ are the dominant poles of the desired CL system H_d :

$$H_d(s) = \frac{\omega_{nd}^2}{s^2 + 2\xi_d\omega_{nd}s + \omega_{nd}^2}$$

To minimize the oscillations, the initial damping $\xi = 0.15$ is increased to $\xi_d = 0.71$. To minimize the use of input, the nominal frequency of the OL system is preserved: $\omega_{nd} = \omega_n = 2.21[\frac{rad}{s}]$.

The roots of

$$s^2 + 2\xi_d\omega_{nd}s + \omega_{nd}^2 = s^2 + 3.14s + 4.88$$

are $s_{1,2} = -1.5690 \pm 1.5561i$, leading to $z_{1,2} = e^{s_{1,2}T_s} = 0.9218 \pm 0.0719i$. Hence,

$$P_d(z) = (z - z_1)(z - z_2) = z^2 - 1.8435z + 0.8548$$

or

$$P_d(q^{-1}) = 1 - 1.8435q^{-1} + 0.8548q^{-2}$$

The polynomial $P_f(q^{-1})$ defines the auxiliary CL poles. Since $n_{P_d} = 2$ but $n_P \leq 9$, 7 auxiliary poles are added:

$$P_f(q^{-1}) = (1 - 0.4q^{-1})^7.$$

They are chosen to be faster than the dominant poles, and tuned to get better CL performances. By solving (25), we can find $R'(q^{-1})$, $S'(q^{-1})$. Injecting the latter in (23), (24) we obtain:

$$R(q^{-1}) = 0.71 - 0.673q^{-1} - 0.7056q^{-2} + 0.6774q^{-3}$$

$$S(q^{-1}) = 1 - 2.6881q^{-1} + 3.1529q^{-2} - 2.0620q^{-3} + 0.8155q^{-4} - 0.250q^{-5} + 0.0007q^{-6} - 0.0139q^{-7}$$

$T(q^{-1})$ is computed for the system to have the same performances in tracking and regulation:

$$T(q^{-1}) = R(1) = 0.0088.$$

4.3.4 Roll stabilizing controller

The controller is obtain by following the exact same procedure as for the pitch controller. An integrator is added, the loop is opened at nyquist frequency, $\omega_{nd} = 1.36$, $\xi_d = 0.71$, and auxiliary poles are added. The discretized model is

$$G_{rd}(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-3}(0.01526q^{-1} + 0.01526q^{-2})}{1 - 1.994q^{-1} + 0.9986q^{-2}}$$

The found coefficients are:

$$R(q^{-1}) = 1.3507 - 1.2885q^{-1} - 1.3473q^{-2} + 1.2919q^{-3}$$

$$S(q^{-1}) = 1 - 2.3094q^{-1} + 2.2724q^{-2} - 1.1895q^{-3} + 0.3373q^{-4} - 0.0948q^{-5} - 0.0160q^{-6}$$

$$T(q^{-1}) = R(1) = 0.0068.$$

4.3.5 Yaw stabilizing controller

The discretized model is

$$G_{yd}(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{0.002016q^{-1} + 0.002013q^{-2}}{1 - 1.995q^{-1} + 0.9954q^{-2}}$$

As $\dot{G}_y(s)$ is a 1st order system, the order n_{P_d} of the dominant polynomial of the CL is chosen to be 1. As the yaw axis dynamic is very slow, the dominant pole is chosen close to -1 to avoid saturating the output.

$$P_d(q^{-1}) = 1 - 0.95q^{-1}$$

4 auxiliary poles are added

$$P_f(q^{-1}) = (1 - 0.8q^{-1})^4$$

as well as fixed terms H_S and H_R . The found coefficients are:

$$R(q^{-1}) = 4.9364 - 4.5492q^{-1} - 4.9265q^{-2} + 4.5592q^{-3}$$

$$S(q^{-1}) = 1 - 2.1246q^{-1} + 1.5647q^{-2} - 0.4001q^{-3}$$

$$T(q^{-1}) = R(1) = 0.0199.$$

4.3.6 Validation

The controllers are implemented on the physical system. The pitch (and roll) actuation causes the system to move about the yaw axis. If the reference were set to $\theta_{pd} = \pm 30^\circ$, the pitch controller would send a control input of amplitude $\Delta V_p = 12V$, which leaves no margin to the yaw controller to compensate the disturbance. Therefore, the (step) reference signal amplitude is limited to $\pm 20^\circ$. The controllers are validated if the system can track this reference signal and reject some disturbance inputs. On the following figure, we can see that the requirements are met. Because of the mentioned power cable, the pitch output does not match perfectly the `lsim` output. The roll and yaw outputs are very similar to the simulated ones. The stabilizing controllers are validated.

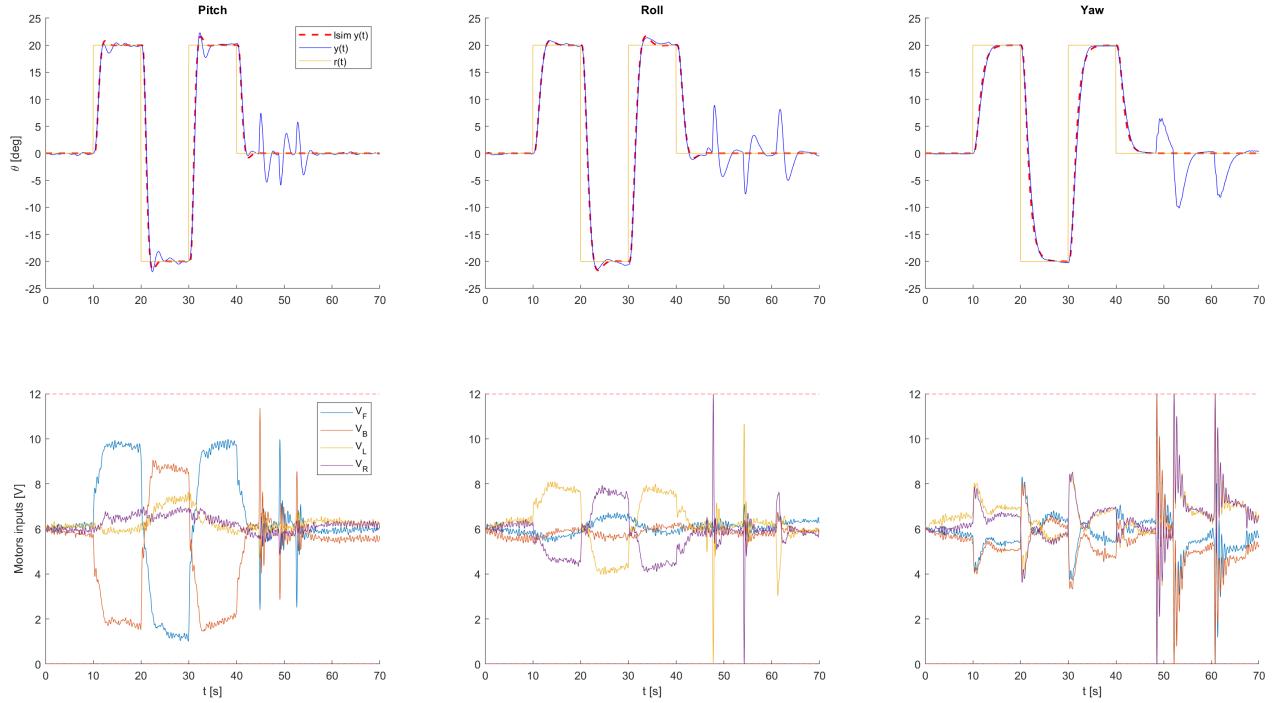


Figure 13: SISOs controllers: validation of tracking and regulation performances

4.3.7 Practical aspects

As the cumulative number of coefficient becomes rather important, a method to transfer the latter from Matlab to LabVIEW without having to write them one after the other is

implemented:

