POLITECNICO MILANO 1863

DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

M.Sc. AERONAUTICAL/SPACE ENGINEERING

051401 - DIGITAL CONTROL TECHNOLOGY FOR AERONAUTICS (DCTA)

Academic year 2025/2026

Prof. Fredy Ruiz

Assistant Alessandro Del Duca

Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

November 2025

Laboratory sessions

The experimental activity is based on the ANT-X platform, available at the Aerospace Systems & Control Laboratory.

It is divided into four sessions:

1. 1DOF modeling, November 20
2. Attitude Control, November 27
3. Cascade Control, December 4
4. Position Control, December 11

https://www.antx.it/

General Information

Activities are conducted at the ***Aerospace Systems and Control Laboratory*** of the DAER department. The ANT-X platform is employed for the experiments

The Laboratory is structured in four sessions:

1DOF Modeling: Using a single-degree-of-freedom configuration, students calibrate the drone's attitude dynamic model. Sensor types and the embedded digital system architecture are explored.

Speed Control: Students apply continuous-time PID controller design methods to manage the rotational speed of a drone using linearized dynamic models. Emphasis is placed on industrial controller architectures, saturation effects, and measurement noise.

Cascade Attitude Control: Using discretization techniques, students develop a velocity–position cascade controller for attitude regulation. The impact of bandwidth separation and discretization methods on system performance is analyzed.

Position Control: Students build discrete-time cascade control systems for 2DOF attitude-position regulation. Based on instructor-provided parameters, they design controllers' parameters and evaluate system performance.

ANT-X Platform

Software architecture

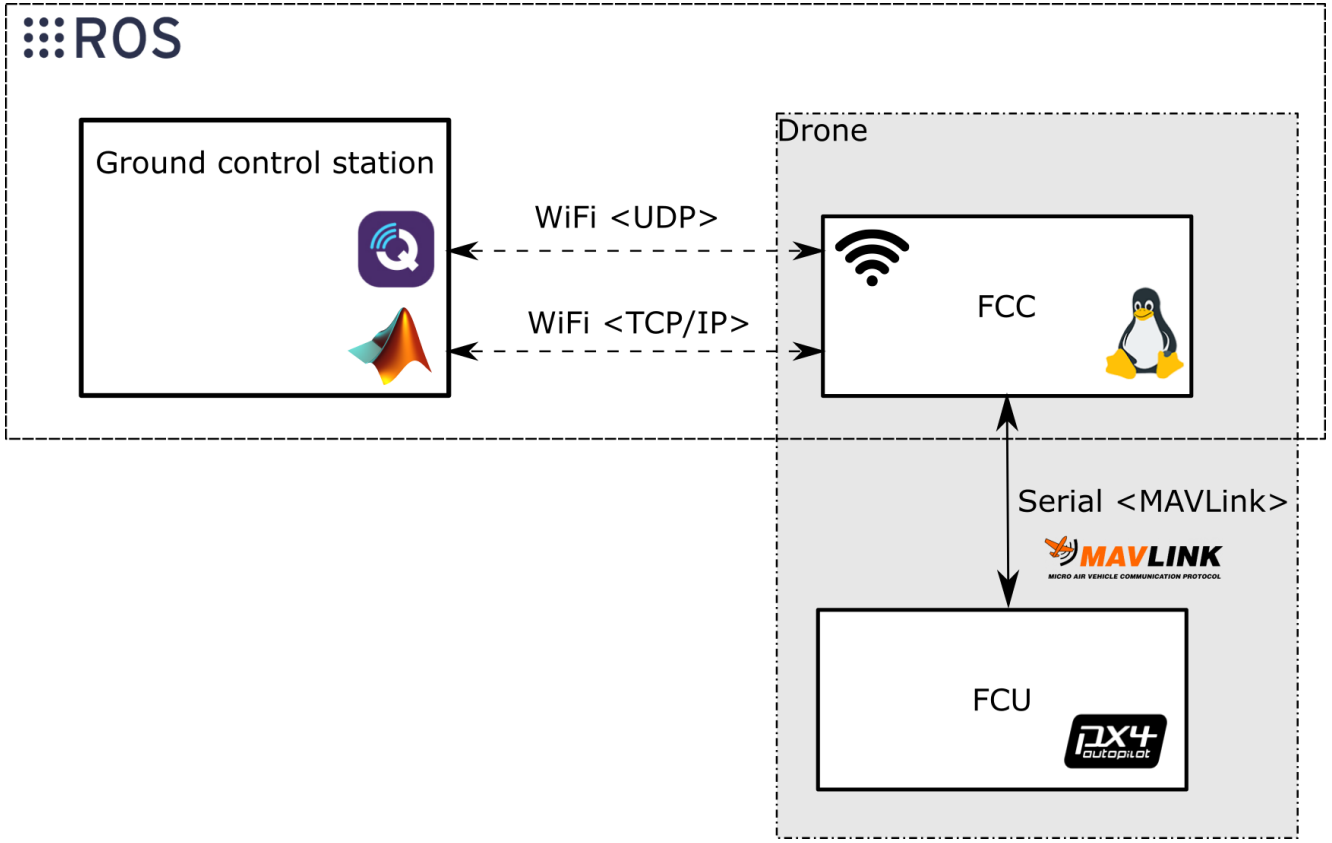
The ANT-X 2DoF Drone is an integrated hardware/software system

Integration of several hardware components and software packages:

- PX4 flight stack
- ROS
- MATLAB*
  - Simulink*
  - SLXtoPX4**
  - DroneCmd**
- QGroundControl

*third-party proprietary software

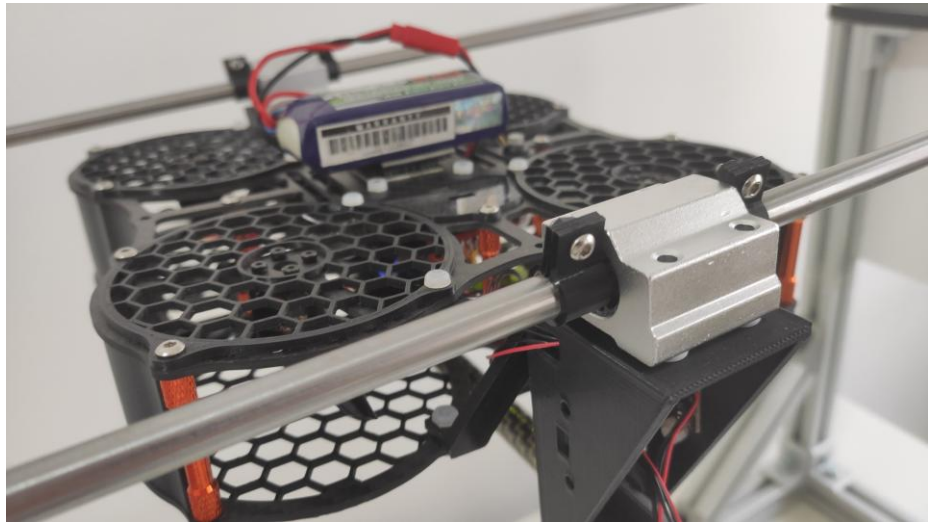
**ANT-X proprietary software tools



Part 1 - Familiarization with the platform ANT-X

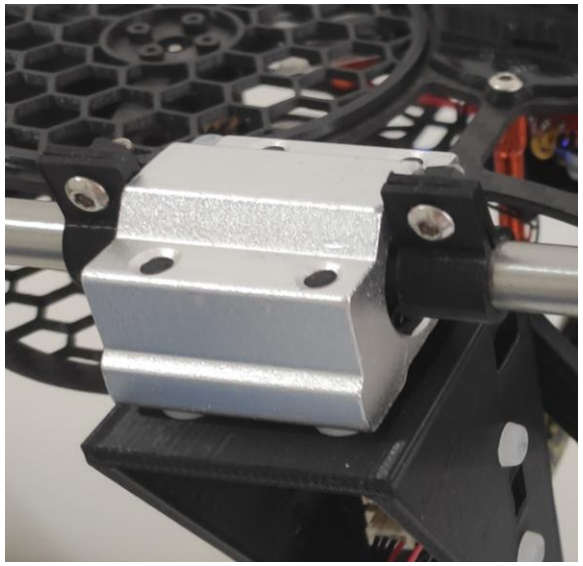
POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

The objective of this task is to characterize the performance of an existing attitude controller for the Drone in 1DoF configuration.



Preliminary setup:

1. Move the drone to the center of the rail
2. Fix the drone using the clamps present on the rail.



Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

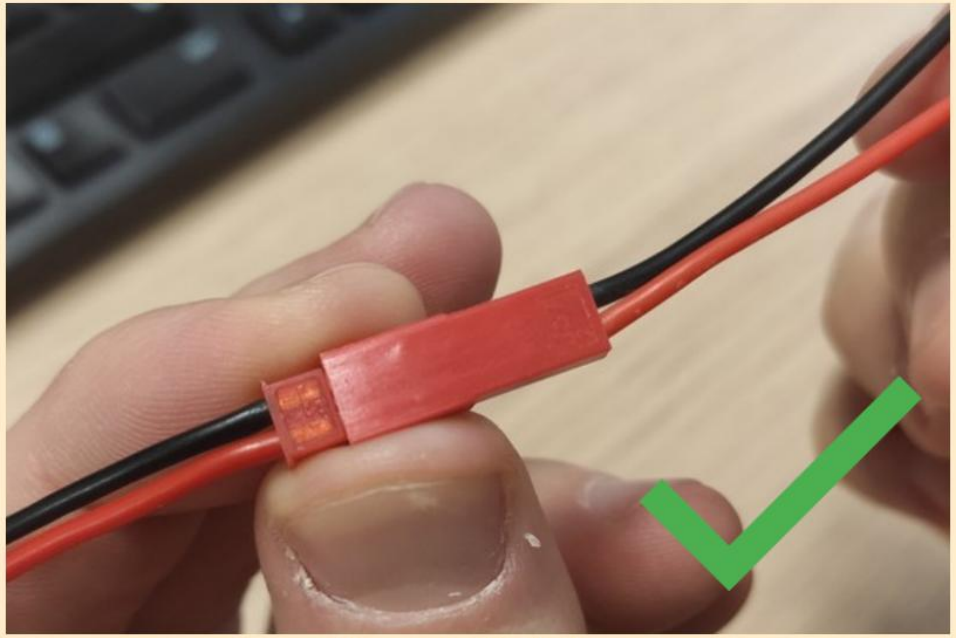
The objective of this task is to characterize the performance of an existing attitude controller for the Drone in 1DoF configuration.

Preliminary setup:

3. Install and connect the battery
4. Wait for a beep indicating drone system started

Warning

Ensure correct polarity to avoid damage.



Ensure correct polarity.

Part 1 - Familiarization with the platform ANT-X

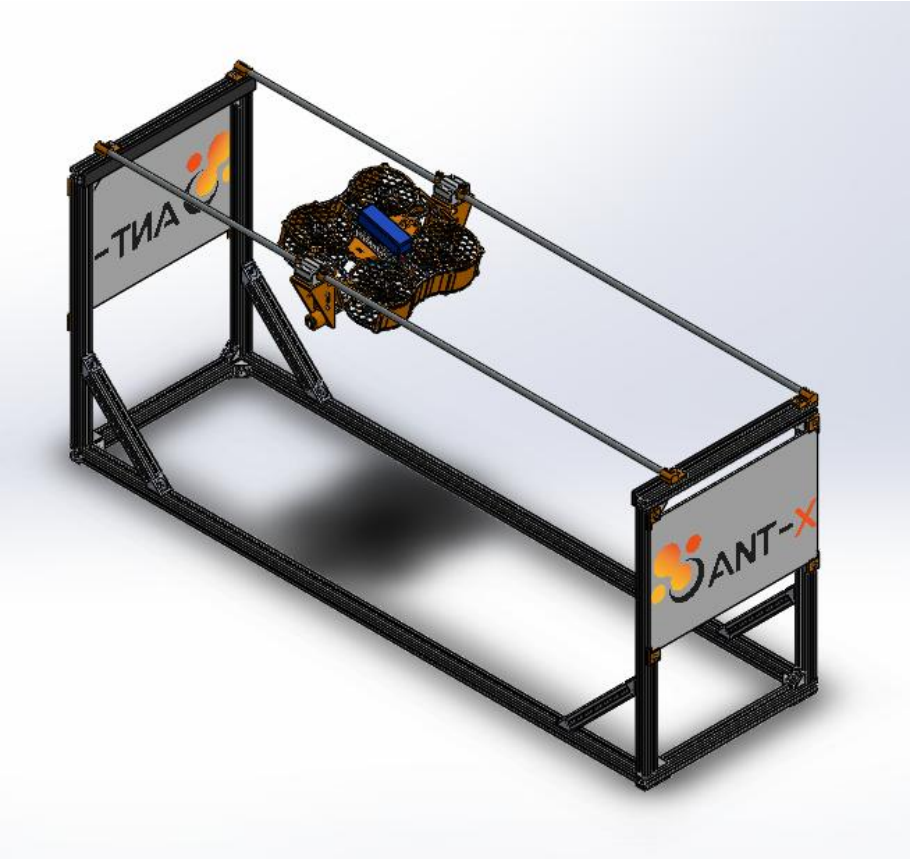
The procedure to activate and control the drone is as follows:

1. **Connect to the Wi-Fi network of the Drone**: run the Wi-Fi script located on the desktop of the PC.
2. Open Matlab
3. In Matlab, go to folder

home\ant-x\ANT-X_tools\DroneCmd\DCTA\

4. Open the File

S3_drone2dof_attitudeStep.m

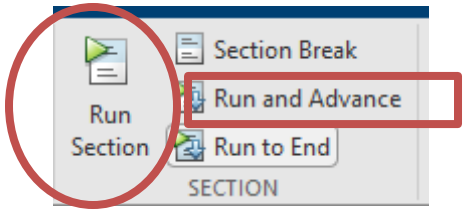


Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The procedure to activate and control the drone is as follows:

5. Run the first sections to connect to the drone within the ROS environment.

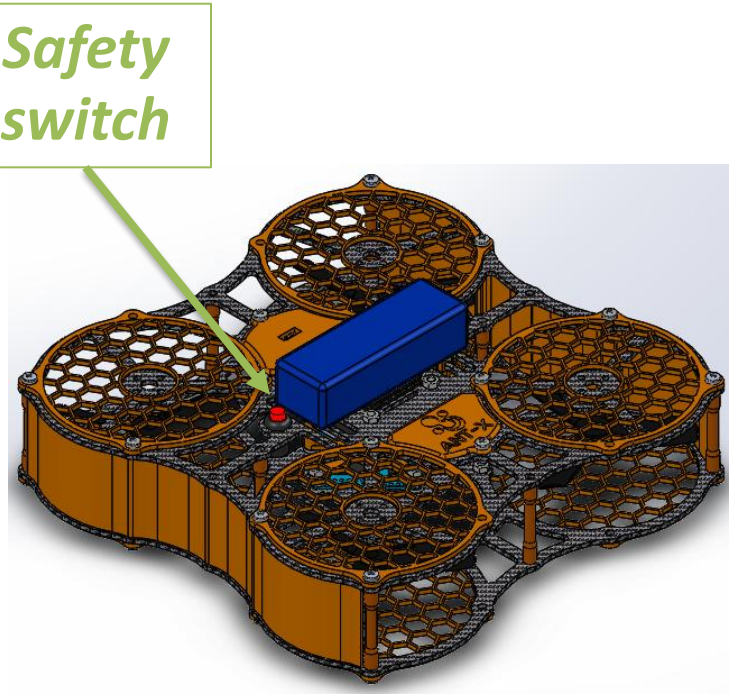


The script must be run step-by step using the *Run Section* or *Run and Advance* buttons.

```
%% Start up the ROS environment. This operation needs to be done only at MATLAB
% startup, and is valid for the entire MATLAB session.
% You can also do the complete cycle of startup/shutdown at every experiment.
% In that case, remember to power-cycle the drone at each experiment.
Drone2DOF.startup();

%% Create UAV object
drone = Drone2DOF('/antx');

%% Open connection: create services, publishers, subscribers, set offboard
drone.connect();
```

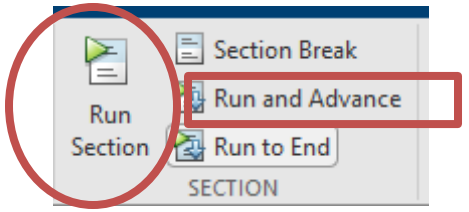


After these three steps, the drone must be started by activating the **safety switch**.

Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

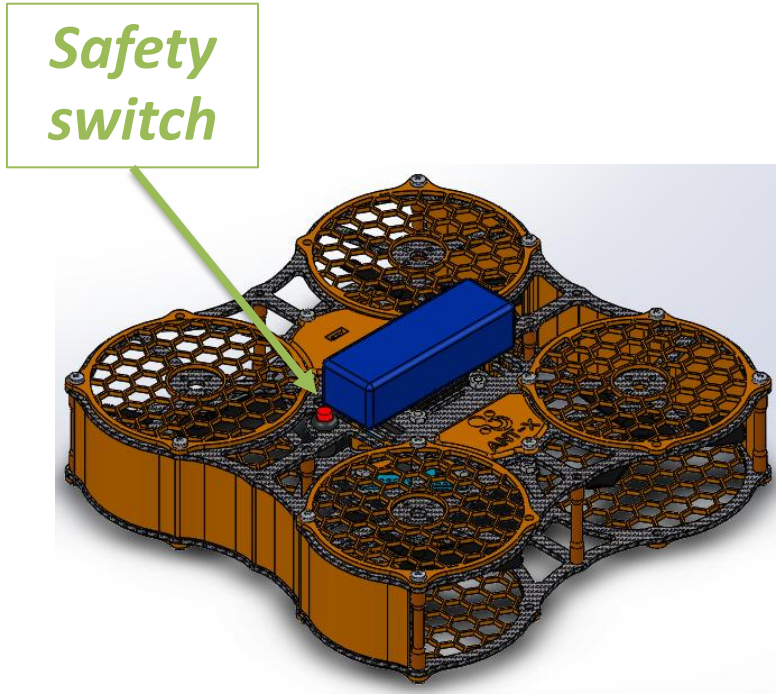
- This script sets the drone in attitude tracking mode.



- A sequence of angular position set points is sent to the FCC.

- In the first test, set the amplitude of the steps to +/-10°

```
%% define sequence of steps
theta0_val = [0; -10; 0; 10; 0; 0]'/180*pi;
dt = .1;
T_period = 2;
```



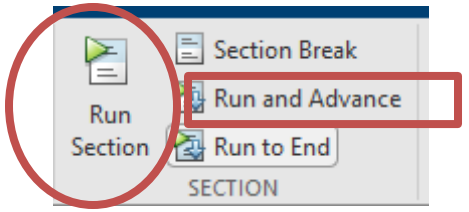
- Once you have set/verified the amplitude of the set point, run the next three sections and observe the movement of the drone.

Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

- Repeat the experiment with a larger amplitude of the reference signal of +/- 25°

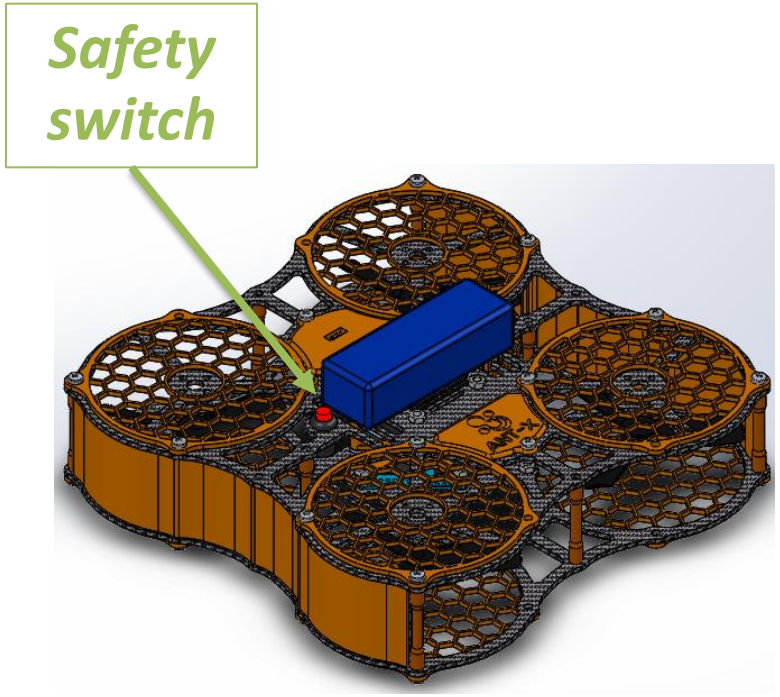
```
%% define sequence of steps
theta0_val = [0; -25; 0; 25; 0; 0]'/180*pi;
dt = .1;
T_period = 2;
```



- Once you have set/verified the amplitude of the set point, run again the *connect->run->disconnect* sections and observe the movement of the drone.

- Run the disconnect section

```
%% Close connection
drone.disconnect();
```



Part 1 - Familiarization with the platform ANT-X

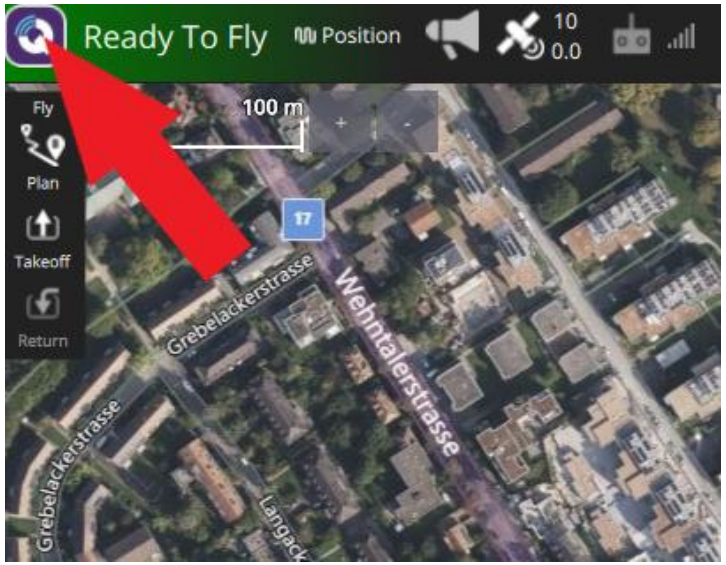
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

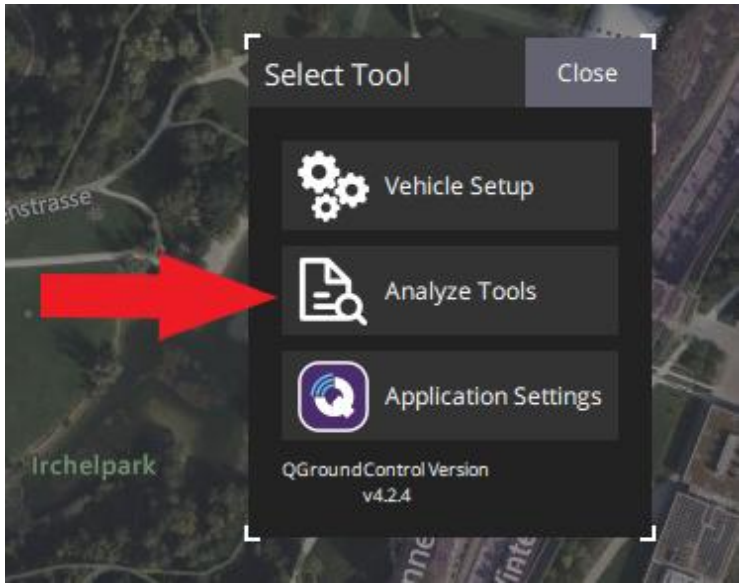
Post-processing - data download

Once the tests are finished, follow this procedure:

- Disconnect the battery
- Connect USB cable to the drone top connect, and wait for a beep
- On the PC desktop launch QgroundControl
- Click on the Q icon in the top left corner.
- Open *Analyze tool*
- Click on *Log Download* in the left column.
- Click on *refresh*
- Select and download file *.ulg to parser folder:

/home/ant-x/ANT-X_tools/px4-log-parser/ulg_logs





Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

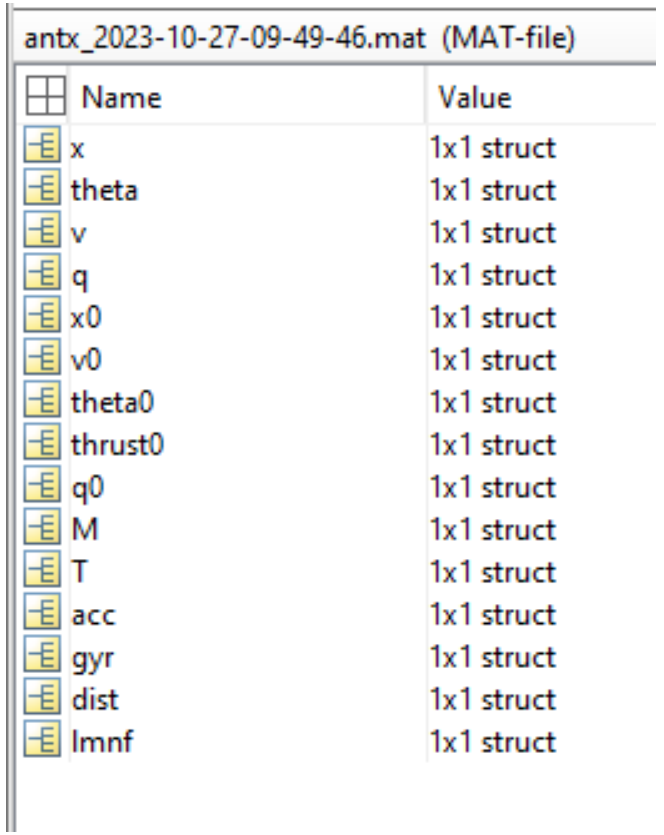
Post-processing

- Run script `parser_script.m`, updating the filename with the name of your downloaded file,

```
twoDofLogConverter('log_1_2025-xxx-xxxx')
```

This will export the logged data to Matlab format.

- Save the *.mat file in a USB memory and transfer to your PC
- Load the file to Matlab in your PC. You must see a set of struct variables like the one on the right.
- For this experiment we are interested in the following variables:
  - theta: pitch angle (rad)
  - theta0: pitch reference (rad)
  - q: pitch rate (rad/s)
  - M: normalized torque (dimensionless)

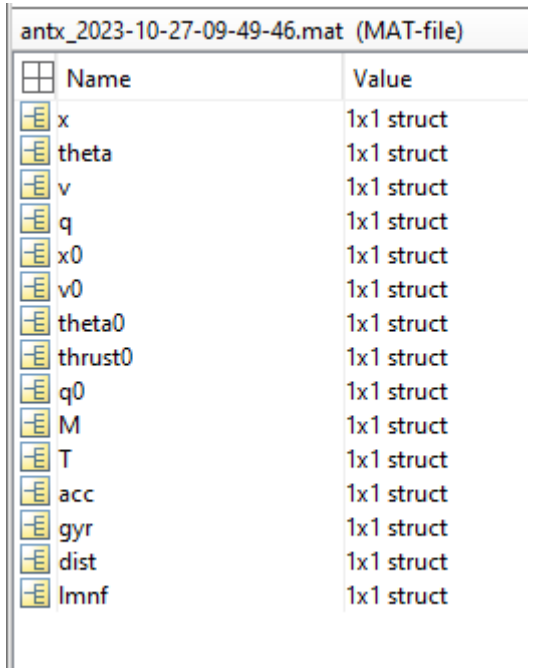


Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Post-processing:

- Determine the effective sampling rate of the logger. Use the command *diff(var.timestamp)*, for one of the variables of interest.
  - What is the average sampling time?
  - What is the worst-case
- Plot the step response of the pitch control system and characterize the performance of the loop in terms of standard KPIs
    - Rise time
    - Settling time
    - Overshoot
    - Oscillation period
    - Input peak amplitude and saturation events



Part 2 - Angular rate dynamics identification

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

Objective

Obtain a sufficiently accurate model describing the pitch motion of a quadrotor when the translational degree of freedom is constrained.

Methodology

Time-domain identification based on a simplified physical model of the attitude dynamics.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The linearized equations of motion describing the pitch dynamics of a quadrotor:

$$\dot{\theta} = q$$
$$J_\theta \, \dot{q} = M_c + M_e$$

where

- $\theta \in R$ is the pitch angle;
- $q \in R$ is the pitch angular rate;
- $J_\theta \in R_{>0}$ is the pitch inertia moment;
- $q \in R$ is the pitch rate;
- $M_c \in R$ is the pitch control torque;
- $M_e \in R$ is an exogenous torque perturbing the pitch motion.

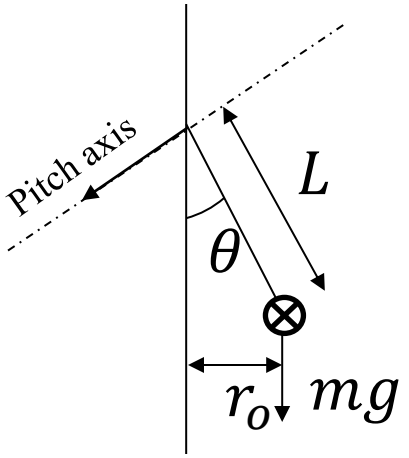
Note that we dropped herein the Δ notation with respect to *the Introductory description* to represent perturbation of the states.

Part 2 - Angular rate dynamics identification

The external torque acting on the pitch axis of the ANT-X 2DoF is the sum of several contributions, *i.e.*, $M_e = M_e^g + M_e^f + M_e^a$, which are defined as follows.

- The axis of the shaft on which the quadrotor is mounted does not pass exactly through the CoM
  - the **gravitational** force generates a small torque about shaft axis:

$$r_o = L\sin(\theta)$$
$$M_e^g = -mgr_o = -mgL\sin(\theta) = -k_o\sin(\theta) \approx -k_o\theta$$



Note that the drone is designed such that the CoM is <u>below</u> the shaft axis.

- The rotative ball bearings cause a small **friction** at the interface with the shaft
  - damping effect on the pitch dynamics: $M_e^f = -c_f q$.
- The quadrotor is affected by **aerodynamic** forces
  - damping effect due to the interaction with air: $M_e^a = -c_a q$.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Summing up all the contributions, a general form for the exogenous torque M_e is

$$M_e = -cq - k\theta$$

where

- $c = c_a + c_f$ is the damping coefficient (aerodynamic damping + ball bearings friction);
- $k = k_o$ is the stiffness coefficient (proportional to the gravity force and the offset of the CoM with respect to the shaft axis).

The delivered control torque M_c can be related to the desired input M_c^d (representing the desired control torque) by considering a scaling factor η:

$$M_c = \eta M_c^d.$$

The actuator dynamics, approximated by a delay block in *Introductory description* will be dealt with latter.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

By introducing the following definitions

$$2\xi\omega_n = \frac{c}{J_\theta}, \qquad \omega_n^2 = \frac{k}{J_\theta}, \qquad \mu = \frac{\eta}{J_\theta},$$

the pitch axis dynamics can be conviniently rewritten as the following set of first order linear differential equations:

$$\dot{\theta} = q$$
$$\dot{q} = -2\xi\omega_n q - \omega_n^2\theta + \mu M_c^d$$

where

- ξ is the damping ratio;
- ω_n is the natural frequency;
- μ is the input gain.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The dynamical model can be compactly written in state-space form:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

where $\mathrm{x} = \begin{bmatrix} \theta \\ q \end{bmatrix}$ is the state vector, $\mathrm{A} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix}$ is the system dynamics

matrix, and $\mathrm{B} = \begin{bmatrix} 0 \\ \mu \end{bmatrix}$,

$C = [0 \quad 1]$ are the input and output matrix, respectively.

$$G(s) = \frac{q}{M_c^d} = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The identification problem can be split in two subproblems:

1. estimate the parameters related to the free response ω_n and ξ (*i.e.*, the natural frequency and the damping ratio);
2. estimate the gain μ.

The overall approach requires performing two experiments:

1. Free response to estimate ω_n and ξ;
2. Forced response to estimate μ.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Assuming the observed response belongs to a 2nd order underdamped system, the natural frequency ω_n and damping ratio ξ can be estimated based on the oscillation period T_p and the settling time T_s:

$$T_P = \frac{2\pi}{\omega_n\sqrt{1-\xi^2}} \approx \frac{2\pi}{\omega_n}$$

$$T_s \approx \frac{4.6}{\xi\omega_n}.$$

To measure such parameters, a viable solution is to perturb the system by injecting a small torque for a small amout of time (to avoid hitting the carbon rod) and then analyze the free response.

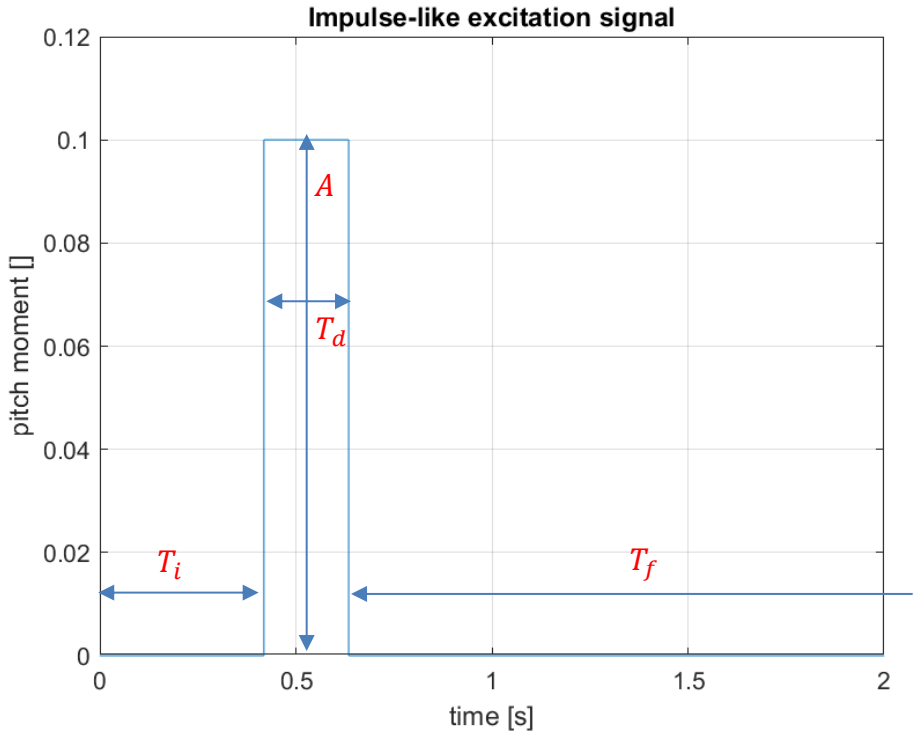
- This approach is safe and allows one to obtain a repeatable experiment.
- It is not possible to perturb the system with a step torque (from which one could deduce the same parameters of interest) because a constant torque applied for a sufficiently long time will likely make the platform hit the carbon rod.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Data collection task

1. Go to the folder …\Antx\DroneCmd\DCTA\
2. Open the File *S1_drone2dof_ident_1dofmode_impulse.m*
3. Set up the Matlab script to perform the estimation experiment for the free response parameters.



An impulse-like excitation signal is used to:

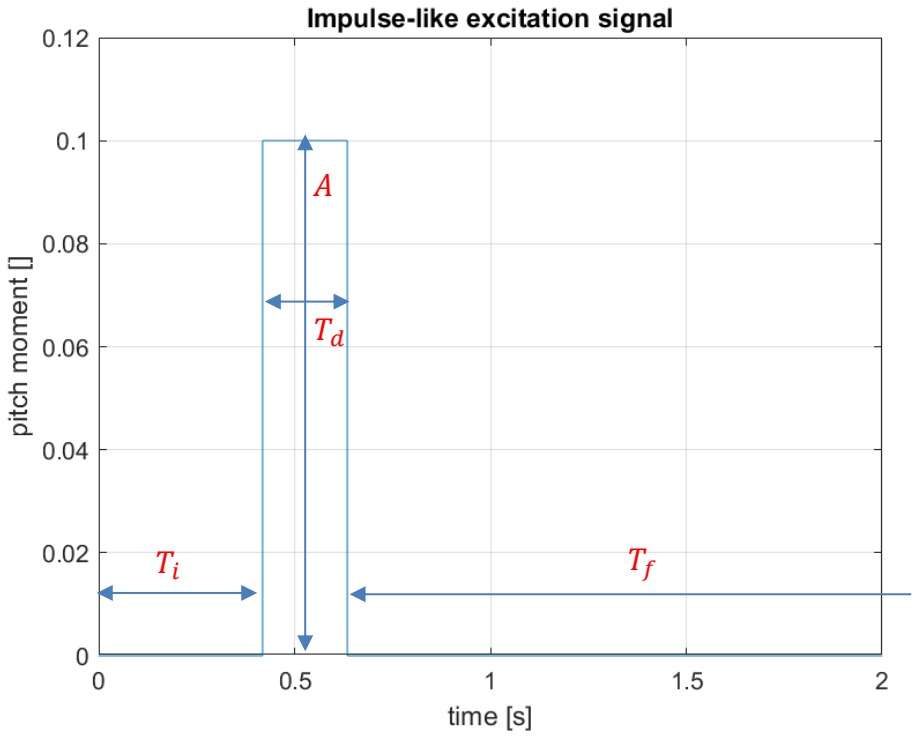
- Drive the system far away from equilibrium;
- Observe the free response starting from the time instant the «impulse» is removed.

The experiment is carried out in open-loop!

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Data collection task



The input signal (impulse-like) is characterized by the following parameters:

Amplitude A	0.1
Duration T_d	$0.2s$
Initial time T_i	$0.4s$
Final time T_f	$20s$

A thrust value of 0 should be applied in the experiment (motors spin at minimum speed). The experiment must be carried out in open-loop.