The function `labviewRST.m` takes as input 3 vectors and a mode parameter, and returns a single vector. If the mode '`solo`' is set, the 3 input vectors are the R,S,T coefficients of a single axis and the output vector is

$$v_{out} = [(n_R+1) \quad \{-r_0 \quad \cdots \quad -r_{n_R}\} \quad (n_S) \quad \{-s_1 \quad \cdots \quad -s_{n_S}\} \quad (n_T+1) \quad \{t_0 \quad \cdots \quad t_{n_T}\}]$$

If the mode '`trio`' is set, the 3 input vectors v_{out1} , v_{out2} and v_{out3} are the RST parameters of the pitch, roll and yaw axis, respectively. The output vector is

$$v_{out} = [(n_{v_{out1}}) \quad \{v_{out1}\} \quad (n_{v_{out2}}) \quad \{v_{out2}\} \quad (n_{v_{out3}}) \quad \{v_{out3}\}]$$

This allows to encode the coefficients in a binary file, which is then read by the sub-vi `unstackArray.vi` in LabVIEW.

5 System identification

The identification of the initial MIMO system G_0 and the decoupled system G is performed. A Frequency-Domain identification is conducted by means of a Fourier analysis. A PRBS signal is chosen as excitation signal. On fig. 14, G is represented as the dashed-box containing the initial system and the decoupling matrix D^{-1} .

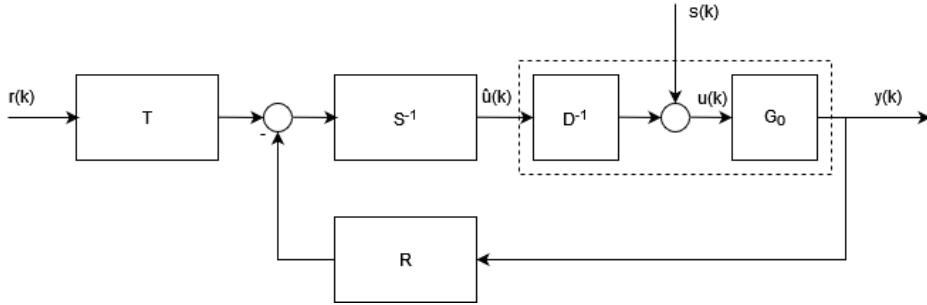


Figure 14: System in closed-loop

The blocs $R \in \mathbb{R}^{3 \times 3}$, $S^{-1} \in \mathbb{R}^{3 \times 3}$, $T \in \mathbb{R}^{3 \times 3}$ contains the tf of the stabilizing controllers on their diagonal. For example

$$R = \begin{bmatrix} R_p & 0 & 0 \\ 0 & R_r & 0 \\ 0 & 0 & R_y \end{bmatrix}$$

$r(k) \in \mathbb{R}^3$, denotes the desired trajectories

$$r(k) = [\theta_{pd}(k) \quad \theta_{rd}(k) \quad \theta_{yd}(k)]^T$$

$s(k) \in \mathbb{R}^4$, denotes the disturbances applied to the motors

$$s(k) = [\delta_{V_F}(k) \quad \delta_{V_B}(k) \quad \delta_{V_L}(k) \quad \delta_{V_R}(k)]^T$$

$y(k)$, $u(k)$, and $\hat{u}(k)$ are the vectors (9), (10), and (14), respectively.

5.1 Input signal

The chosen input signal is a pseudo random binary sequence (PRBS) signal. The Matlab file `labviewInOut.m` is used to easily design input signals and to send them to the myRio device. The PRBS signal is generated with the function `f.generatePRBS.m`, taking as input parameters: the length of shift register n , the number of period p , the frequency divider factor fD , the sampling period (in [s]) `Ts_sec`, and a '`info`' parameter. If the latter is set to 1, the function also returns information about the generated signal, such as its degree of excitation $M = fD(2^n - 1)$, the sampling frequency $\omega_s = \frac{2\pi}{Ts}$, the sampling

interval ω_s/N , and the duration of the experiment. It also plot the spectrum of the signal to remind that it is not uniform anymore when fD is not equal to 1.

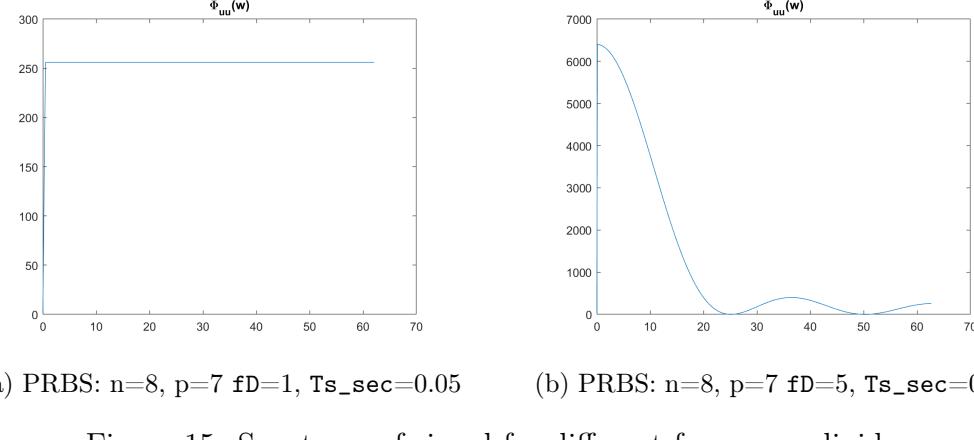


Figure 15: Spectrum of signal for different frequency divider

5.2 Fourier analysis with periodic signal

The coefficients of the Fourier series of a periodic signal $u(k)$ are computed by:

$$U(e^{j\omega_n}) = \frac{1}{M} \sum_{k=0}^{M-1} u(k) e^{-jkw_n T_s} \quad \omega_n = \frac{2\pi n}{MT_s} \quad n = 0, \dots, M-1$$