Part 2 - Angular rate dynamics identification

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

Analysis of the results

4. Run the experiment and export the data log as explained in the first part.

5. To validate the model, repeat the experiment twice.

6. In your PC load the data sets and plot the pitch rate response.

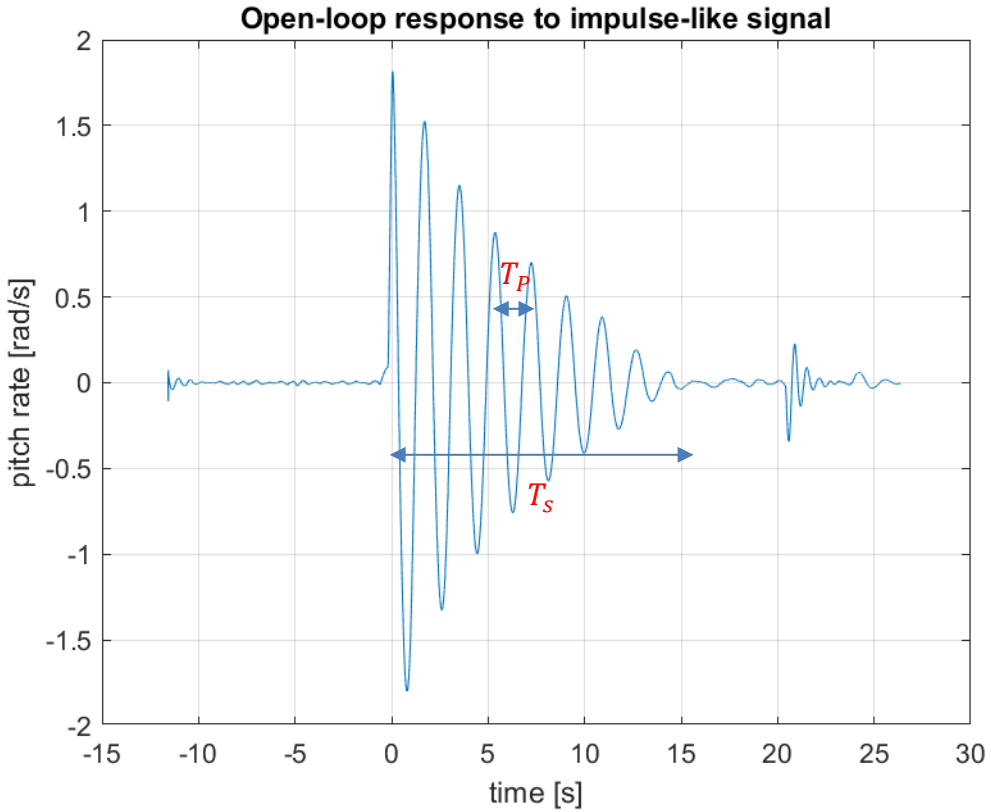
Remember to check the sampling time of the log.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the results

The data collected during the first experiment should correspond to a plot like the one bellow. Use the time series to estimate the model parameters:



Oscillation period T_P	
Settling time T_s	

$$T_P = \frac{2\pi}{\omega_n\sqrt{1-\xi^2}} \approx \frac{2\pi}{\omega_n}$$

$$T_s \approx \frac{4.6}{\xi\omega_n}.$$

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The second experiment is intended to estimate the gain μ of the transfer function

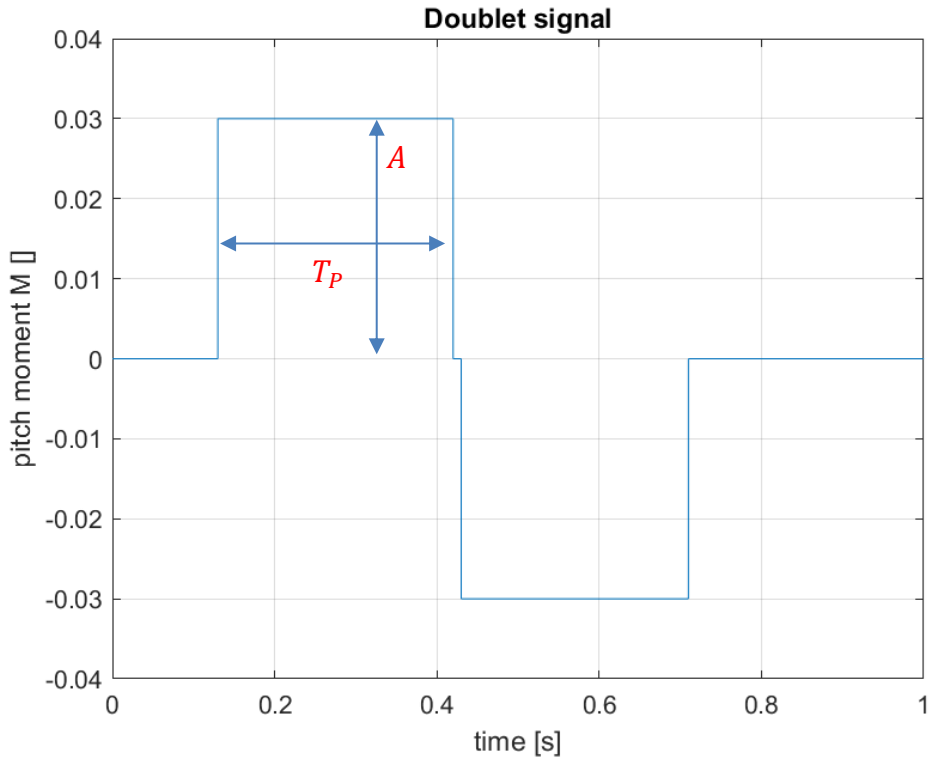
$$G(s) = \frac{q}{M_c^d} = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

To estimate the gain, one can appeal to the assumed linearity of the model and compare the response of the platform with a simulated one obtained using the previously estimated parameters.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

To achieve our goals, a suitable excitation signal is a doublet, which is made by steps and counter-steps with given amplitude and period:



- The amplitude A shall be large enough for the signal-to-noise ratio to be acceptable and to sufficiently excite the vehicle dynamics, but not so large to avoid exceeding the physical limits of the test-bed:
  - *the attitude angle shall not exceed 50 deg or the drone will likely hit the carbon rod.*

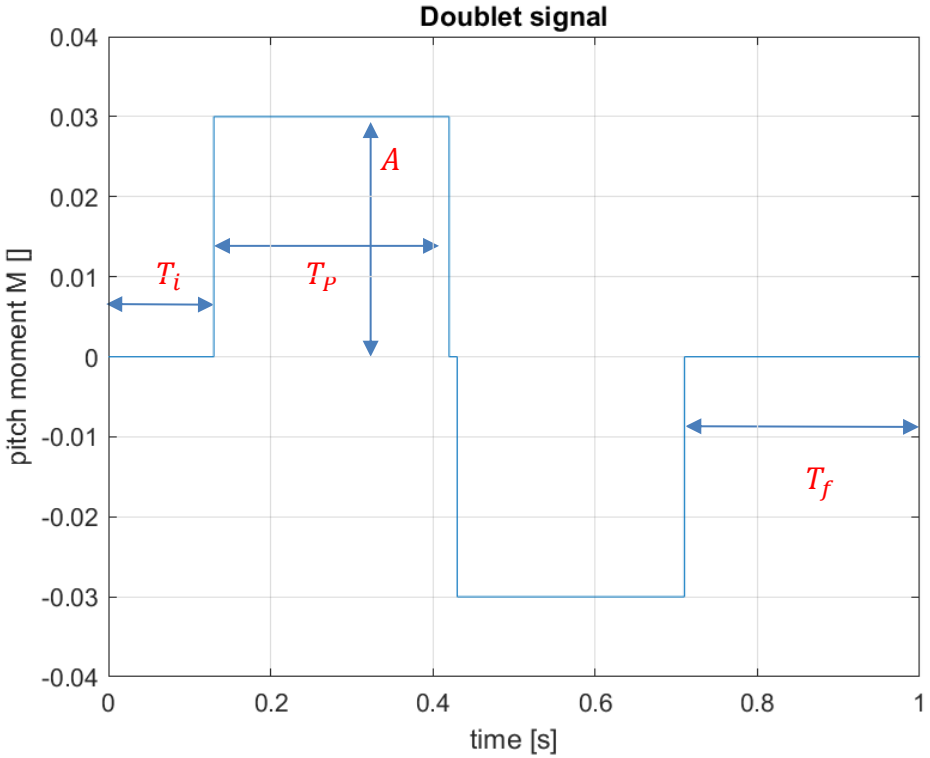
- For the same reason, the period T_p of the doublet shall not be too large. It is also advisable to excite the system at a sufficiently larger frequency than the natural one to avoid resonance.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Data collection task

1. Go to the folder …\Antx\DroneCmd\DCTA\
2. Open the File *S1_drone2dof_ident_1dofmode_doublet.m*
3. Set up the Matlab script to perform a doublet response experiment. The parameters to be used for the doublet signal are:



Amplitude A	0.03
Period T_p	$0.6s$
Initial time T_i	$0.15s$
Final time T_f	$0.3s$

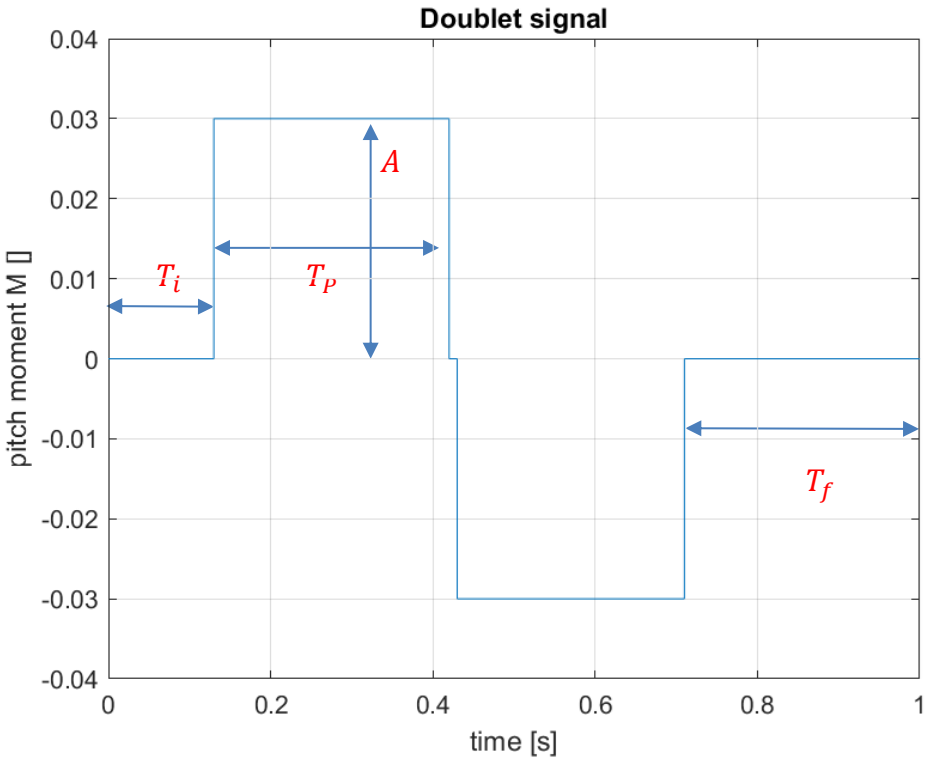
- A thrust value of 0.4 should be applied in this experiment.
- The experiment must be carried out in open-loop.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Data collection task

4. Repeat the experiment with a different doublet signal for validation purposes. The parameters to be used for the second doublet signal are:



Amplitude A	0.02
Period T_p	$0.8s$
Initial time T_i	$0.15s$
Final time T_f	$0.3s$

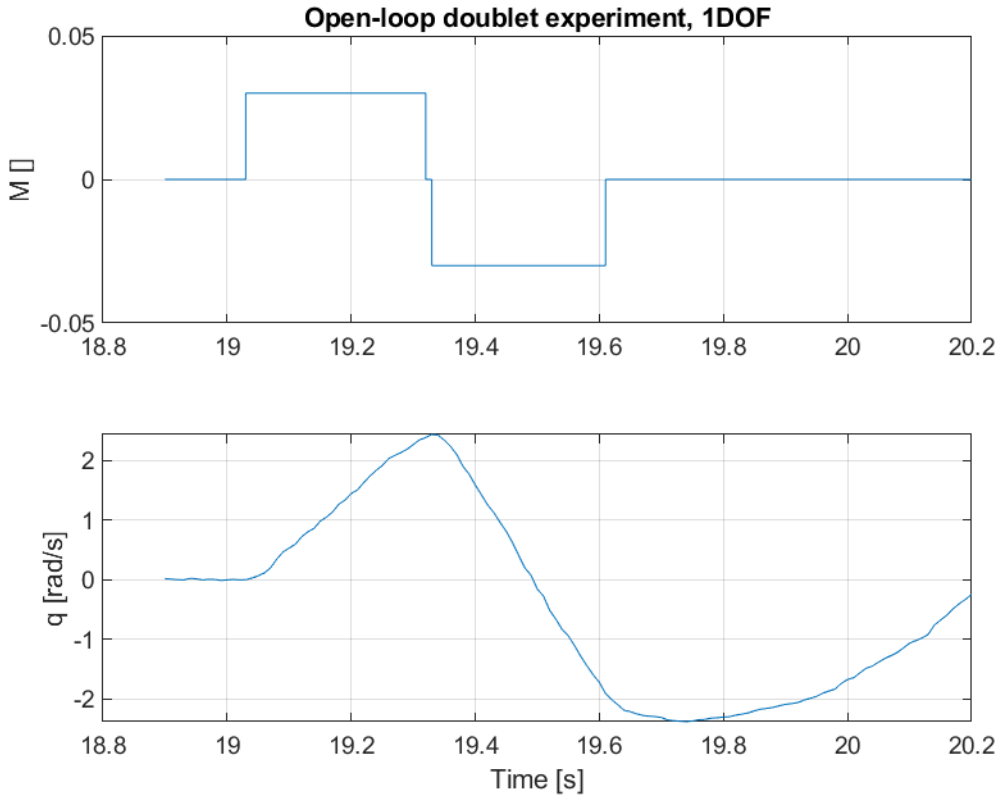
- A thrust value of 0.4 should be applied in this experiment.
- The experiment must be carried out in open-loop.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the results

5. Run the experiments and export the data log as explained in the first part.
6. The data collected during the first doublet experiment should correspond to a plot like the one below. Use the time series to estimate the model gain μ.



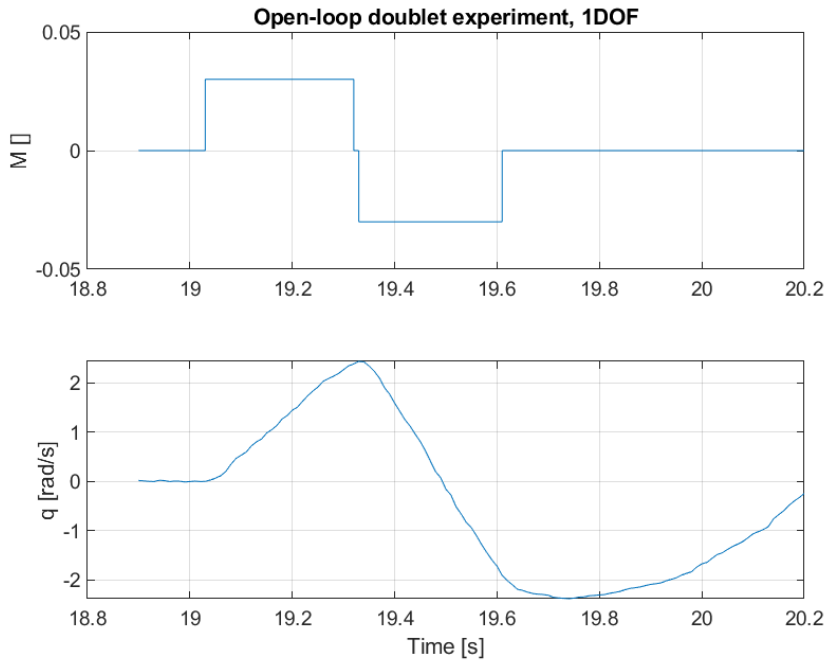
Procedure to estimate μ:

- Plug the estimated ω_n and ξ into $G_1(s)$ with $\mu = 1$.
- Simulate $G_1(s)$ with the input M_c^d used in the experiment to obtain $y_1(t)$ (simulated output)
- Since $\frac{G(s)}{G_1(s)} = \mu$, if $y(t)$ and $y_1(t)$ are obtained from the same input M_c^d, then $\frac{y(t)}{y_1(t)} = \mu$;
- Compute peak of responses of $y(t)$ and $y_1(t)$ (y_{PK} and y_{PK}^{SIM}, respectively)
- Estimate μ by computing $\mu = \frac{y_{PK}}{y_{PK}^{SIM}}$.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the results



Procedure to estimate μ:

- Plug the estimated ω_n and ξ into $G(s)$ with $\mu = 1$.

$$G(s) = \frac{q}{M_c^d} = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

- Simulate $G'(s)$ with the input M_c^d used in the experiment to obtain $y_1(t)$ (simulated output)
- Since $\frac{G(s)}{G'(s)} = \mu$, if $y(t)$ and $y_1(t)$ are obtained from the same input M_c^d, then $\frac{y(t)}{y_1(t)} = \mu$;
- Compute peak of responses of $y(t)$ and $y_1(t)$ (y_{PK} and y_{PK}^{SIM}, respectively)
- Estimate μ by computing $\mu = \frac{y_{PK}}{y_{PK}^{SIM}}$.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Several effects have been neglected in the second-order model:

- Actuator/sensor dynamics
- Computational delays
- Zero-order-hold (delay due to digital implementation of the control law)

All these effects can be approximated with a pure delay τ, such that the transfer function from the pitch moment command to the pitch rate can be written as:

$$G_q(s) = G_a(s)G(s) = e^{-\tau s}\frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

where $G_a(s) = e^{-\tau s}$.

The delay must be estimated from data. To estimate the time-delay τ, one can either:

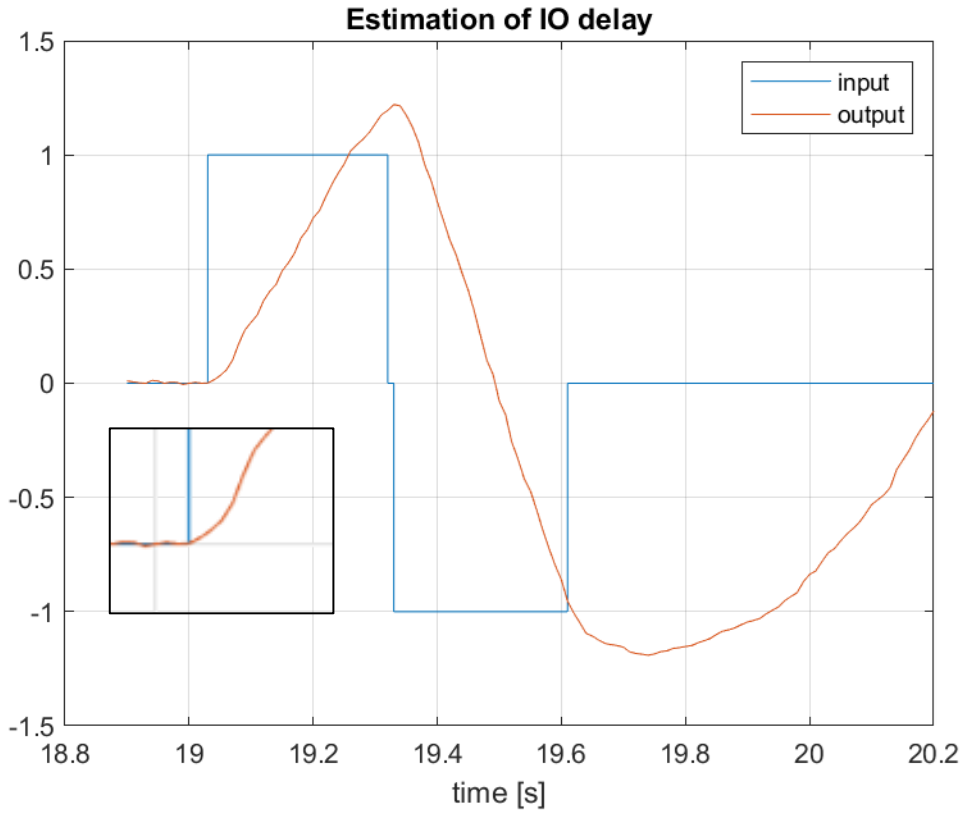
1) manually count how many time samples pass between the moment in which the input was applied and the moment in which the output begins moving from equilibrium;
2) use the Matlab delayest() routine on the collected data.

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the results

The data collected during the second experiment should correspond to the following plot:



- From the logged data, the delay can be estimated to be approximately $n_d = 2$ samples.

Verify this on your data!!!