Similarly, the Fourier coefficients of the output of the system $y(k)$ are given by:

$$Y(e^{j\omega_n}) = \frac{1}{M} \sum_{k=L}^{L+M-1} u(k) e^{-jkw_n T_s} \quad \omega_n = \frac{2\pi n}{MT_s} \quad n = 0, \dots, M-1$$

where L is the settling time of the discrete system. If p periods are available, the measurement error can be reduced by averaging the Fourier coefficients over the periods. Note that in a closed-loop operation, an averaging over $U(e^{j\omega_n})$ is also necessary [3]:

$$G(e^{j\omega_n}) = \frac{\frac{1}{p} \sum_{i=1}^p Y(e^{j\omega_n})}{\frac{1}{p} \sum_{i=1}^p U(e^{j\omega_n})}$$

5.3 MIMO system identification

On the LabVIEW front panel, `mode` must be set to `FBLR_CL`. In this mode, the input log file is sent to $s(k)$ and $r(k) = \mathbf{0} \quad \forall k$. The output log file contains

$$[y(k) \quad s(k) \quad u(k) \quad \hat{u}(k)]$$

We define $K := RS^{-1}D^{-1}$. For notation simplicity, the signals $X(e^{j\omega_n})$ are denoted by x . We have:

$$u = -Ky + s \quad \text{and} \quad y = G_0 u$$

From $s(k)$ to $u(k)$:

$$\begin{aligned} u &= -KG_0u + s \\ u &= (I + KG_0)^{-1}s = \mathcal{T}_{us}s \end{aligned}$$

where $\mathcal{T}_{us} := (I + KG_0)^{-1}$ denotes the CL transfer function from $s(k)$ to $u(k)$:

$$\mathcal{T}_{us} = \begin{bmatrix} G_{0u_F\delta_F} & G_{0u_F\delta_B} & G_{0u_F\delta_L} & G_{0u_F\delta_R} \\ G_{0u_B\delta_F} & G_{0u_B\delta_B} & G_{0u_B\delta_L} & G_{0u_B\delta_R} \\ G_{0u_L\delta_F} & G_{0u_L\delta_B} & G_{0u_L\delta_L} & G_{0u_L\delta_R} \\ G_{0u_R\delta_F} & G_{0u_R\delta_B} & G_{0u_R\delta_L} & G_{0u_R\delta_R} \end{bmatrix}$$

Similarly, from $s(k)$ to $y(k)$:

$$\begin{aligned} y &= -G_0Ky + G_0s \\ y &= (I + G_0K)^{-1}G_0s = \mathcal{T}_{ys}s \end{aligned}$$

where $\mathcal{T}_{ys} := (I + G_0K)^{-1}G_0$ denotes the CL transfer function from $s(k)$ to (k) :

$$\mathcal{T}_{ys} = \begin{bmatrix} G_{0\theta_p\delta_F} & G_{0\theta_p\delta_B} & G_{0\theta_p\delta_L} & G_{0\theta_p\delta_R} \\ G_{0\theta_r\delta_F} & G_{0\theta_r\delta_B} & G_{0\theta_r\delta_L} & G_{0\theta_r\delta_R} \\ G_{0\theta_y\delta_F} & G_{0\theta_y\delta_B} & G_{0\theta_y\delta_L} & G_{0\theta_y\delta_R} \end{bmatrix}$$

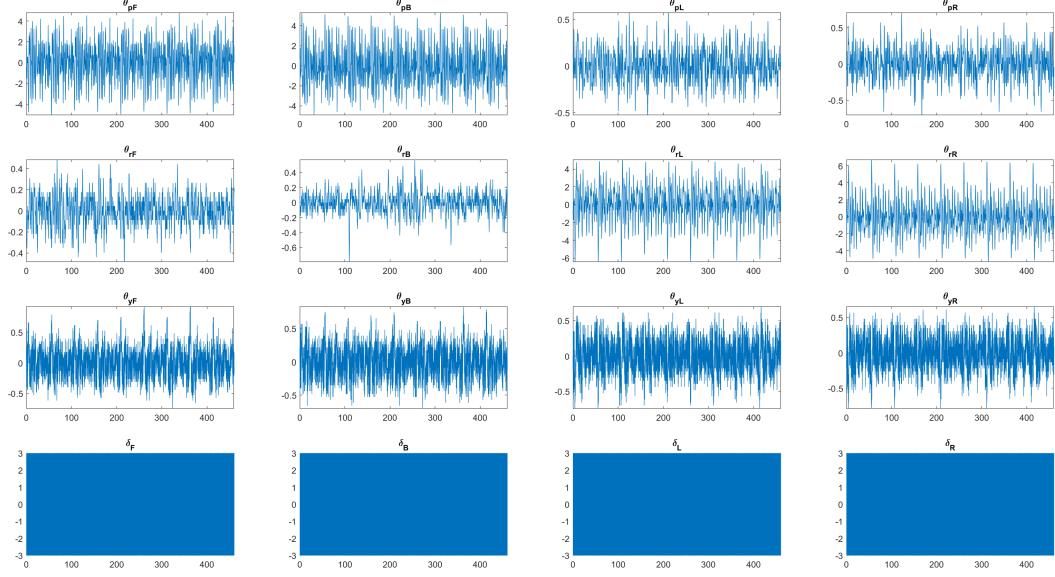
If \mathcal{T}_{us} and \mathcal{T}_{ys} are known, we can get G_0 from:

$$G_0 = \mathcal{T}_{ys}\mathcal{T}_{us}^{-1}$$

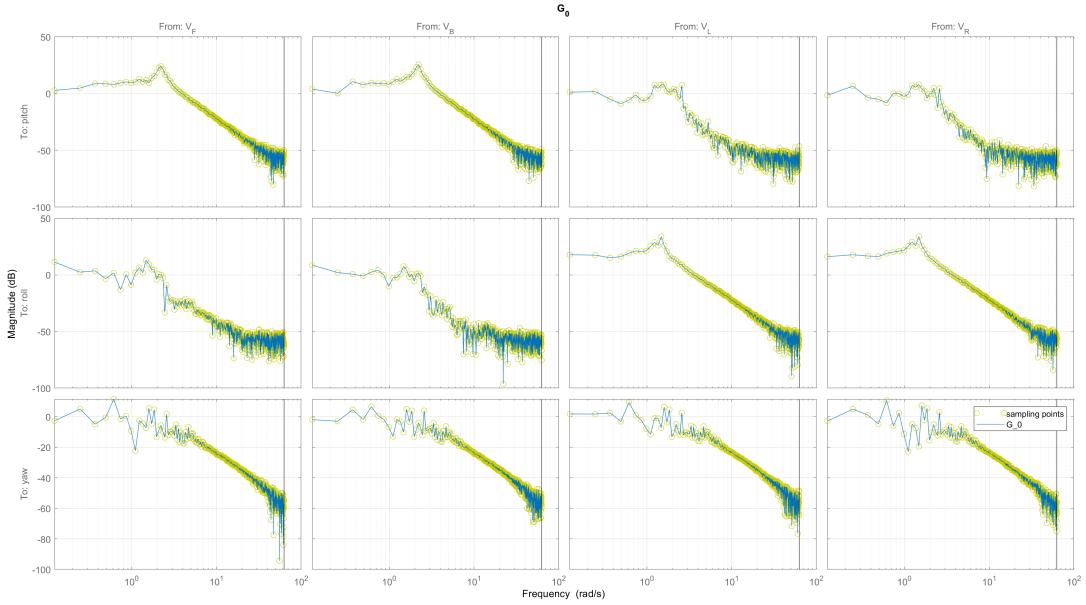
To obtain those matrices 4 experiments are conducted. One after the other, V_F , V_B , V_L , V_R are disturbed by a PRBS signal ($n = 10$, $p = 10$, $fD = 1$) of amplitude $a = 3V$. This is the maximal amplitude keeping $u(k) \in [0, 12]V$. The signal $s_i(k)$, $u_i(k)$, and $y_i(k)$, $i = 1, \dots, 4$ are recorded. Those experiments allow to identify \mathcal{T}_{ys} and \mathcal{T}_{us} . For example, experiment 1:

$$\begin{bmatrix} \theta_{1p} \\ \theta_{1r} \\ \theta_{1y} \end{bmatrix} = \begin{bmatrix} G_{0\theta_p\delta_F} & G_{0\theta_p\delta_B} & G_{0\theta_p\delta_L} & G_{0\theta_p\delta_R} \\ G_{0\theta_r\delta_F} & G_{0\theta_r\delta_B} & G_{0\theta_r\delta_L} & G_{0\theta_r\delta_R} \\ G_{0\theta_y\delta_F} & G_{0\theta_y\delta_B} & G_{0\theta_y\delta_L} & G_{0\theta_y\delta_R} \end{bmatrix} \begin{bmatrix} \delta_{V_F} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

allows to compute the 1st column of \mathcal{T}_{ys} . One period is composed of $M = 2^{10} - 1 = 1023$ samples, which takes $1023 \cdot 0.05 \simeq 50s$ to be executed. As the system is operating in closed-loop, $50 > T_s$ for any axis. Therefore, discarding the first period is sufficient. The recorded signals $y(k)$ and $s(k)$ for the remaining periods are shown on the next figure. The first column corresponds to experiment 1, and so on.

Figure 16: G_0 time-domain responses

Performing a Fourier analysis on the recorded data we obtain:

Figure 17: G_0 frequency-domain responses

As the design of a MIMO controller is left for a future project, the frequency responses are not further investigated.

5.4 SISO systems identification

The same procedure is applied to the decoupled system. On the LabVIEW front panel, `mode` must be set to `pry_CL`. In this mode, the input log file is sent to $r(k)$ and $s(k) = \mathbf{0} \quad \forall k$. The output log file contains

$$[y(k) \quad r(k) \quad u(k) \quad \hat{u}(k)]$$

We have

$$u = S^{-1}(Tr - Ry) \quad \text{and} \quad y = Gu$$

From $r(k)$ to $u(k)$:

$$\begin{aligned} u &= S^{-1}(Tr - RGu) \\ u &= (I + RG)^{-1}S^{-1}Tr = \mathcal{T}_{ur}r \end{aligned}$$

From $r(k)$ to $y(k)$:

$$\begin{aligned} y &= GS^{-1}(Tr - Ry) \\ y &= (I + GS^{-1}R)^{-1}GS^{-1}Tr = \mathcal{T}_{yr}r \end{aligned}$$

We can get G from:

$$G = \mathcal{T}_{yr}\mathcal{T}_{ur}^{-1}$$

As there is only 3 inputs, 3 experiments are required. The input signals are PRBS signal ($n = 10$, $p = 10$, $fD = 1$) of amplitude $a_p = 40^\circ$, $a_r = 50^\circ$, $a_y = 50^\circ$ for pitch, roll, and yaw, respectively. This is the maximum reference amplitude for the system to stay within $\pm 30^\circ$. The output responses to the inputs are shown on fig. 18, after having removed the 1st period.

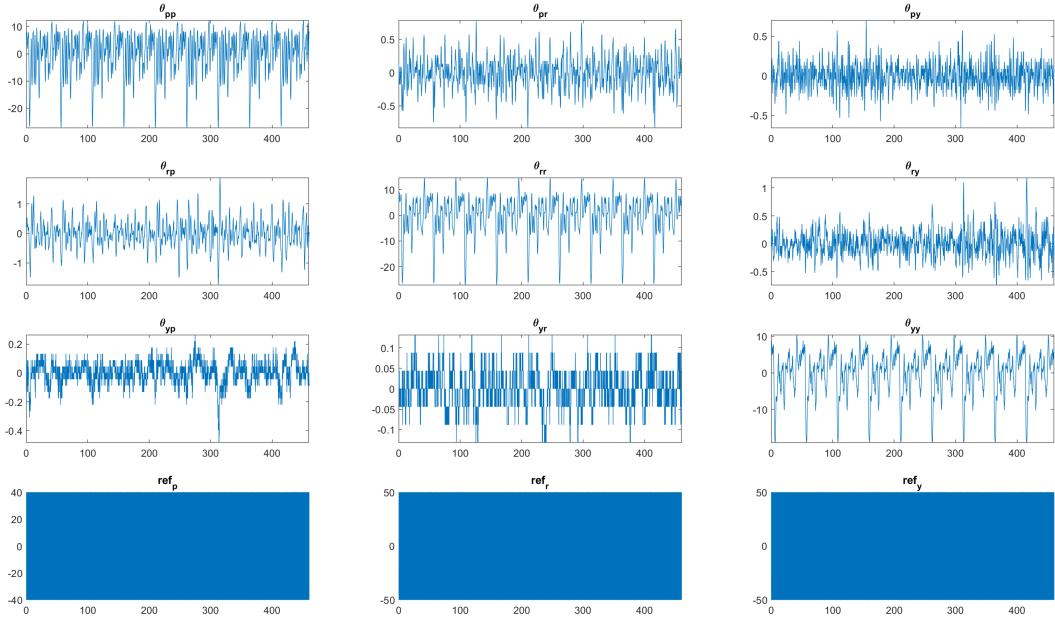


Figure 18: G time-domain responses

Performing a Fourier analysis on the collected data, we can obtain the frequency-domain response of the system:

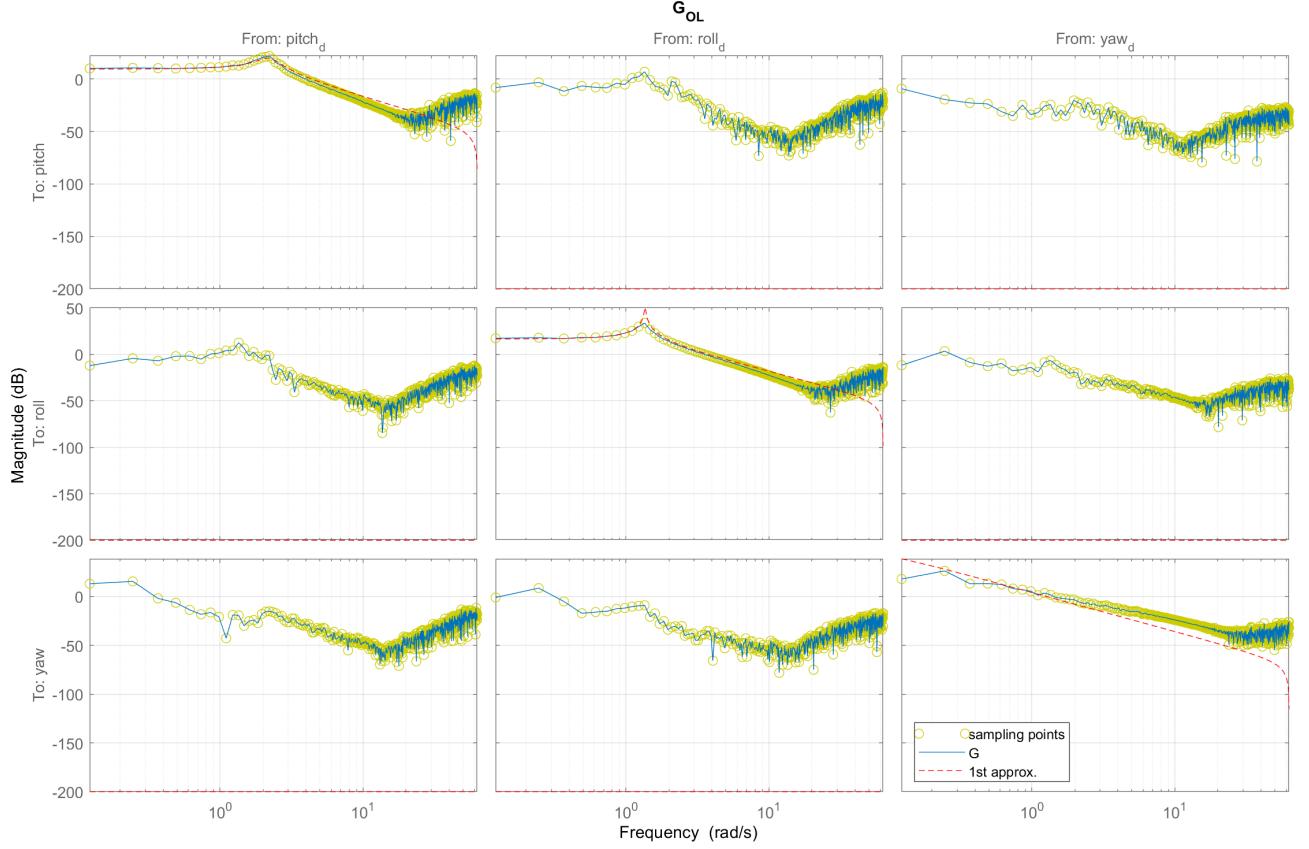
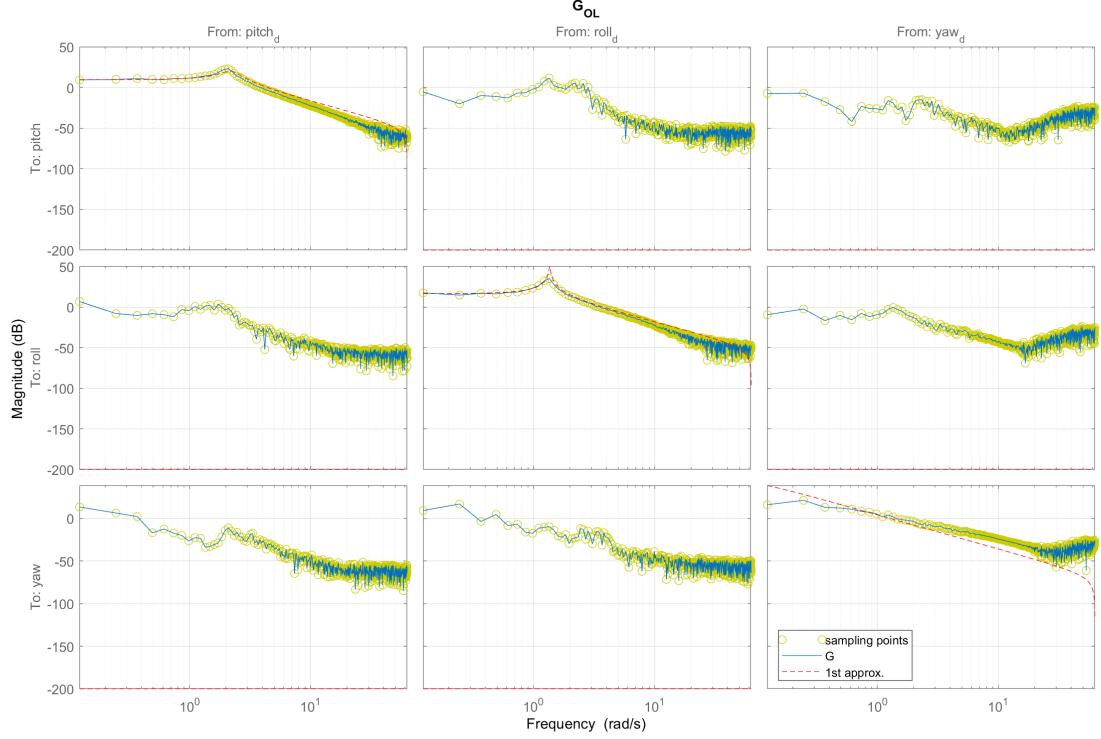


Figure 19: G frequency responses, $3 \times$ CL systems

The parametric transfer functions obtained by means of a step input (red dashed line) turn out to be rather accurate. We can see that the amplitudes of the responses start to increase again after $\simeq 15$ [$\frac{rad}{s}$]. This comes from the computation of G from the data collected in CL operating mode. Therefore, the system identification is reiterated with the same input signals but with the pitch and roll system in OL. As shown on fig. 20, the found frequency responses now fit even better the 1^{st} approximated systems. Based on those results, the use of a frequency divider is considered unnecessary and the system identification is validated.

Figure 20: G frequency responses, pitch/roll OL and yaw CL.

The coupling amplitudes are compared to the nominal ones on fig. 21. We can see that the signal to noise ratio (SNR) pitch/roll_dist is meaningful around the nominal frequency of the roll axis ω_{nr} (a). The roll axis decoupling is acceptable (b). The yaw axis is not well decoupled (c). This comes from the fact that, if the pitch (or roll) is required to move, one of two related motor will have a strong acceleration, which is equivalent to a yaw torque input.