- Given the sampling period of the data log evaluate the estimated delay as

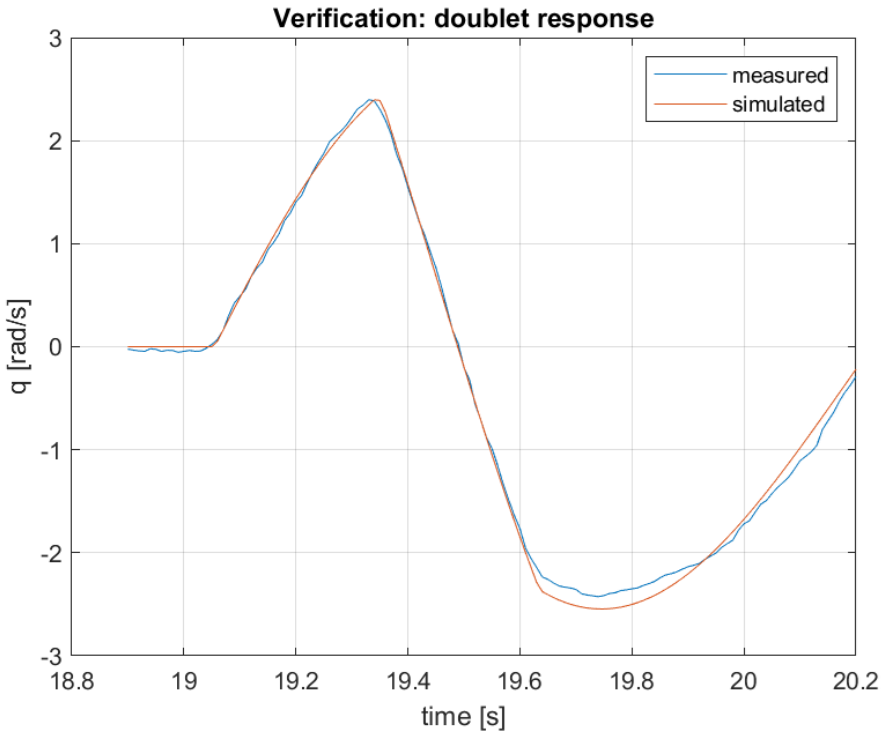
$$\tau = n_d t_s$$

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Verification of the identified model

Verification is carried out by comparing the response obtained with the same data-set used to estimate the parameters.



- Simulate the final model

$$G_q(s) = e^{-\tau s} \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

using the double input.

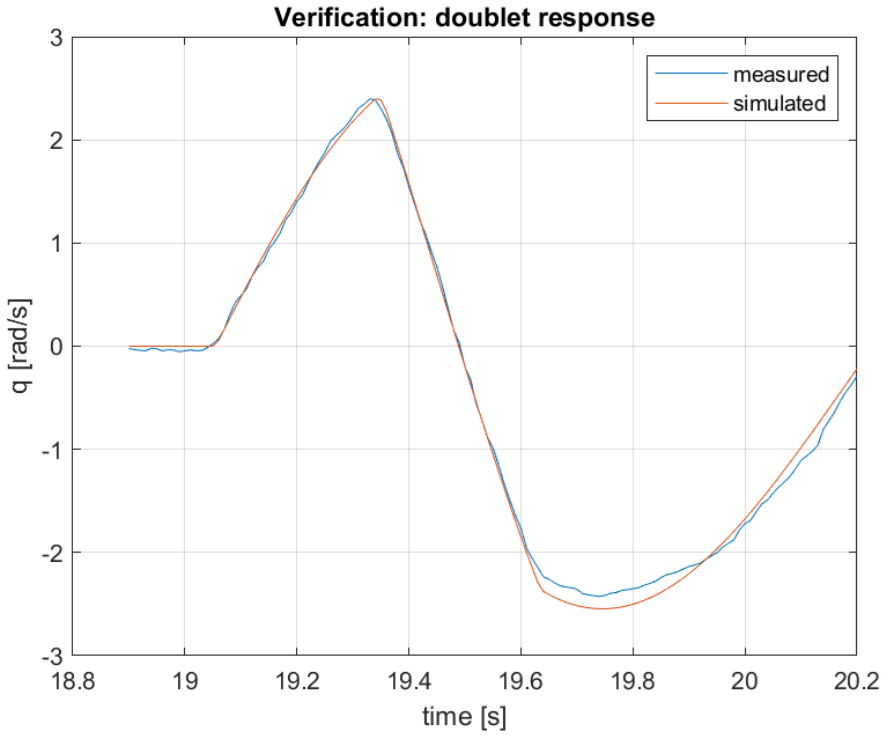
- Compare the simulated responses with the experimental ones. What is the Root Mean Squared Error (RMSE)?

Part 2 - Angular rate dynamics identification

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Validation of the identified model

- Fresh data (not used in the estimation procedure) must be used to validate the model.
- Compare the response on the model obtained with the second impulsive input and the second doublet.



- Simulate the model

$$G_q(s) = e^{-\tau s}\frac{\mu s}{s^2+2\xi\omega_n s+\omega_n^2}$$

using the new inputs.

- Compare the simulated responses with the experimental ones. What is the Root Mean Squared error (RMSE)?
- Is the RMSE higher or lower than the one obtained in verification?

Part 2 - Angular rate dynamics identification

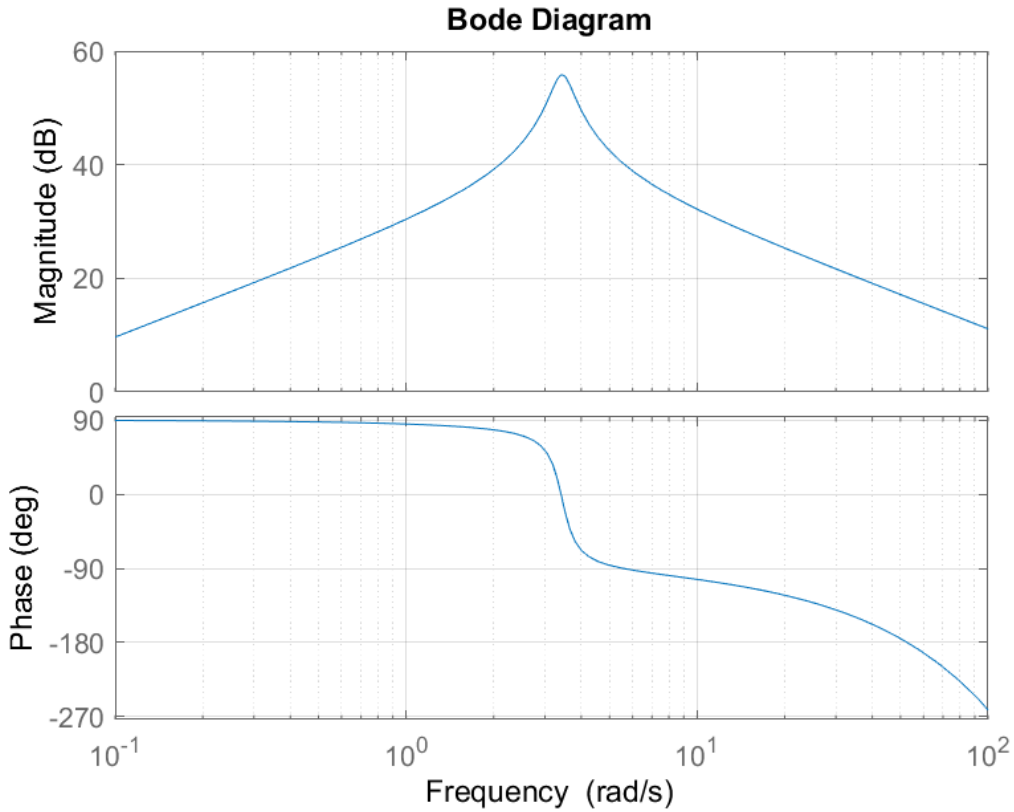
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the identified model

$$G_q(s) = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-s\tau}$$

Using the estimated parameters plot the frequency response of the model.

It must look like the one on the right.



This is the end of the activity.

POLITECNICO MILANO 1863

DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

M.Sc. AERONAUTICAL/SPACE ENGINEERING

051401 - DIGITAL CONTROL TECHNOLOGY FOR AERONAUTICS (DCTA)

Academic year 2025/2026

Prof. Fredy Ruiz

Assistant Alessandro Del Duca

Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

November 2025

Laboratory sessions

The experimental activity is based on the ANT-X platform, available at the Aerospace Systems & Control Laboratory.

It is divided into four sessions:

1. 1DOF modeling, November 20
2. Attitude Control, November 27
3. Cascade Control, December 4
4. Position Control, December 11

https://www.antx.it/

Preliminary work - Angular rate dynamics identification

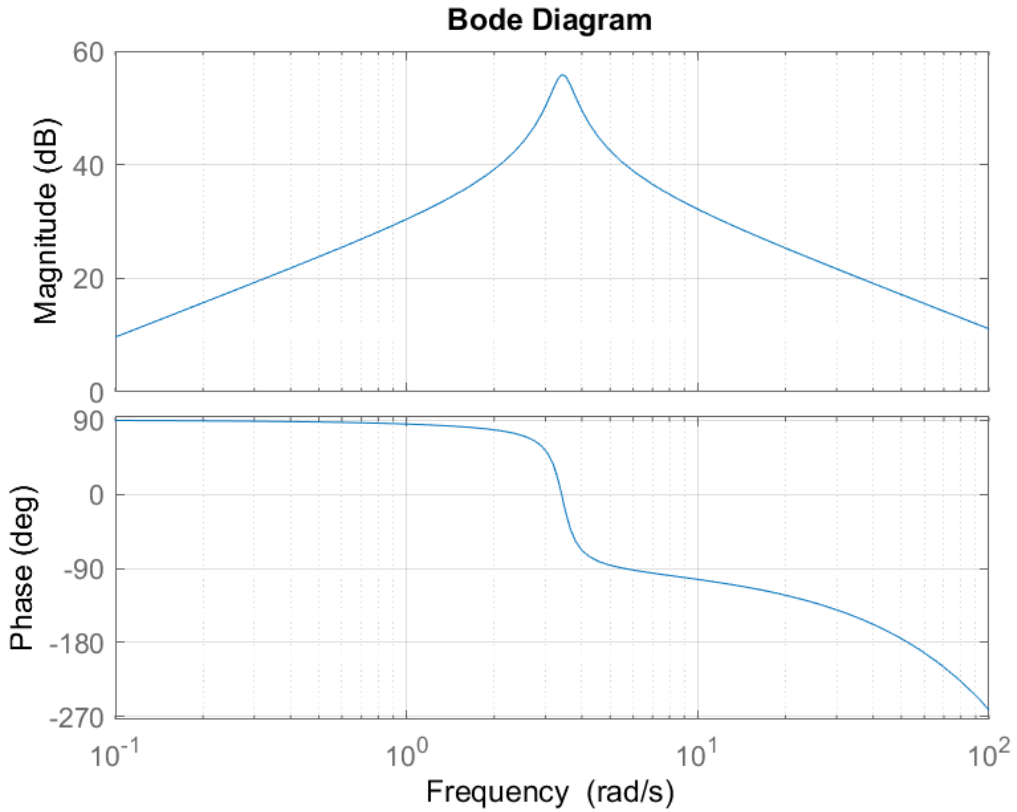
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Identified model

The transfer function from the pitch moment command to the pitch rate was estimated as:

$$G_q(s) = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-s\tau}$$

It must look like the one on the right.



Expected parameters:

$\omega_n = 3.4 \frac{rad}{s}$,

$\xi = 0.085$,

$\mu = 360$

$\tau = 20\ ms.$

The specific values of parameters can be different for each setup.

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Objective

Starting from the identified model describing the pitch motion of the ANT-X 2DoF drone, tune and implement a PID controller to guarantee closed-loop stability and a desired performance for the pitch rate dynamics.

Methodology

- PID design by loop-shaping in continuous-time
- Digital implementation

Outline of the lecture

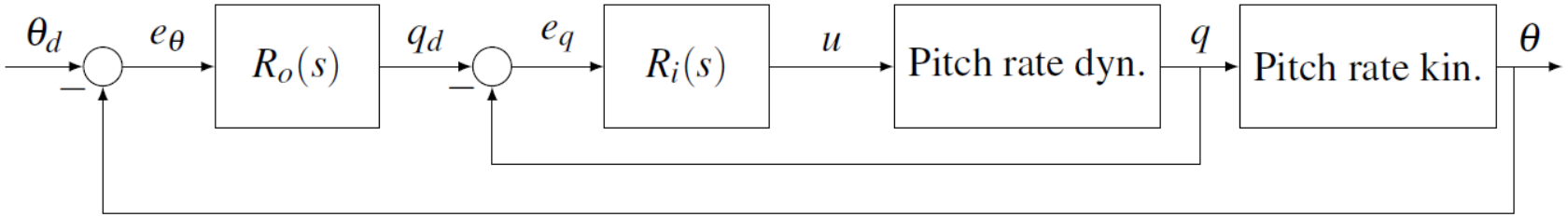
- PID controller structure
- Tuning through loop-shaping
- Digital implementation
- Experiment setup, data collection and analysis of the results.

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Most of the autopilots for attitude control in multi-rotors are based on a cascade control structure for each axis, in which:

- the outer loop computes a reference angular velocity q_d starting from the angular error
  $$e_\theta = \theta_d - \theta;$$
- the inner loop computes the torque u required to track the reference angular velocity q_d.



- The objective of the pitch rate control design is to provide a sufficiently fast response and the rejection of exogenous disturbances, which affect the pitch rate dynamics and not the kinematics.
- Thanks to the large bandwidth of the propellers for small-scale multirotors $(30 - 40 rad/s)$ with respect to the typically desired closed-loop bandwidth (from θ_d to θ, about $10\ rad/s$) , a separation between the inner and outer loop bandwidth can be effectively enforced:
  - the desired closed-loop bandwidth of the inner loop can be set between $15 - 30\ rad/s$ to allow a sufficient separation between the loops

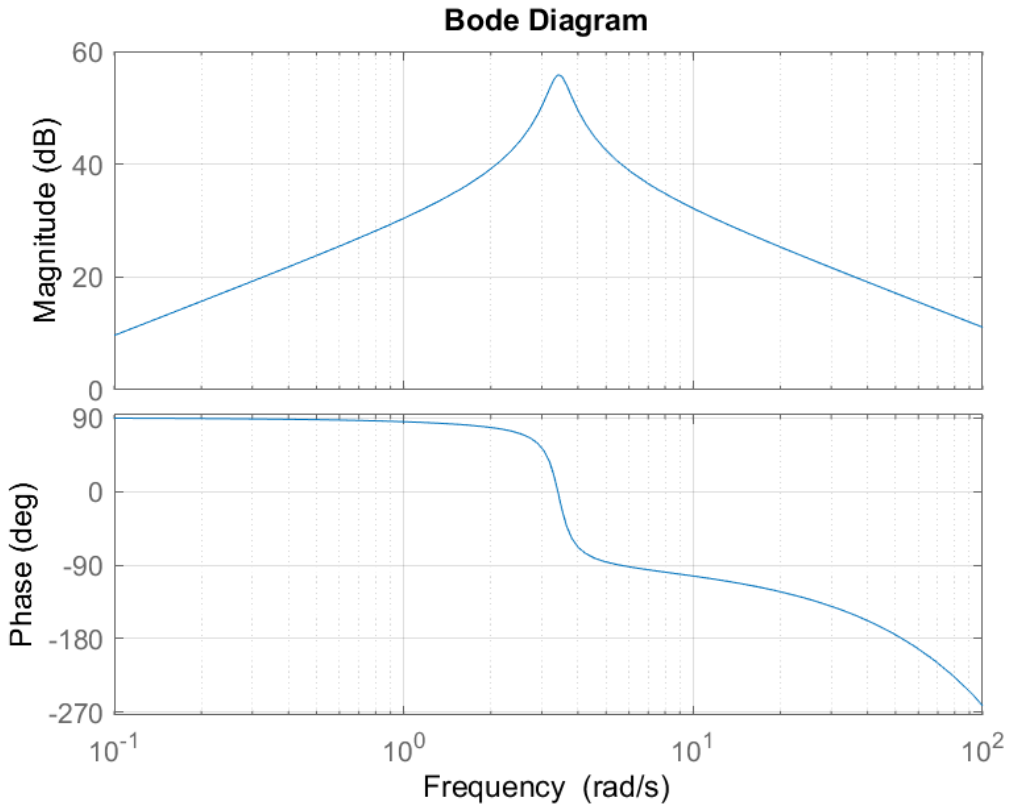
Session 2 - Angular rate control

Let us recall the pitch axis dynamics identified in *Session 1*, which is described by the following set of differential equations:

$$\dot{\theta} = q$$
$$\dot{q} = -2\xi\omega_n q - \omega_n^2\theta + \mu M_c^d,$$

and transfer function

$$G_q(s) = \frac{\mu s}{s^2+2\xi\omega_n s+\omega_n^2} e^{-s\tau}$$



Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Task: Given the T.F. $G_q(s) = \frac{\mu s}{s^2+2\xi\omega_n s+\omega_n^2} e^{-s\tau}$ describing the pitch rate dynamics, tune a PID controller such that the closed-loop system satisfies:

- Phase margin $\phi_m \geq 45^\circ$
- Crossover frequency $\frac{15\text{rad}}{s} \leq \omega_c \leq \frac{30rad}{s}$.

The controller is implemented using a sampling time $T_s = 4ms$ and a backward Euler discretization:

$$s = \frac{z-1}{T_s z}$$

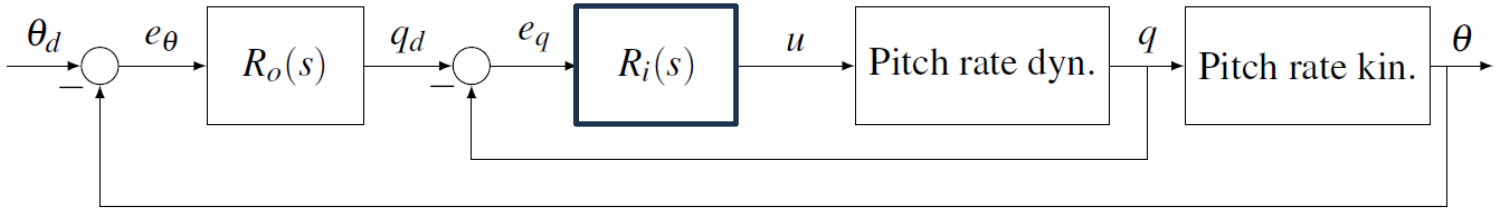
Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The architecture for the angular rate control system contains a **P**roportional-**I**ntegral-**D**erivative (PID) controller:

$$R_i(s) = K_p + \frac{K_i}{s} + K_d \frac{s}{\tau_f s + 1} = \frac{1}{s(\tau_f s + 1)}\left(K_d s^2 + K_i(\tau_f s + 1) + K_p s(\tau_f s + 1)\right).$$

K_p : proportional gain; K_i : integral gain; K_d : derivative gain; τ_f : filter constant



NOTE: The pole of the derivative filter pole is fixed at a frequency higher than the desired closed-loop bandwidth (τ_f=0.01). Then, the two poles of the controller are fixed. The design task requires the selection of the two zeros and the loop gain.

Session 2 - Angular rate control

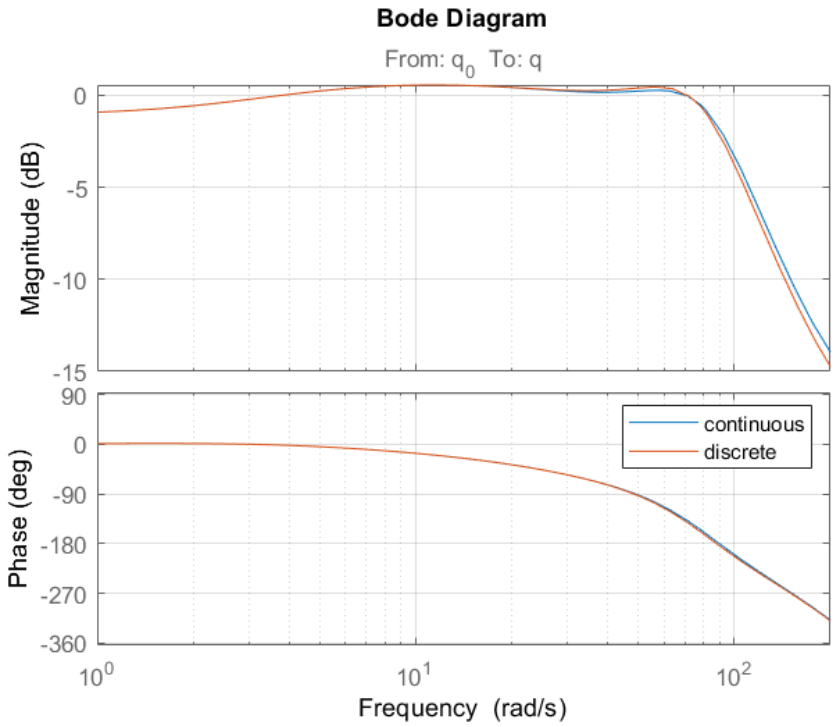
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

- Evaluate the robustness of your controller considering the delay of the ZOH.

- Obtain the expected sensitivity and complementary sensitivity functions, with and without considering the ZOH.

- Analyze the performance degradation caused by the digital implementation.

Note that for the present application the delay due to digital implementation (equal to half the sampling time $\frac{T_c}{2}$) is negligible with respect to the overall delay, due to the high sampling rate. However, this is not the case for many applications

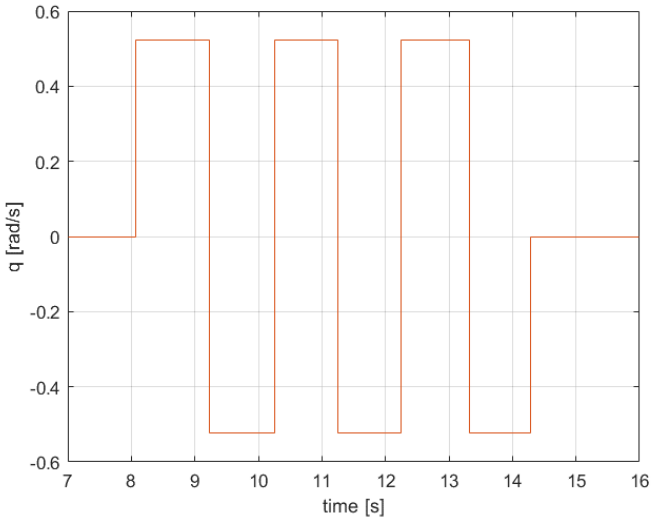


Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

- Evaluate the robustness of your controller considering the delay of the ZOH.

- Test the performance of the control loop using a doublet reference signal:



A doublet signal must be used instead of a more common step because commanding the drone to keep a constant angular velocity for a sufficiently large amount of time would make the drone hit the carbon rod!!

- Evaluate your controller in simulation using the proposed doublet reference signal.

Session 2 - Angular rate control

The controller T.F. is :

$$R(s) = \frac{K_i}{s(\tau_f s + 1)}\left(1 + \frac{K_p + K_i\tau_f}{K_i}s + \frac{K_p\tau_f + K_d}{K_i}s^2\right).$$

On the other hand, the loop-shaping design leads to the selection of a T.F. with three parameters:

$$R(s) = \bar{\mu}\frac{(1 + \tau_1 s)(1 + \tau_2 s)}{s(\tau_f s + 1)}$$

- the location of two zeros through τ_1, τ_2 and the gain $\bar{\mu}$.

Equating the two transfer functions we get:

$$\bar{\mu} = K_i; \quad \frac{K_p + K_i\tau_f}{K_i} = \tau_1 + \tau_2; \quad \frac{K_p\tau_f + K_d}{K_i} = \tau_1\tau_2,$$

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Once the parameters of the controller $R(s)$ are fixed to satisfy the design requirements (if possibile), the PID gains can be easily recovered from the following expressions:

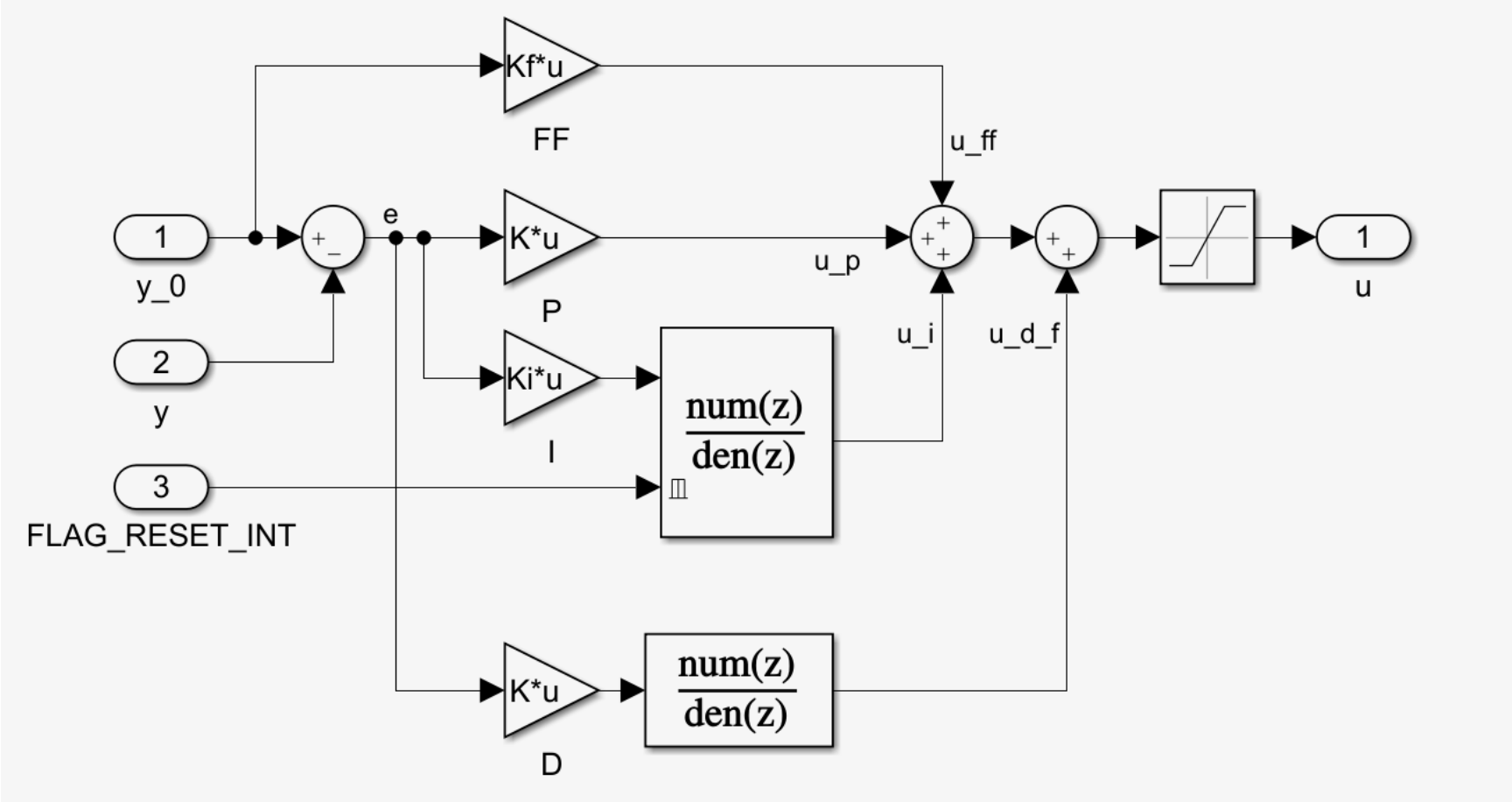
$$K_i = \bar{\mu}$$

$$K_p = (\tau_1 + \tau_2)K_i - K_i\tau_f$$

$$K_d = \tau_1\tau_2 K_i - K_p\tau_f.$$

Session 2 - Angular rate control

Angular rate control: PID controller structure, digital implementation



Integrator transfer function (discrete time)

$$\frac{T_c z}{z-1}$$

Derivative filter transfer function (discrete time

$$\frac{z-1}{(\frac{1}{N}+T_c)z-\frac{1}{N}}$$

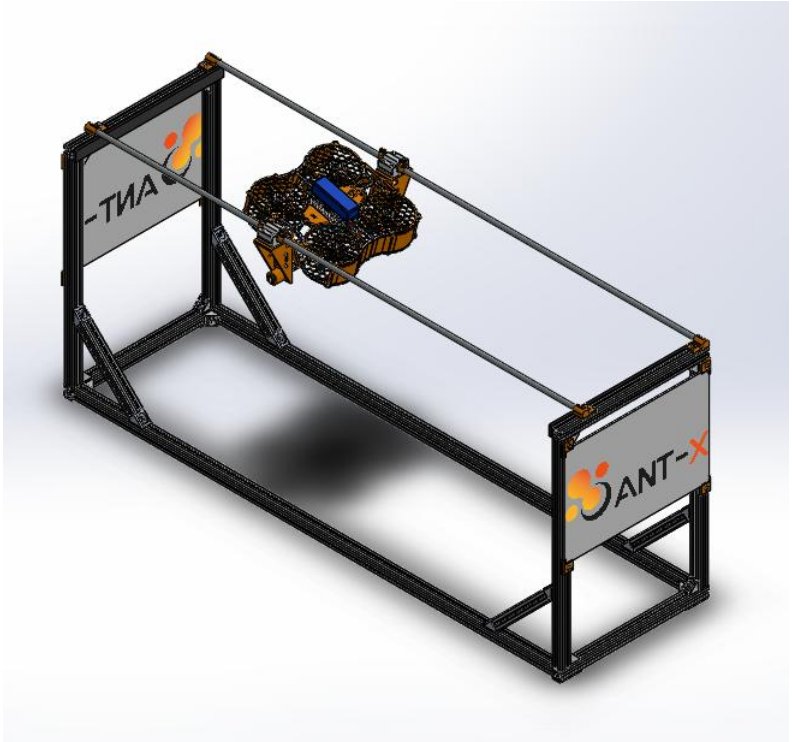
$$N=\frac{1}{\tau_f}$$

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-On test:

1. **Connect to the Wi-Fi network of the Drone**: run the Wi-Fi script located on the desktop of the PC.
2. Open Matlab
3. Go to the folder …\Antx\DroneCmd\DCTA\
4. Open the File *S2_drone2dof_angularRateDoublet_1dofmode.m*



The procedure to activate and control the drone is as follows:

5. Run the first three sections to connect to the drone within the ROS environment, up to **drone.connect();**

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-On test:

The controller parameters are set using the QGroundControl application.

6. Open QGroundControl
7. Go to the vehicle setup menu located in the upper-left corner (the button with a gearbox)
8. Choose the parameters window in the lower-left corner (the gearbox symbol)
9. Adjust the corresponding gain in the controller based on the next table:

Parameter in QGC	Controller gain	Designed value	Admissible range
FC_PARAM_4	angular rate P gain		[0 – 0.12]
FC_PARAM_5	angular rate I gain		[0 – 0.5]
FC_PARAM_6	angular rate D gain		[0 – 0.003]

References: https://ant-x.gitlab.io/documentation_2dofdrone/_chapters/operations.html#modifying-controller-parameters-from-qgroundcontrol

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Once you have updated the controller gains, the drone must be started by activating the **safety switch.**

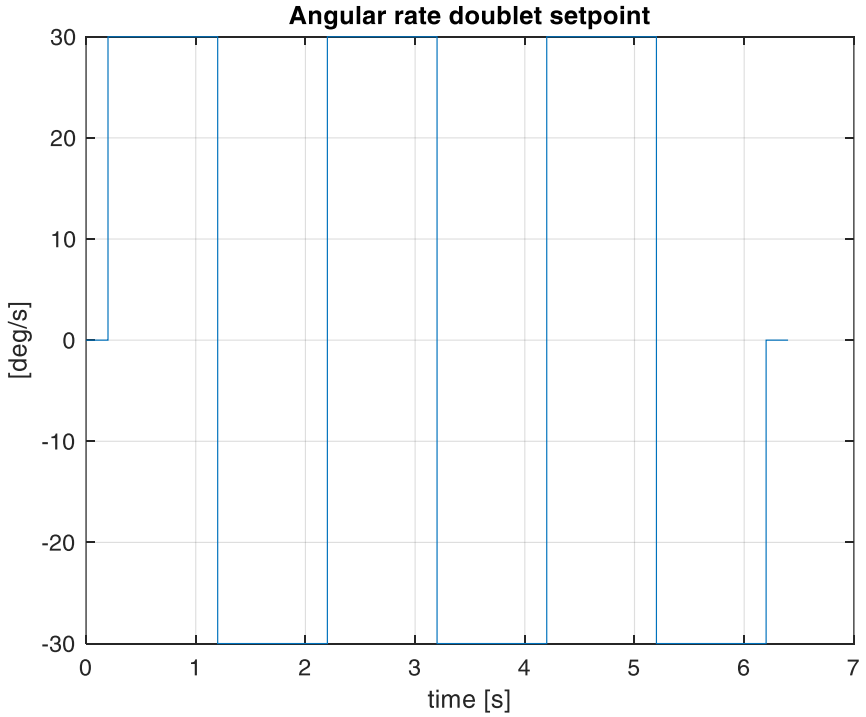
10. Continue the execution of the script, activating the control system.

```
%% define setpoint signal
% DOUBLET SIGNAL
%% ARM
%% determine thrust
%% send level attitude setpoint
%% send angular rate setpoint signal

T_doublet = 1; % s
A = deg2rad(30); % rad/s
```

11. Disarm the drone to stop the control system

```
%% DISARM
drone.disarm();
```



Session 2 - Angular rate control

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

12. Extract the data of the experiments for post-processing

13. Save the file *.mat to a USB stick for analysis on your own PC

14. Plot the angular rate q and the setpoint q_o

15. Plot the normalized torque (manipulated variable) M

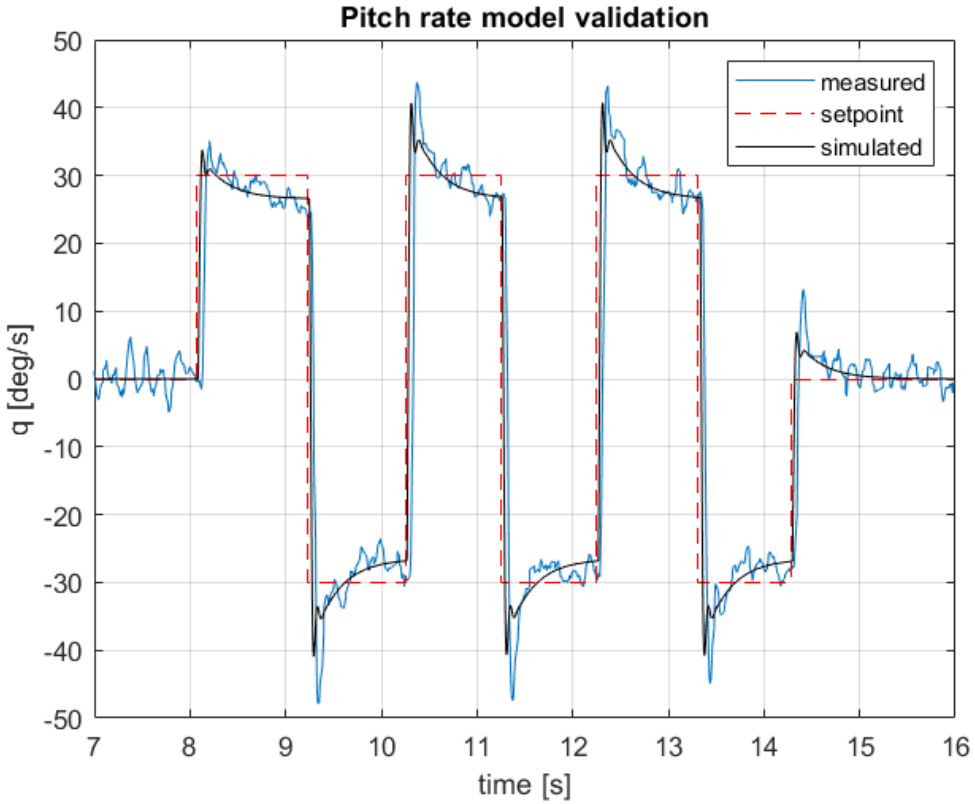
16. Compare the observed behavior with the expected response obtained in the simulation.

Session 2 - Angular rate control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-on task:

Expected response of a control system with $\omega_c = 30\ rad/s$

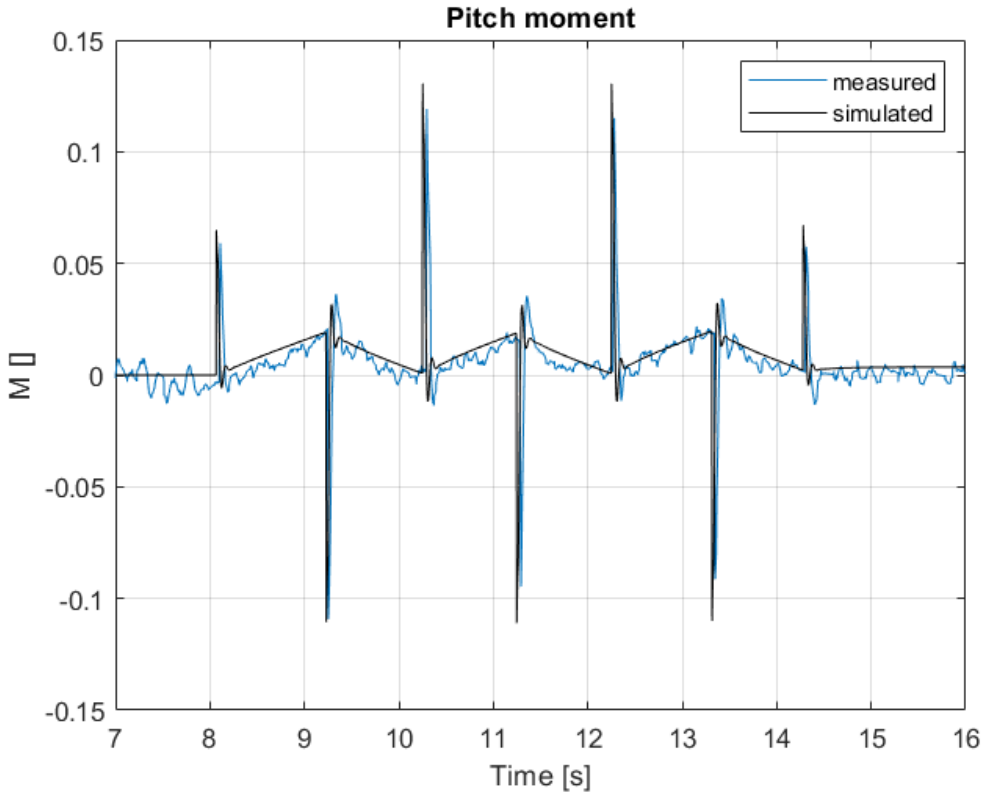


Session 2 - Angular rate control

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

Hands-on task:

Expected response of a control system with $\omega_c = 30\ rad/s$



Session 2 - Angular rate control

Final considerations:

- What would happen if we augment the sampling period?

- What would be the largest sampling time to guarantee stability and robustness of the control loop.

Extra Task: repeat the controller synthesis, starting from different requirements:

- Low bandwidth controller: $\omega_c^d = 10\ rad/s$
- High bandwidth controller: $\omega_c^d = 40\ rad/s$

This is the end of the activity.