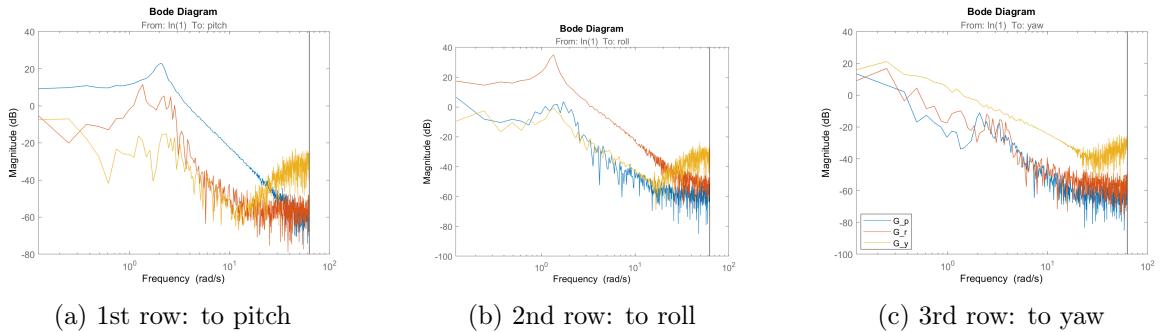


Figure 21: Signal vs. noise

6 Robust controller design

As said before, designing a MIMO controller is left for a future project. Instead, 3 distinct controllers are computed. We want a modulus margin $M_m \geq 0.5$. Furthermore, we would like the new CL operating system to be at least twice faster than the dynamic imposed by the stabilizing controller and to better match the simulated response.

6.1 H_∞ data-driven controller

The data-driven method comes in handy as it bypasses the need of an axis-wise parametric-identification procedure.

6.1.1 Structure

The stabilizing controllers were computed based on fig. 12. However, the data-driven controller $K = \frac{K_n}{K_d}$ computation is based on fig. 22a and is implemented following fig. 22b.

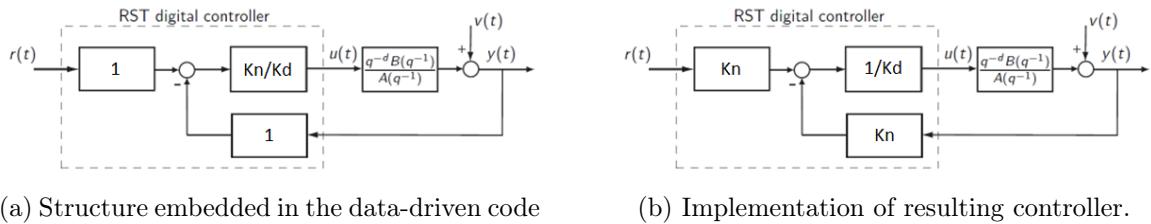


Figure 22: System representations

6.1.2 Pitch and roll weighting filters design

Based on fig. 21, the disturbance from yaw to pitch and from yaw to roll are considered negligible. However, the roll has an impact on the pitch and vice versa. For the pitch and roll axis systems together with the disturbances coming in from the two other axis can be modeled as follows:

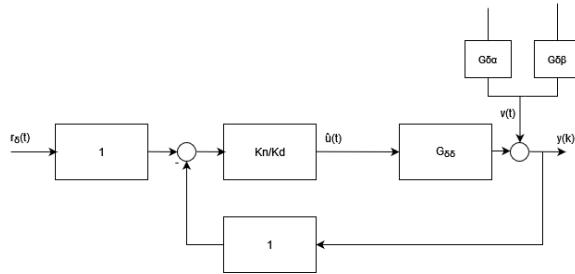


Figure 23: Single axis system modelling, with coupling disturbances

To diminish the impact of those disturbances, the filter W_1 is designed as follows:

W_1 weighting filter

The filter W_1 will be used to optimize over to following criterion:

$$\|W_1 \mathcal{S}\|_\infty < 1$$

where \mathcal{S} denotes the tf from $v(t)$ to $y(t)$. As we want a modulus margin of at least 0.5, this means that:

$$|1 + L| > 0.5, \quad \forall w$$

as $\mathcal{S} = (1 + L)^{-1}$

$$|(1 + L)^{-1}| > |W_1| \geq 0.5 = -6dB, \quad \forall w$$

To reject step perturbation an integrator $\frac{1}{s}$ is needed. The pole at zero is slightly moved by ϵ [4], and a coefficient β is added: $W_1 = \frac{1}{\beta s + \epsilon}$. To get the desired shape of the filter a zero has to be added: $W_1 = \frac{\alpha s + 1}{\beta s + \epsilon}$. We can write $W_1(jw) = \frac{\alpha jw + 1}{\beta jw + \epsilon}$. As we know that $W_1(\infty) \geq 0.5$, it means that $\frac{\alpha}{\beta} = 0.5$. Furthermore, to impose a bandwidth ω_b we set $\alpha = 1/\omega_b$. This leads to the filter:

$$W_1^{-1} = \frac{s + \epsilon}{M_m(s + \omega_b)} \quad (26)$$

From performance requirement, the bandwidth for the pitch is $\omega_{bp} \simeq 4.5[\frac{rad}{s}]$, and $\omega_{br} \simeq 2.5[\frac{rad}{s}]$ for the roll. To deal with the coupling effects, some Notch filters are added. Fig. 24b shows the initial coupling response from roll to pitch G_{pr} , the corresponding filter F_{pr} , and the resulting filtered response with peaks smoothed out.

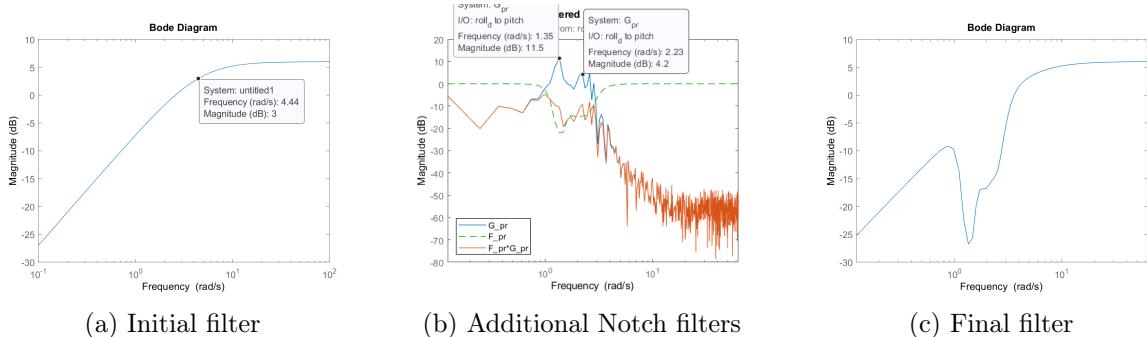


Figure 24: Pitch axis: design of W_1^{-1}

W_3 weighting filter

The filter W_3 will be used to optimize over to following criterion:

$$\|W_3 \mathcal{U}\|_\infty < 1$$

where \mathcal{U} denotes the tf from $v(t)$ to $u(t)$. A static gain is imposed.

$$\|\mathcal{U}\|_\infty < W_3^{-1} = 30dB$$

Some notch filters are further added in high frequencies, mainly to help the optimizer to converge to a solution satisfying the the criteria. This is shown in fig. 26.

6.1.3 Yaw weighting filters design

Based on fig. 21 $\frac{G_{yp}+G_{yr}}{G_{yy}} \simeq cst$. The disturbance can be modeled as follows:

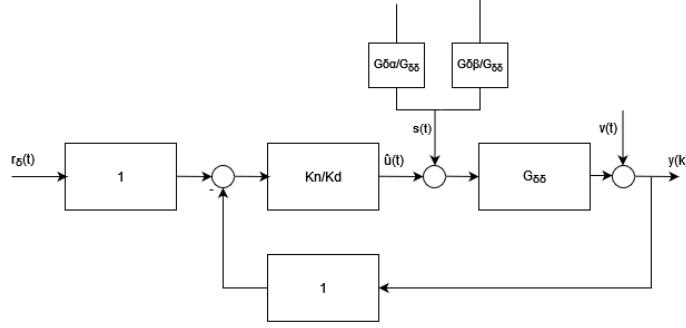


Figure 25: Yaw system with coupling disturbances modelling

To diminish the impact of those disturbances, the filter W_4 is designed as follows:

W_4 weighting filter

The filter W_4 will be used to optimize over the following criterion:

$$\|W_4\mathcal{V}\|_\infty < 1$$

where \mathcal{V} denotes the tf from $s(t)$ to $y(t)$. Since $\frac{G_{yp}+G_{yr}}{G_{yy}} \simeq cst$, a static gain is imposed.

$$\|\mathcal{V}\|_\infty < W_4^{-1} = -20dB$$

W_1 and W_3 weighting filters

The filter W_1 is designed according to (26) with $\omega_{by} = 2[\frac{rad}{s}]$. The filter $W_3^{-1} = 15dB$

6.1.4 Optimization process and results

For the pitch and roll axis, a sub-optimal controller will be computed with the following criterion:

$$\min \| [W_1\mathcal{S} \quad W_3\mathcal{U}] \|_\infty$$

For the yaw axis, a sub-optimal controller will be computed with the following criterion:

$$\min \| [W_1\mathcal{S} \quad W_3\mathcal{U} \quad W_4\mathcal{V}] \|_\infty$$

The degree of the controller is augmented as long as the optimizer does not output satisfying results, or if the results quality is not increasing anymore.

For the pitch axis, the obtained CL tf are displayed on fig. 26, with from top to bottom and from left to right: 11: \mathcal{S} , 12: \mathcal{U} , 21: \mathcal{V} , 22: \mathcal{T} . The CL tf of the stabilizing controller are displayed in dashed red lines, the imposed filters are in green.

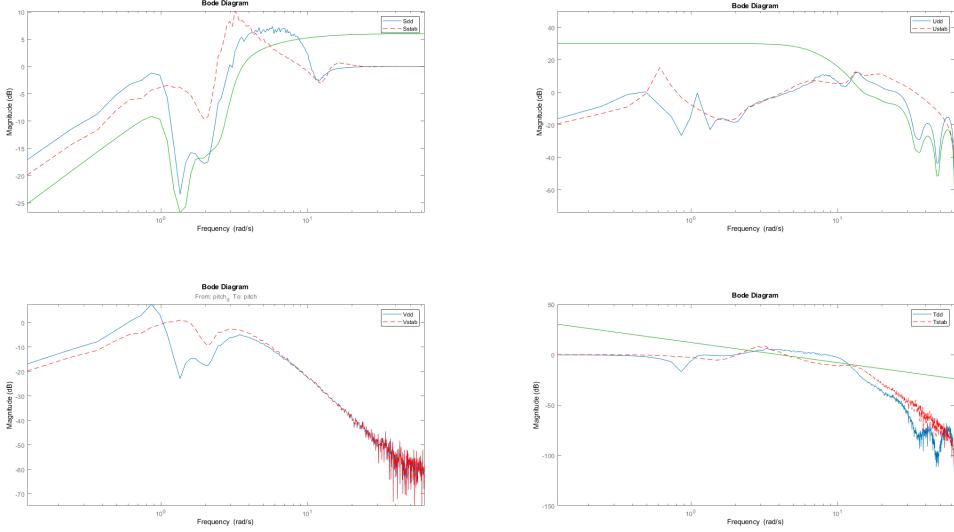


Figure 26: CL transfer function obtained from optimization process.

6.1.5 Validation

According to those results, a satisfying sub-optimal controller of degree 30 is found for the pitch axis. The found controller is implemented on the real system. We can see that i) regulation performances are present, but the system takes a lot of time to stabilize, and ii) the system cannot track a step of 20° of amplitude. A smaller step input of only 2° of amplitude is imposed, but is poorly followed. The debugging is yet to be performed. As long as the issue is not found, the optimization process for the roll and yaw axis is not ran.

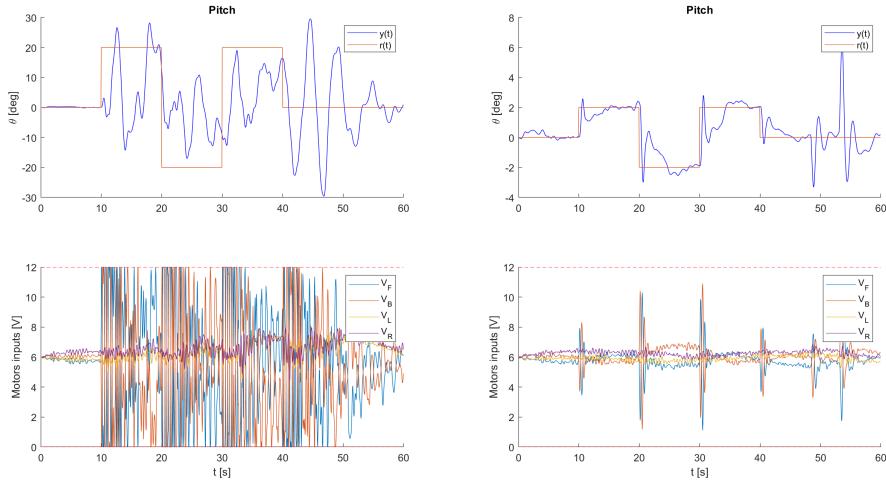


Figure 27: SISOs dd controllers: validation of tracking and regulation performances

6.2 Q-parametrization

As the implementation of the data-driven controller causes some issues, the identified parametric models are used to perform a robust pole-placement by Q-parametrization. In order to enhance the reactivity of the system, the sampling period is reduced to $T_s = 10[ms]$. We have:

$$R(q^{-1}) = R_0(q^{-1}) + A(q^{-1})Q(q^{-1}) \quad (27)$$

$$S(q^{-1}) = S_0(q^{-1}) - q^{-d}B(q^{-1})Q(q^{-1}) \quad (28)$$

with

$$Q(q^{-1}) = q_0 + q_1q^{-1} + \dots + q_{n_Q}q^{-n_Q}$$

The main advantage of the Q-parametrization is that while the desired closed-loop poles remain the same, the sensitivity functions depend linearly on Q as follows:

$$\mathcal{S}(q^{-1}, Q) = \frac{A(q^{-1})((S_0(q^{-1}) - B(q^{-1})Q(q^{-1}))}{P(q^{-1})} \quad (29)$$

$$\mathcal{U}(q^{-1}, Q) = \frac{A(q^{-1})((R_0(q^{-1}) + A(q^{-1})Q(q^{-1}))}{P(q^{-1})} \quad (30)$$