POLITECNICO MILANO 1863

DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

M.Sc. AERONAUTICAL/SPACE ENGINEERING

051401 - DIGITAL CONTROL TECHNOLOGY FOR AERONAUTICS (DCTA)

Academic year 2025/2026

Prof. Fredy Ruiz

Assistant Alessandro Del Duca

Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

December 2025

Laboratory sessions

The experimental activity is based on the ANT-X platform, available at the Aerospace Systems & Control Laboratory.

It is divided into four sessions:

1. 1DOF modeling, November 20
2. Attitude Control, November 27
3. Cascade Control, December 4
4. Position Control, December 11

https://www.antx.it/

General Information

Cascade Attitude Control: Using discretization techniques, students develop a velocity–position cascade controller for attitude regulation. The impact of bandwidth separation and discretization methods on system performance is analyzed.

Preliminary work - Angular rate dynamics identification

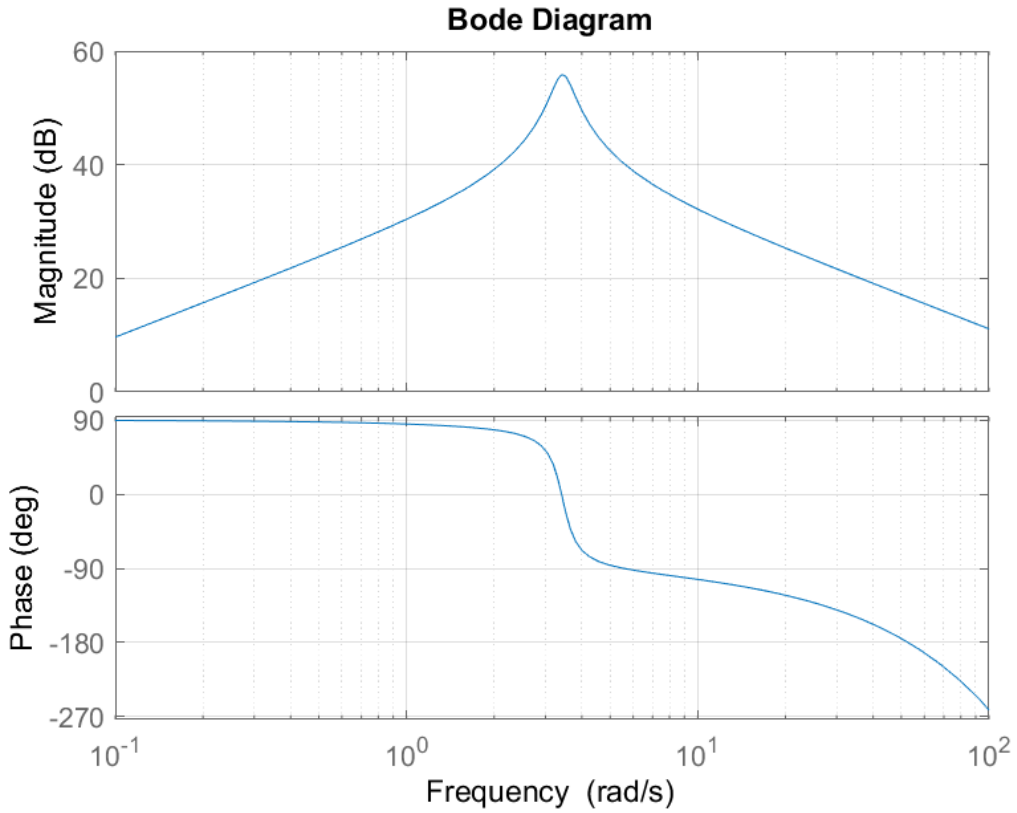
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Analysis of the identified model

$$G_q(s) = \frac{\mu s}{s^2 + 2\xi\omega_n s + \omega_n^2} e^{-s\tau}$$

Using the estimated parameters plot the frequency response of the model.

It must look like the one on the right.



Expected parameters:

$\omega_n = 3.4 \frac{rad}{s}$,

$\xi = 0.085$,

$\mu = 360$

$\tau = 20\ ms.$

Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Objective

Starting from the speed control loop designed in Session 2, tune and implement a P controller to ensure closed-loop stability and achieve the desired performance for the pitch dynamics.

Methodology

- P design by loop-shaping in continuous-time
- Digital implementation

Outline of the lecture

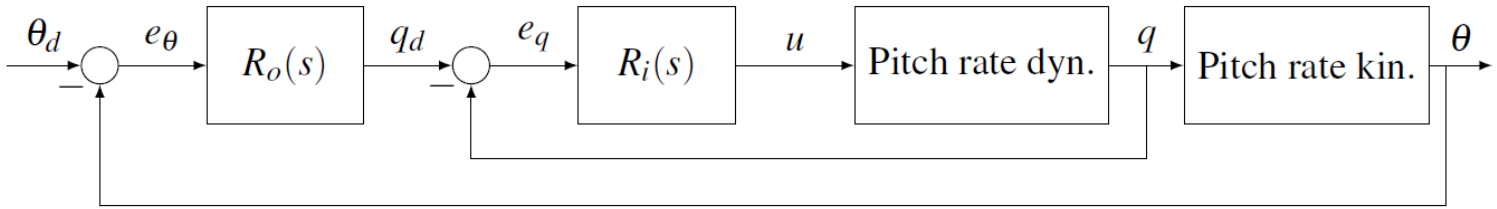
- Tuning of the attitude loop
- Digital implementation
- Experiment setup, data collection, and analysis of the results.

Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

We recall the structure of the overall control scheme:

- the outer loop computes a reference angular velocity q_d starting from the angular error $e_\theta = \theta_d - \theta$;
- the inner loop computes the torque u required to track the reference angular velocity q_d.



- The inner loop controller $R_i(s)$ was designed in Laboratory Session 2.
- The objective of the session is to design the controller $R_o(s)$ such that the overall pitch loop is asymptotically stable and has desired tracking performance.
- Since the inner loop has a sufficiently fast closed-loop dynamics with respect to the desired performance from θ_d to θ, the design of $R_o(s)$ will be carried out assuming that the transfer function from q_d to q is a static gain (frequency separation approach).

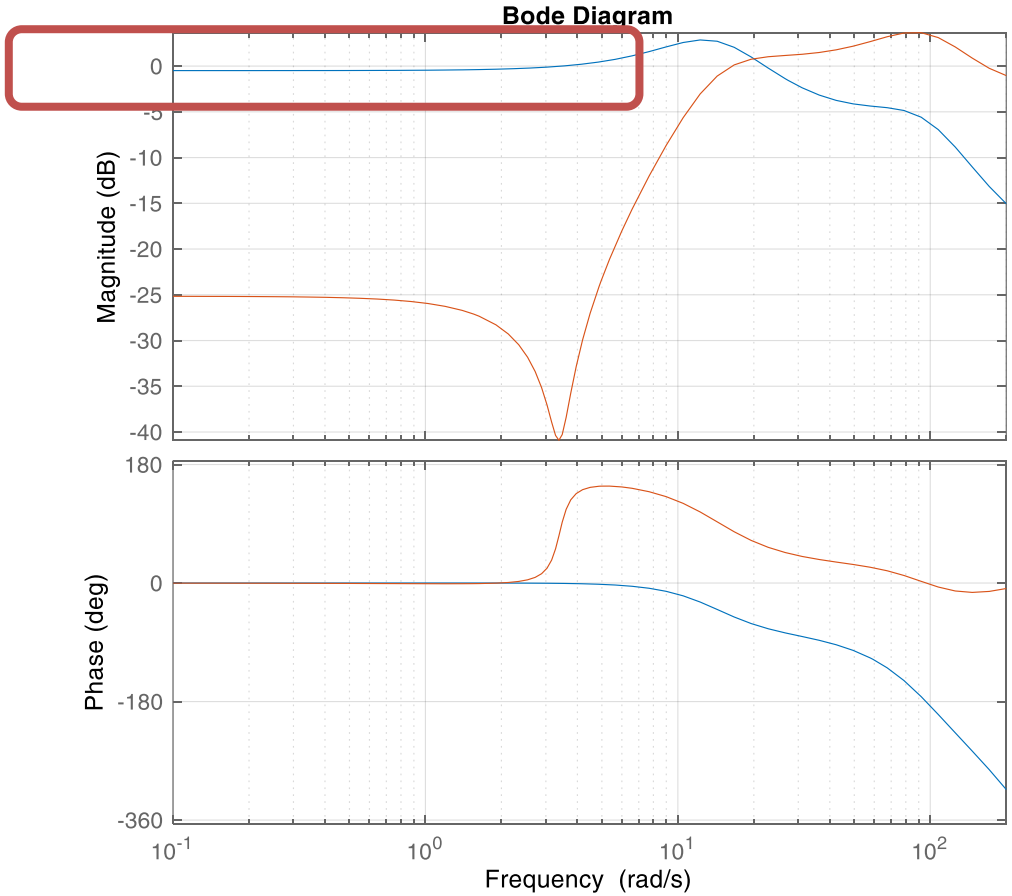
Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Attitude control: recap on angular rate control tuning

The Complementary sensitivity of the inner loop ($T_q(s)$) obtained with the simplified model used in the tuning of the angular rate controller is of the form:

NOTE: The inner loop is of type 0.
The DC gain is not unitary!



Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Control design requirements

1. Maximize the crossover frequency ω_c^o of the outer loop transfer function for good tracking performance;
2. Ensure frequency separation between the inner and outer loop: $\omega_c^i > 5\omega_c^o$, where ω_c^i is the crossover frequency of the inner loop transfer function.

Since the inner loop has a crossover frequency $\omega_c^i \in [15,\ 30]\frac{rad}{s}$, the maximum admissible crossover frequency to satisfy the second control requirement is

$$\omega_c^{o,d} = \frac{1}{5}\omega_c^i \in [3,\ 6] rad/s.$$

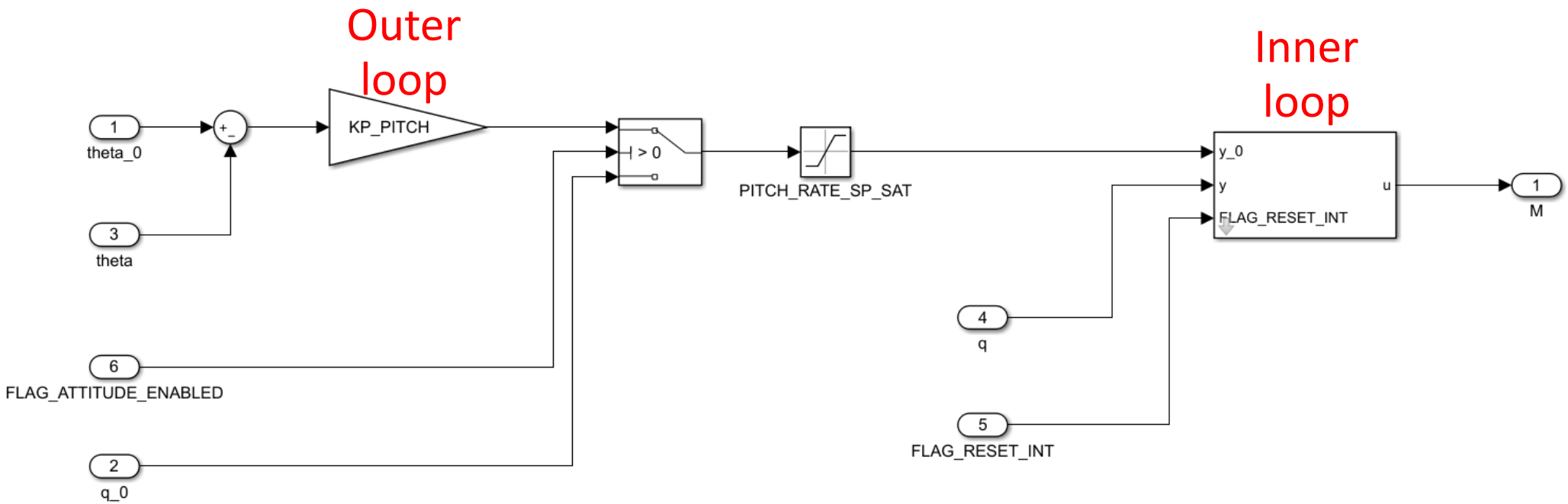
In this case, the outer loop control design reduces to the design of a control law $R_o(s)$ for the system

$$\theta = G_\theta(s) q_d$$

where q_d is the (virtual) control input.

Session 3 - Attitude control

Since $G_\theta(s) \approx \frac{1}{s}$, a viable solution (adopted in autopilots) is to employ a proportional controller for the outer loop.



The saturation function is introduced for safety reasons.
It avoids the application of large angular rate commands to the inner loop.

Session 3 - Attitude control

Following the loop shaping tuning approach, we start by computing the loop transfer function for the outer loop:

$$L_\theta(s) = R_\theta(s)G_\theta(s) = \frac{K_p^o \mu_q}{s}$$

To achieve the desired crossover frequency, the value of the gain can be obtained by assigning

$$\left|L_\theta\left(j\omega_c^{o,d}\right)\right| = 1$$

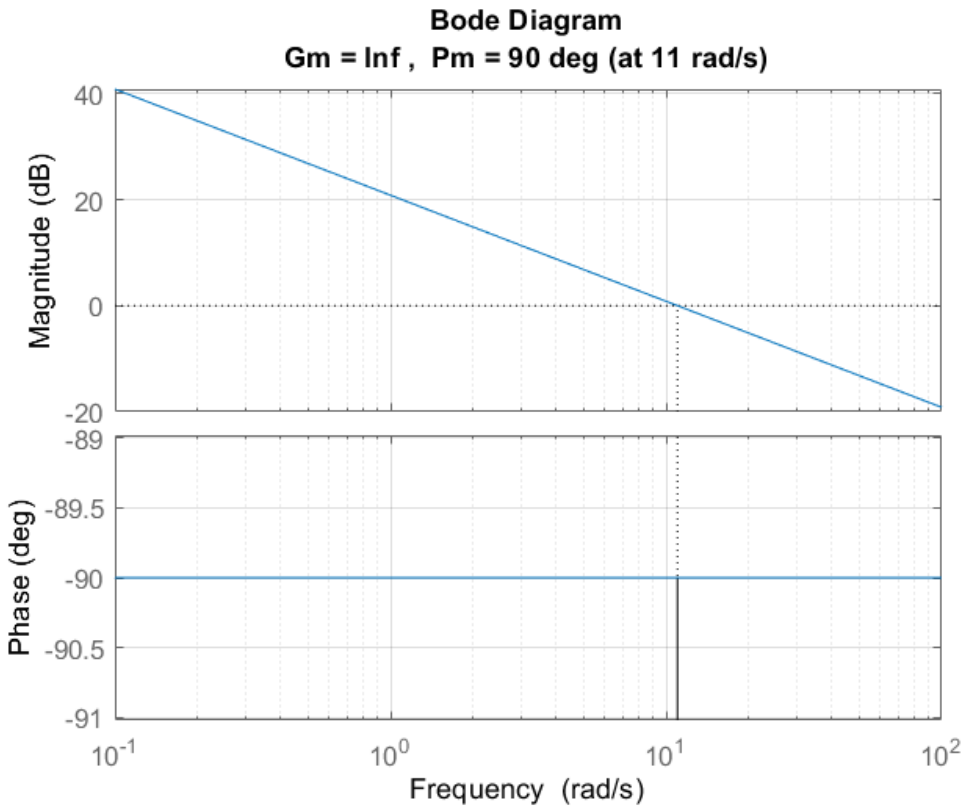
From which we get

$$\frac{K_p^o \mu_q}{\omega_c^{o,d}} = 1 \rightarrow K_p^o = \frac{\omega_c^{o,d}}{\mu_q}$$

Hence, the loop gain is set equal to the desired crossover frequency.

Session 3 - Attitude control

The IDEAL Loop transfer function used for control design (approximate $T_q(s) = \mu_q$).

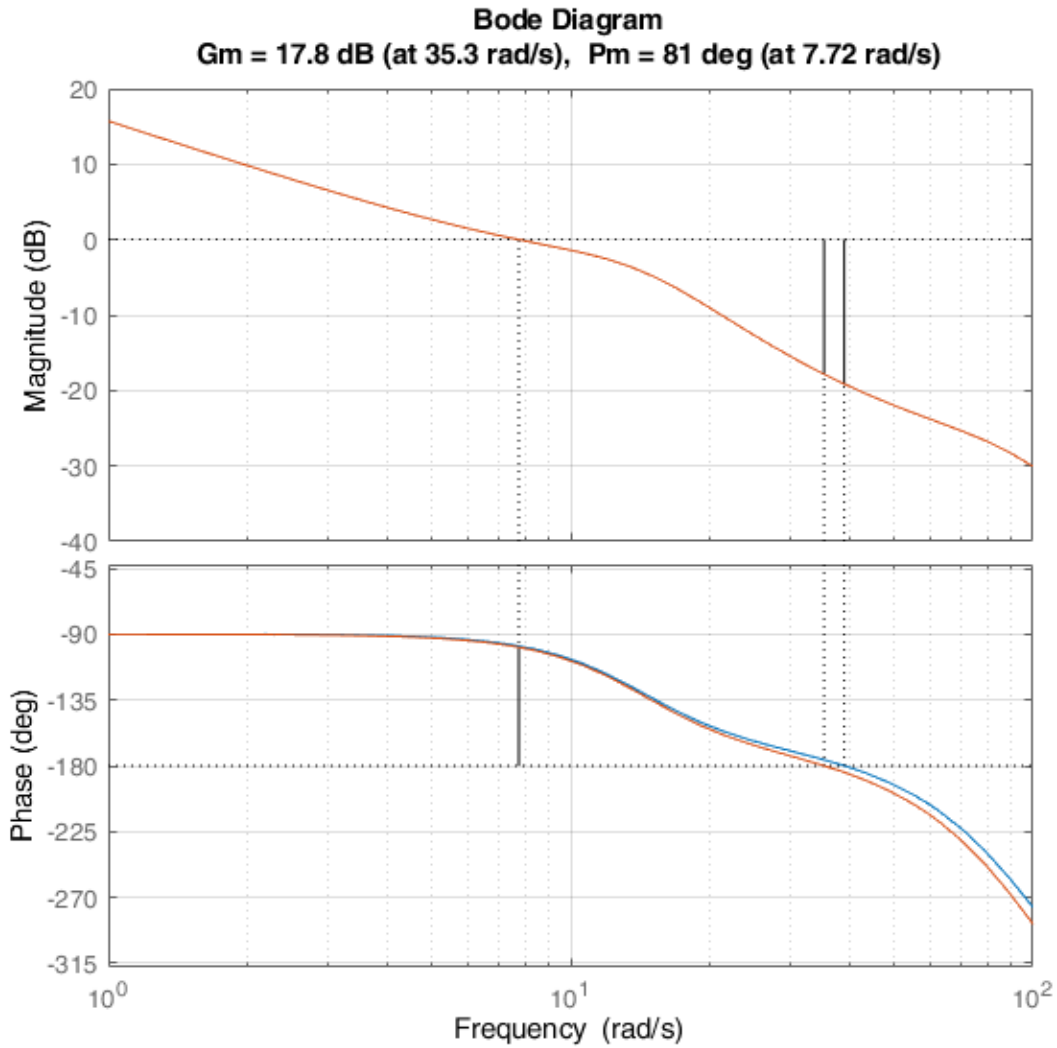


Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Verification of design requirements:

Compare the outer loop transfer function obtained considering the (full order) inner closed loop angular rate dynamics (including the identified model and delays) with the same loop transfer function, simplified by neglecting the angular rate dynamics (ideal, $T_q(s) = 1$).



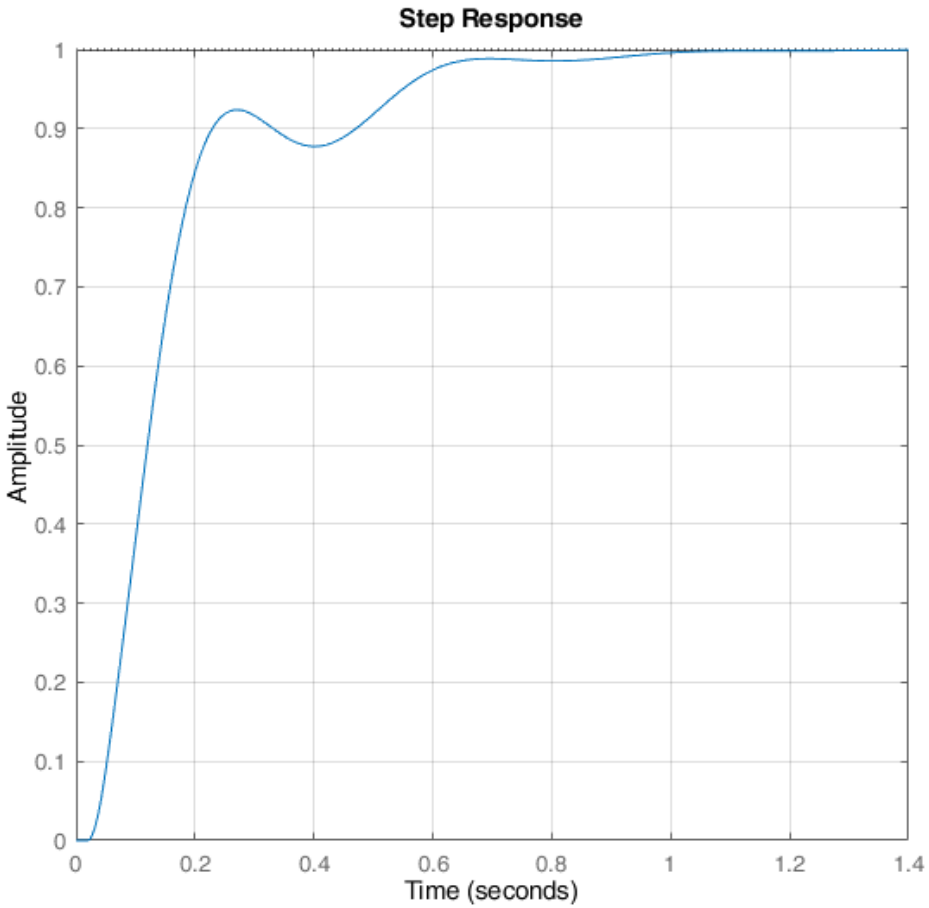
Session 3 - Attitude control

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

Verification of design requirements:

step response comparison

- Regardless of the simplifying assumptions used in the control design, the step response obtained with the simplified and identified model is comparable.



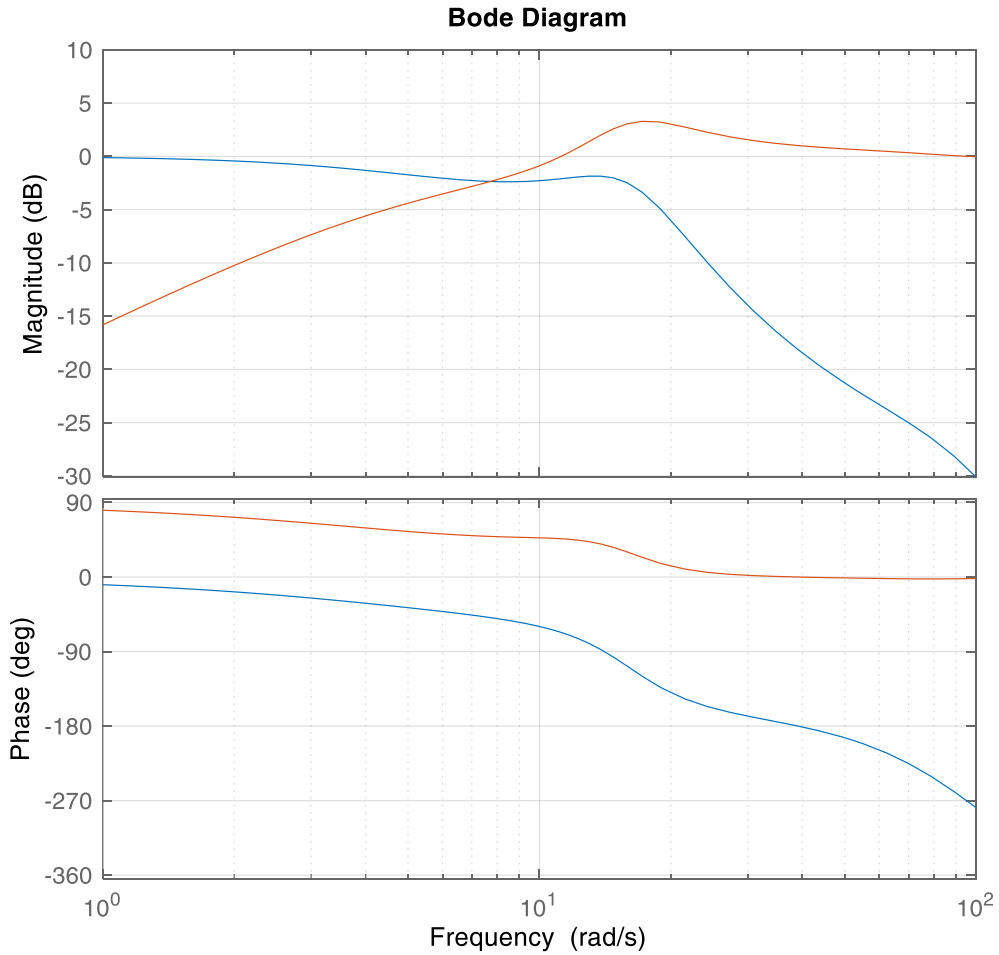
Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Verification of design requirements:

Frequency response comparison

Obtain the actual (complementary sensitivity) bandwidth and compare it with the desired one.



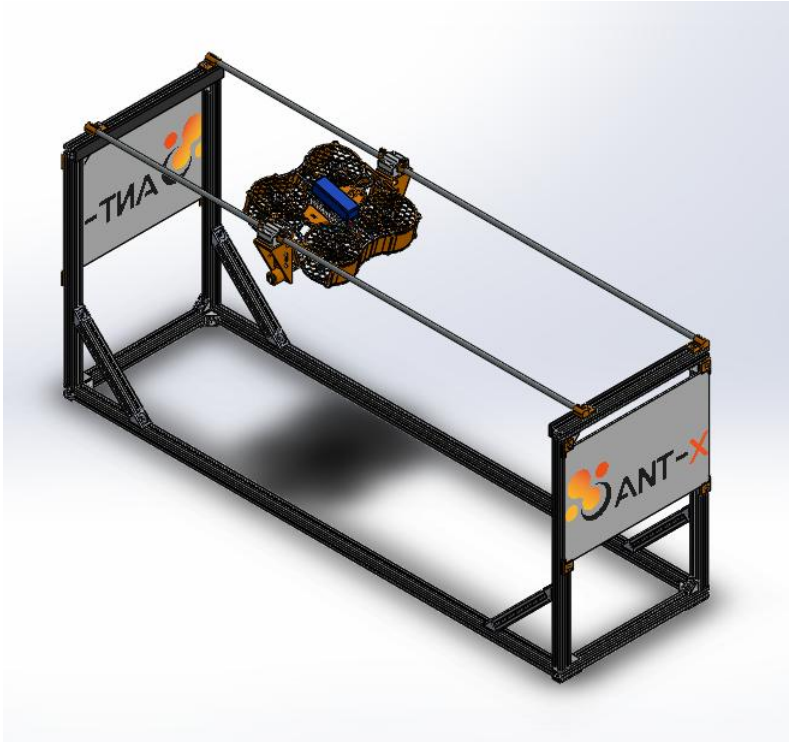
Session 3 - Attitude control

Hands-On test:

1. **Connect to the Wi-Fi network of the Drone**: run the Wi-Fi script located on the desktop of the PC.
2. Open Matlab
3. Go to the folder …\Antx\DroneCmd\DCTA\
4. Open the File

   S3_drone2dof_attitudeStep.m

It is the same script used in the first session!



The procedure to activate and control the drone is as follows:

5. Run the first three sections to connect to the drone within the ROS environment, up to **drone.connect();**

Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-On test:

The controller parameters are set using the QGroundControl application.

6. IN UBUNTU Run the script QGroundControl
7. Go to the vehicle setup menu located in the upper-left corner (the button with a gearbox)
8. Choose the parameters window in the lower-left corner (the gearbox symbol)
9. Adjust the corresponding gain in the controller based on the next table:

Parameter in QGC	Controller gain	Designed value	Admissible range
FC_PARAM_4	angular rate P gain		[0 – 0.12]
FC_PARAM_5	angular rate I gain		[0 – 0.5]
FC_PARAM_6	angular rate D gain		[0 – 0.003]
FC_PARAM_11	attitude P gain		[0 –15]

References: https://ant-x.gitlab.io/documentation_2dofdrone/_chapters/operations.html#modifying-controller-parameters-from-qgroundcontrol

Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Once you have updated the controller gains, the drone must be started by activating the **safety switch**.

10. Continue the execution of the script by activating the control system,

```
% ARM
drone.arm();
```

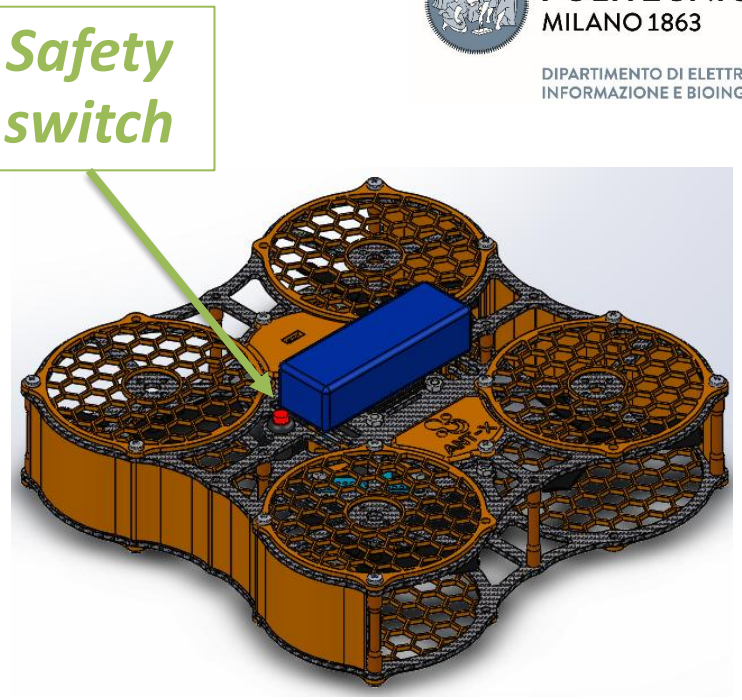
- This script sets the drone in attitude tracking mode.

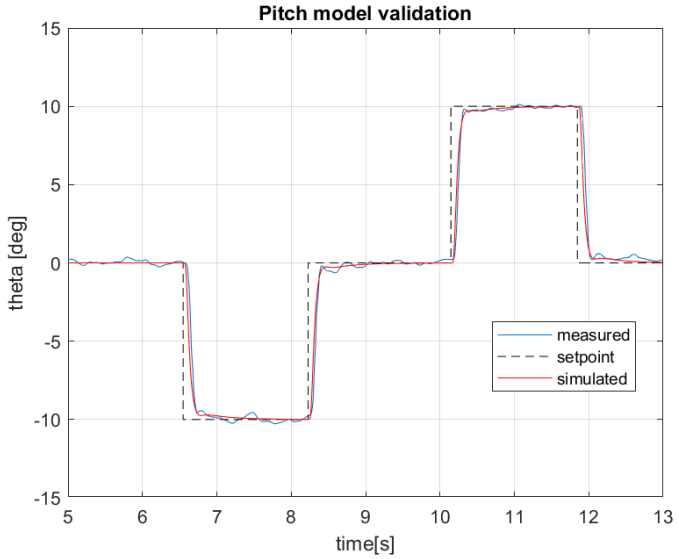
- A sequence of angular position set points is sent to the FCC.

- In the first test, set the amplitude of the steps to +/-10°

```
%% define sequence of steps
theta0_val = [0; -10; 0; 10; 0; 0]'/180*pi;
dt = .1;
T_period = 2;
```

- Once you have set/verified the amplitude of the set point, run the next three sections and observe the movement of the drone.



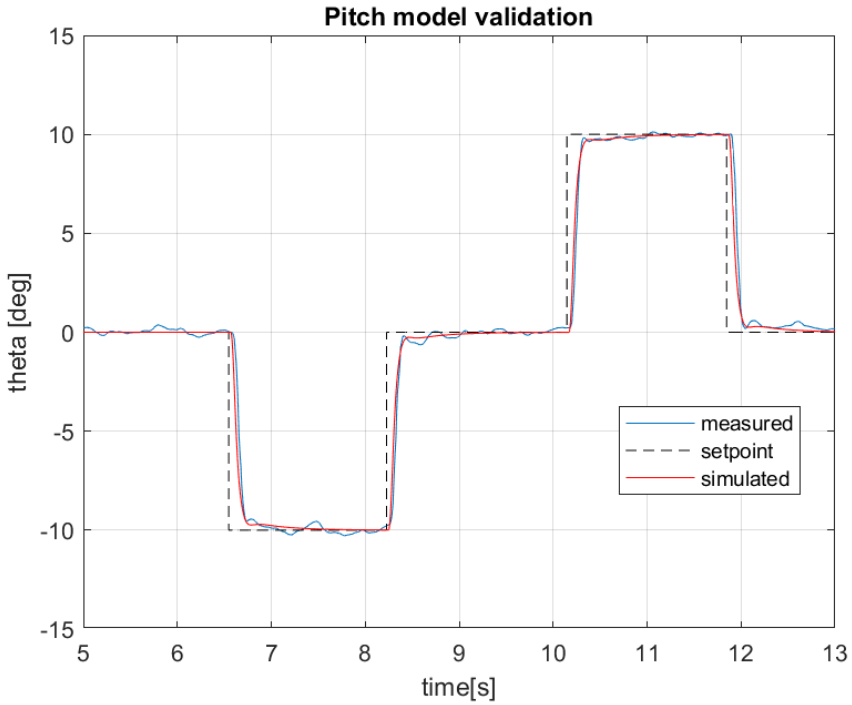


Part 1 - Familiarization with the platform ANT-X

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

- Repeat the experiment with a larger amplitude of the reference signal of +/- 25°

```
%% define sequence of steps
theta0_val = [0; -25; 0; 25; 0; 0]'/180*pi;
dt = .1;
T_period = 2;
```



- Once you have set/verified the amplitude of the set point, run again the *connect->run->disconnect* sections and observe the movement of the drone.

- Finally, extract the experiment data for post-processing. Save the *.mat files to a USB stick for analysis on your own PC.

Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Post-processing:

- In your PC, save and load the *.mat file in Matlab.
- For this experiment we are interested in the following variables:
  - theta: pitch angle (rad)
  - theta0: pitch reference (rad)
  - q: pitch rate (rad/s)
  - M: normalized torque (dimensionless)

- Plot the step response of the pitch control system and characterize the performance of the loop in terms of standard indexes
  - Rise time
  - Settling time
  - Overshoot
  - Oscillation period
  - Input peak amplitude and saturation events
- Compare the performance of your controller with the response obtained in simulation and with the nominal controller evaluated in Session 1.

This is the end of the activity.

POLITECNICO MILANO 1863

DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

M.Sc. AERONAUTICAL/SPACE ENGINEERING

051401 - DIGITAL CONTROL TECHNOLOGY FOR AERONAUTICS (DCTA)

Academic year 2025/2026

Prof. Fredy Ruiz

Assistant Alessandro Del Duca

Politecnico di Milano

Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB)

December 2025

Laboratory sessions

The experimental activity is based on the ANT-X platform, available at the Aerospace Systems & Control Laboratory.

It is divided into four sessions:

1. 1DOF modeling, November 20
2. Attitude Control, November 27
3. Cascade Control, December 4
4. Position Control, December 11

https://www.antx.it/

Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Objective

Starting from the closed-loop attitude control system developed in the previous sessions, tune and implement a cascade velocity/position controller for the drone.

Methodology

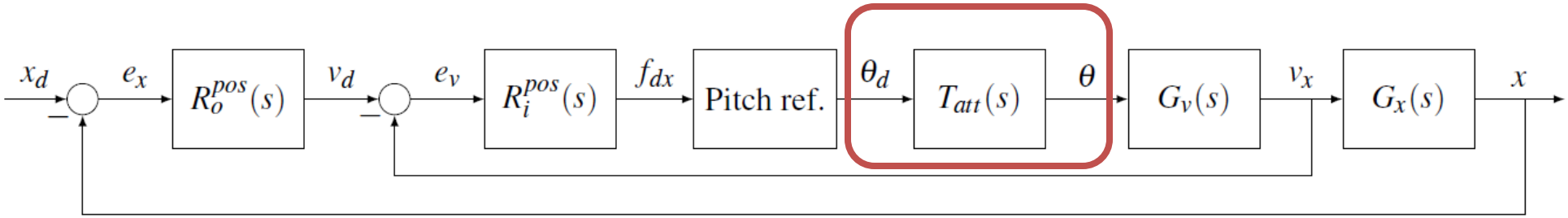
- PI design by loop-shaping in continuous-time
- Digital implementation

Outline of the lecture

- PID controller structure
- Tuning through loop-shaping
- Digital implementation
- Experiments setup, data collection and analysis of the results.

Session 4 – Position control

In this session, we consider the same approach developed for the attitude dynamics and design a cascade controller also for the position dynamics.



The outer loop controller $R_o^{pos}(s)$ provides a velocity setpoint v_d to the inner loop controller $R_i(s)$.

We employ a P/PI architecture:

- $R_o^{pos}(s) = K_p^o$
- $R_i^{pos}(s) = K_p^i + \frac{K_i^i}{s}$

Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The longitudinal dynamics of the quadrotor are described as:

$$\dot{x} = v_x$$
$$m\dot{v}_x = -T_c\theta + f_{xe}$$
$$\dot{\theta} = q$$
$$J_\theta \dot{q} = M_c + M_e$$

- x, v_x are the position and velocity along the x-axis;
- θ, q are the pitch angle and rate;
- f_{xe} is the exogenous force acting on the drone (aerodynamic+ friction due to bearings);
- M_e is the exogenous torque;
- M_c is the pitch control moment;
- T_c is the control thrust.

➢ The longitudinal dynamics have a **cascade** structure in which the pitch angle influences the translational dynamics.

➢ By controlling the attitude one can *indirectly* control the translation (cascade control design for controllable underactuated systems).