This linear dependency of the sensitivity functions on Q result in a convex optimization problem. The objective is the minimization of the two norm of $\mathcal{U}(Q)$ (also denoted as UoV in this report). Whereby two inequality constraints are applied. Namely that the infinity norm of $M_m\mathcal{S}(Q)$ is smaller than 1 and that the infinity norm of the input sensitivity function is smaller than a desired limit U_{max} .

$$\begin{aligned} & \min_Q \|\mathcal{U}(Q)\|_2 \\ \text{s.t. } & \|M_m\mathcal{S}(Q)\|_\infty < 1 \\ & \|\mathcal{U}(Q)\|_\infty < U_{max} \end{aligned}$$

The following table gather the parameters used to obtain R_0 and S_0 , as well as the parameters used to perform the optimization for each axis:

	Order of imposed CL system	ω_{nd}	ξ_d	nQ	M_m	Umax
Pitch	2	2×2.2	0.7	20	0.5	30
Roll	2	2×1.35	0.7	15	0.5	25
Yaw	2	2.2	0.7	30	0.5	30

On fig. 28, we can see that the sensitivity function \mathcal{S} does not exceed 6dB. The input sensitivity function goes slightly above the imposed limit by an acceptable amount. Hence, the optimization process is validated.

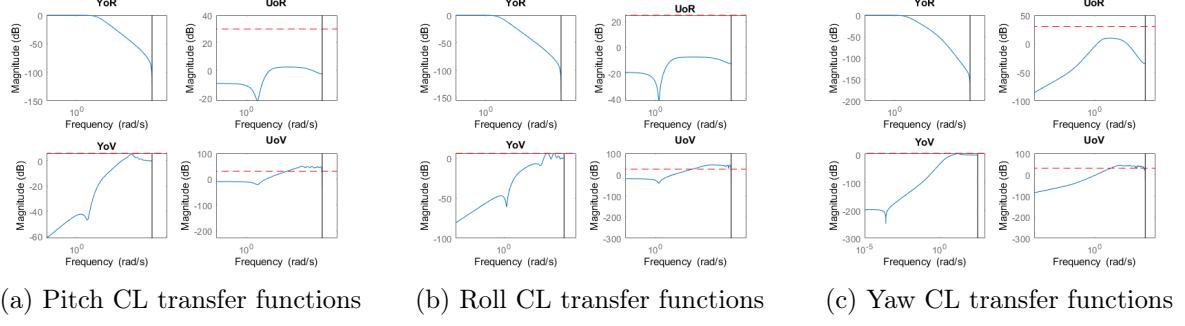


Figure 28: CL transfer function of system with Qp optimized controller.

6.2.1 Validation

The found controllers are implemented on the real system, and the performances are evaluated and compared to the output of `lsim`. The tracking performances are satisfying. Not only the residual oscillations have disappeared, but also the system is now faster, and match the simulated output. Furthermore, it is able to reject input disturbances. The controllers are validated.

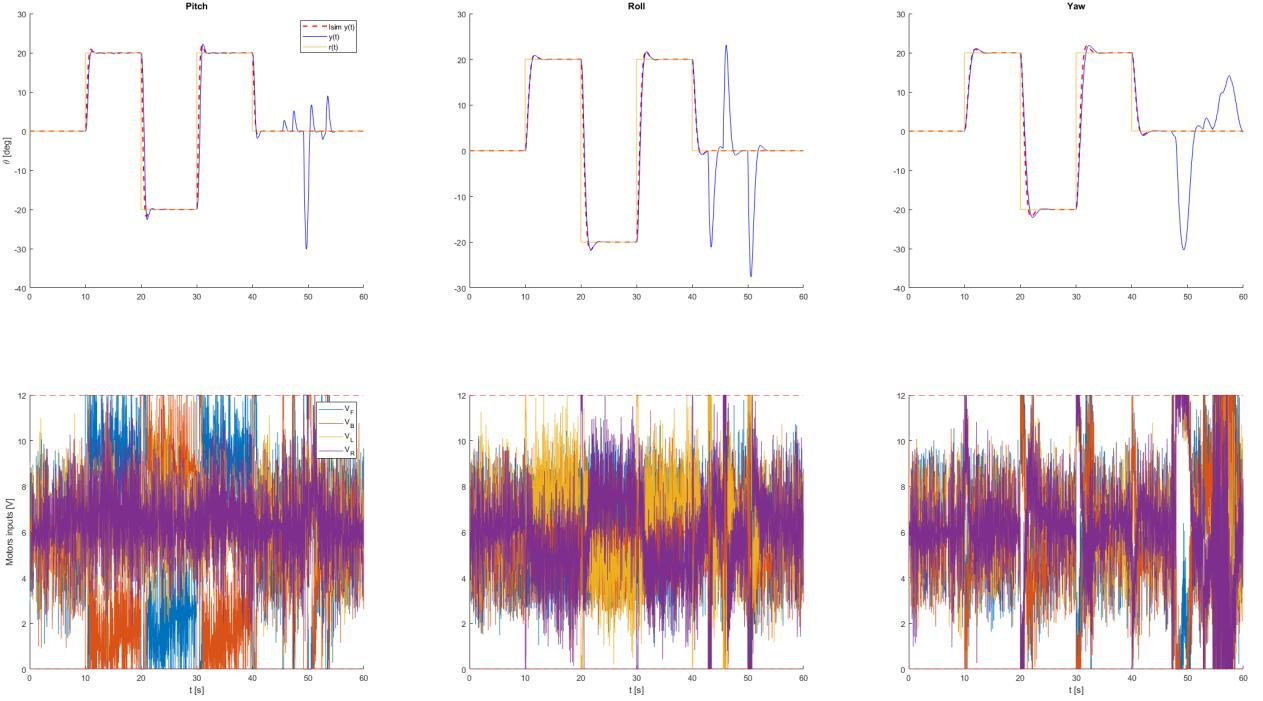


Figure 29: SISOs Qp controllers: validation of tracking and regulation performances

7 Conclusion

The objectives of this project are fulfilled. On one hand, the 3DOF system is properly set up and each component is connected to one another by means of a craft PCB, offering a reliable final installation. Furthermore, the implemented software allows the user not only to easily design input signals and controllers, but also to collect the data for further analysis. The available modes makes possible to operate the system as a MIMO system or as decoupled SISO systems, both in open-loop and closed-loop, testifying that the software was designed by taking into account the requirement of use of the system for further experiments. On the other hand, the objectives of stabilizing and identifying the system are achieved, and a set of robust controller was designed and implemented, which permits to operate the Hover.

As future work, the data-driven controller implementation could be debugged, and a MIMO controller could be designed.

References

- [1] <https://www.quanser.com/products/3-dof-hover/>. January 15, 2023
- [2] Pr. Alireza Karimi *Digital Control*. Fall 2020
- [3] Pr. Alireza Karimi *System identification*. Fall 2019
- [4] Pr. Alireza Karimi *Advanced control systems*. April 2021