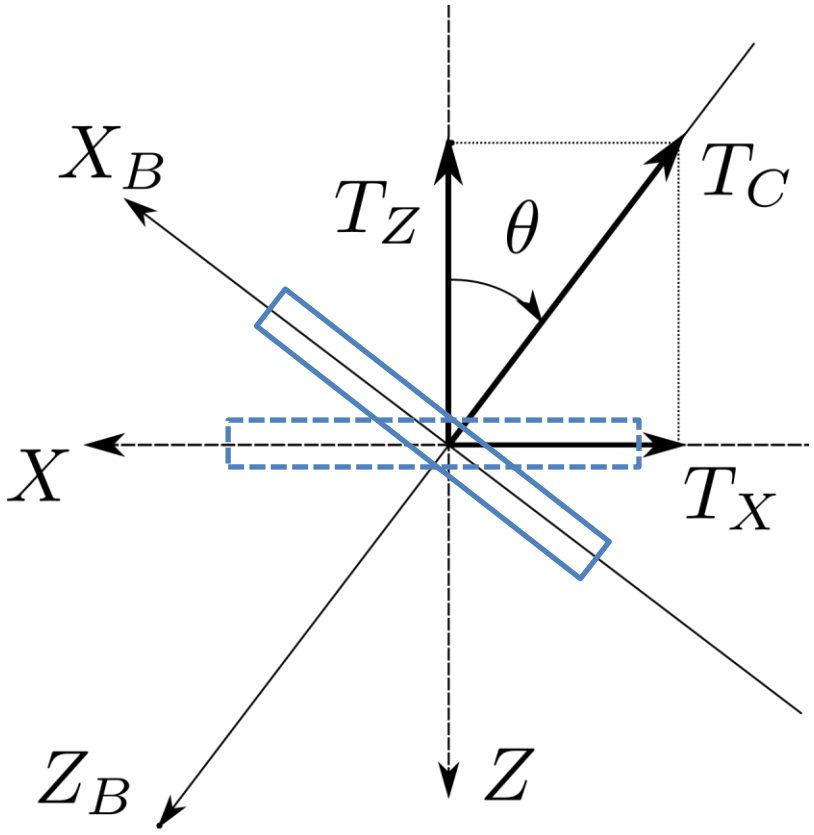
Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

The control force (directed along the rotor axis) is projected onto the longitudinal direction by tilting the drone frame.

Recap on notation:

- $X,\ Z$: inertial reference frame (forward, down)
- X_B, Z_B: body frame
- θ positive nose-up

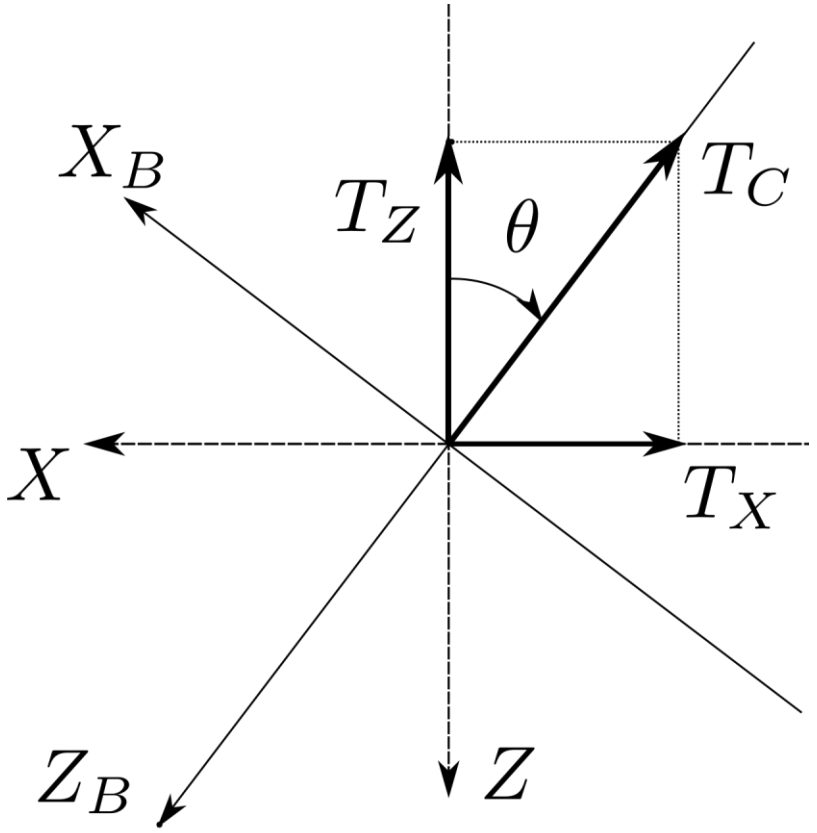


Session 4 – Position control

The control force (directed along the rotor axis) is projected onto the longitudinal direction by tilting the drone frame.

The x and z components of the thrust are:

$$T_x = -T_c \sin\theta$$
$$T_z = -T_c \cos\theta$$



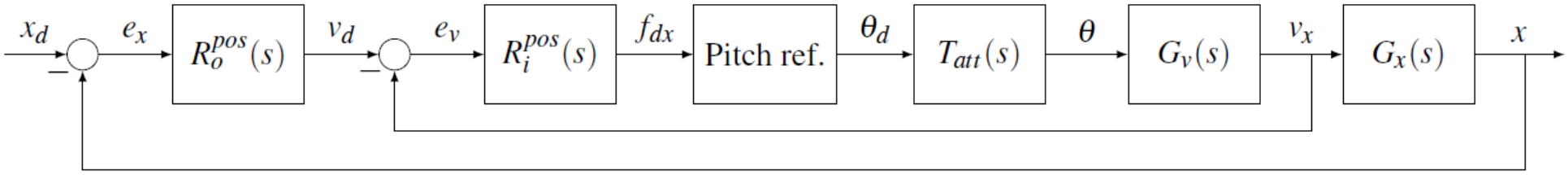
Since the actual thrust applied to the drone is related to the control thrust T_c by a scaling factor

$$T_c = \eta_T T_c^d,$$

the x-component of the force becomes

$$T_x = -\eta_T T_c^d \sin\theta \simeq -\eta_T T_c^d \theta$$

Session 4 – Position control



The output of the position controller (f_{dx}) (physically a force), is converted into a reference pitch angle

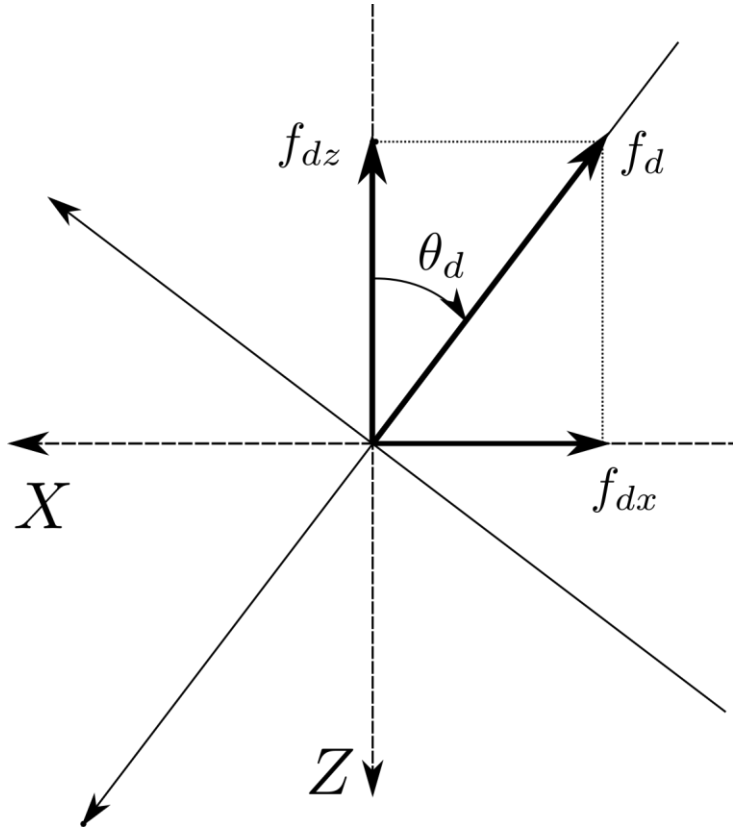
$$\theta_d = \tan^{-1}\left(\frac{f_{dx}}{f_{dz}}\right)$$

Linearization yields

$$\theta_d = \frac{f_{dx}}{f_{dz}}$$

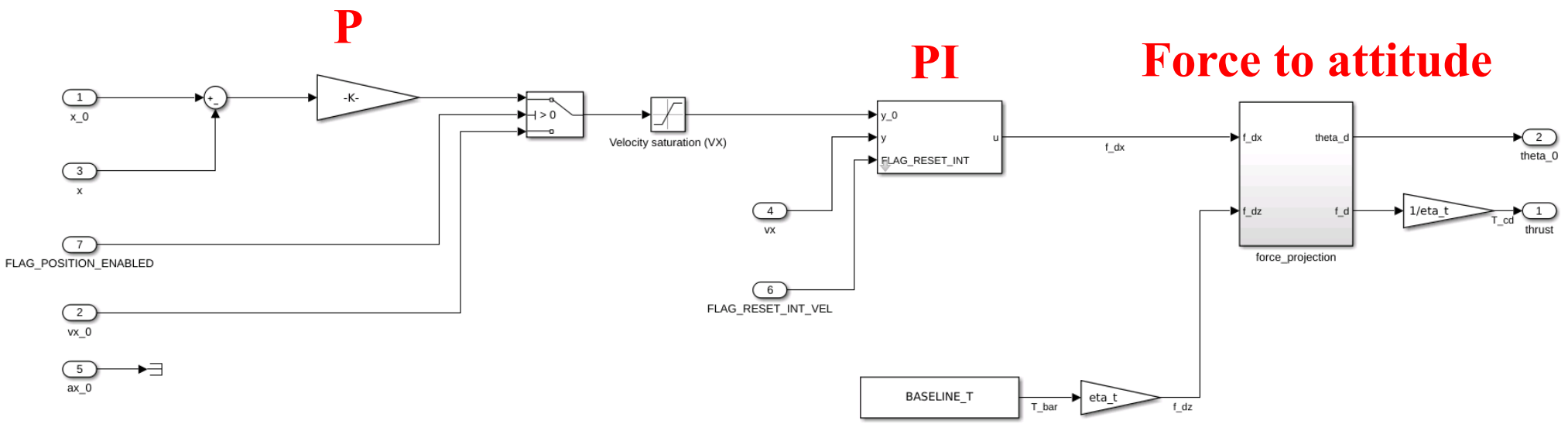
$$\theta_d = -\frac{f_{dx}}{\overline{T}\eta_T}$$

Note that in practical implementations, the exact (nonlinear) expression to extract the pitch angle is used.



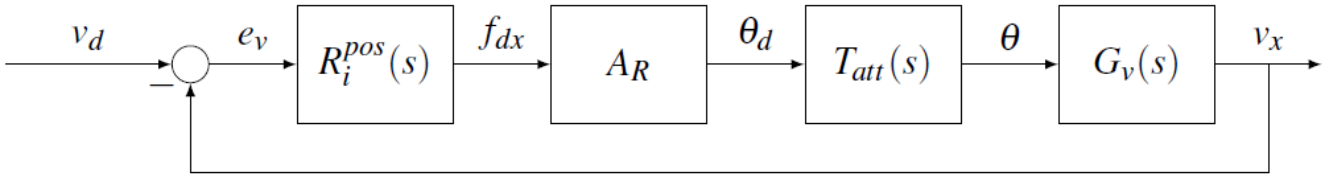
Session 4 – Position control

Cascaded position and velocity control architecture: Simulink implementation



Session 4 – Position control

Velocity control loop



Assumptions:

- Neglect the attitude dynamics, assuming it is much faster than velocity dynamics

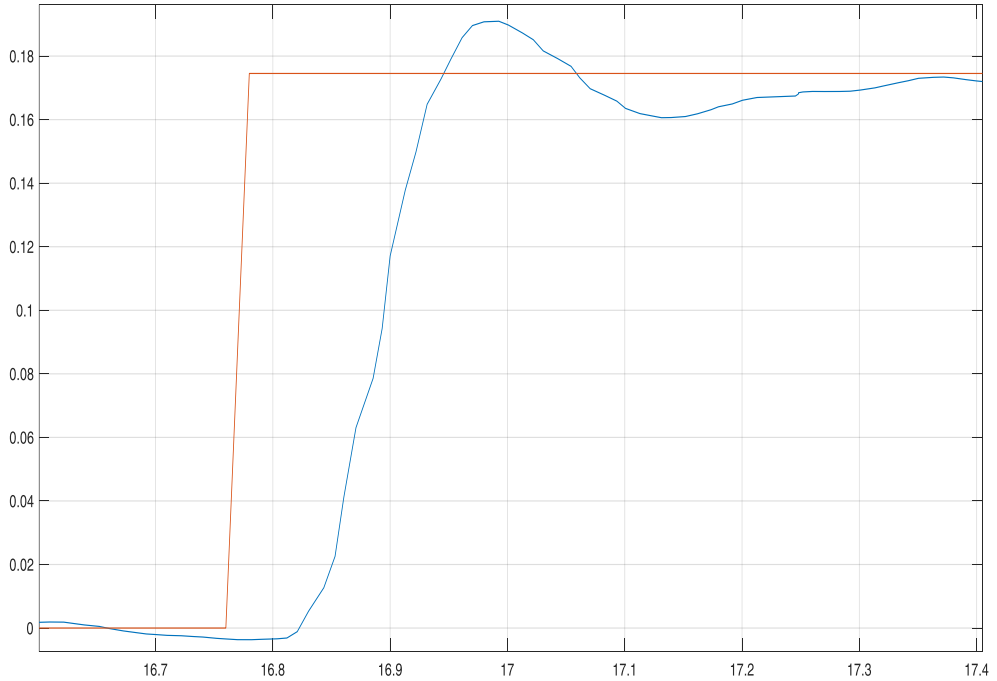
$$T_{att}(s) \simeq 1$$

Design choices:

- Choose $K_d^i = 0$ for simplicity

$$R_i^{pos}(s) = K_p^i + \frac{K_i^i}{s}$$

Closed-loop attitude step response (settling time ≈ 0.63s)



Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

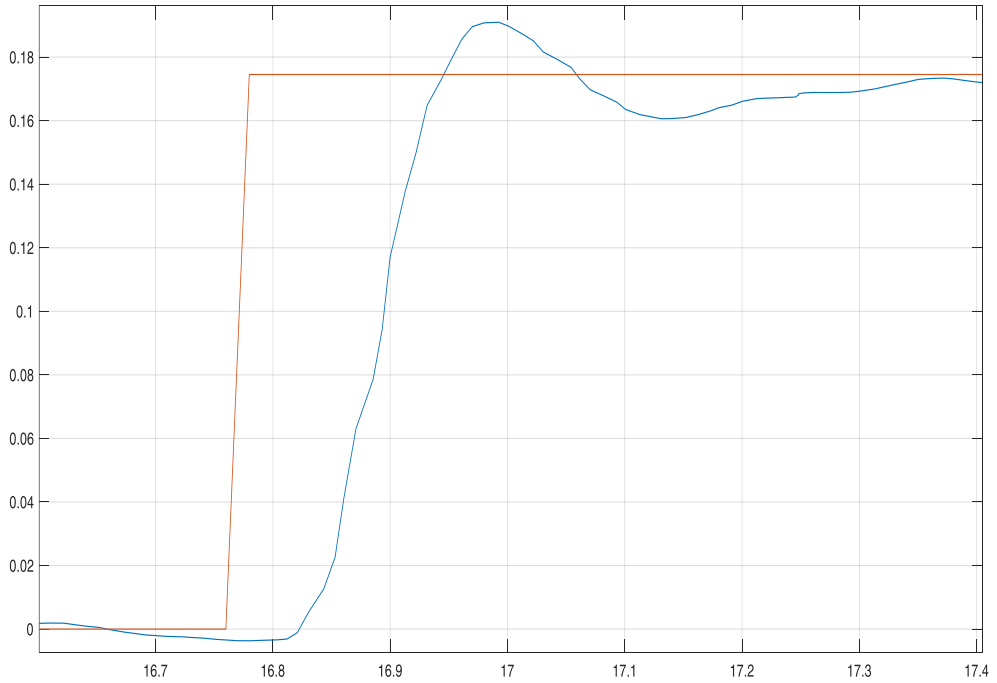
Frequency separation: attitude control bandwidth must be sufficiently larger than velocity control bandwidth.

In turn, velocity control bandwidth must be sufficiently larger than position control bandwidth

Requirements:

- Velocity control loop: $\omega_{c,vel}^{d} \simeq 3.5\ rad/s$

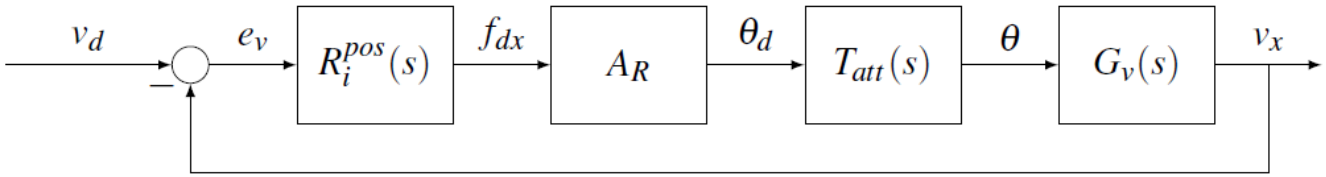
Closed-loop attitude step response (settling time ≈ 0.63s)



Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Velocity control loop



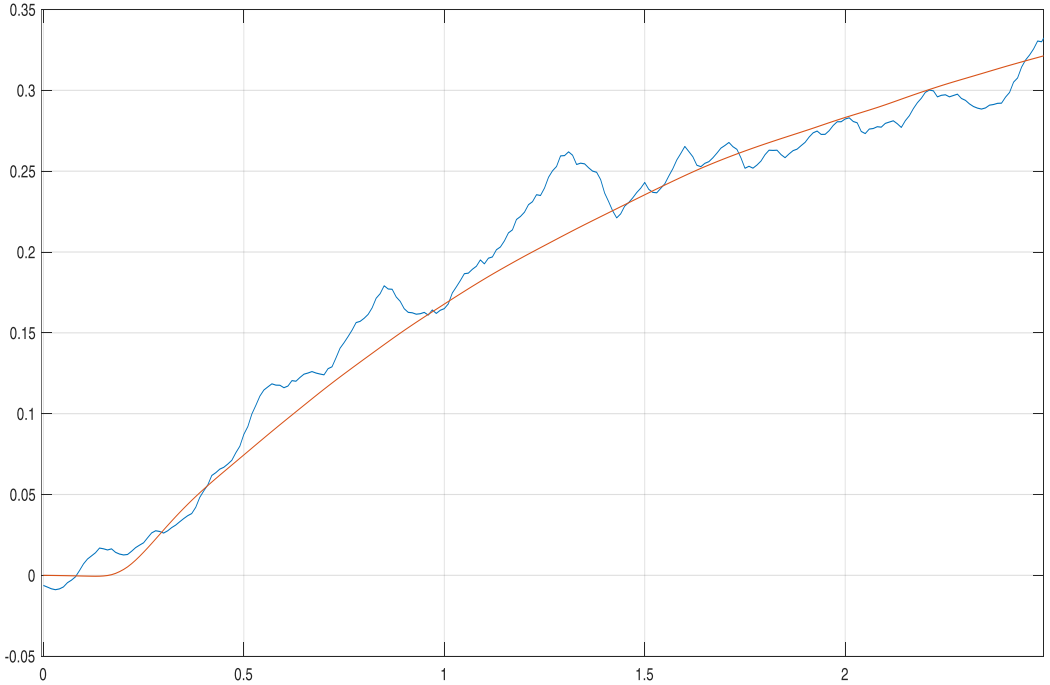
$R_i^{pos}(s)$ velocity controller transfer function
A_R attitude reference generator
$T_{att}(s)$ closed-loop attitude dynamics
$G_v(s)$ velocity dynamics

$$R_i^{pos}(s) = K_p^i + \frac{K_i^i}{s}$$

$$A_R = \frac{1}{f_{dz}} = -\frac{1}{\overline{T}\eta_T}$$

$$G_v(s) = -\frac{\overline{k}}{\tau s + 1} \cong -\frac{2.4}{1.5385\ s + 1}$$

Velocity response to a negative attitude step



Session 4 – Position control

Velocity control loop

Rearrange $R_i^{pos}(s)$ as

$$R_i^{pos}(s) = K_p^i + \frac{K_i^i}{s} = K_p^i \frac{s+z}{s}$$

with $z = \frac{K_i^i}{K_p^i}$

Hint: Tune the controller to cancel the plant pole with z and tune K_p^i to set the bandwidth.

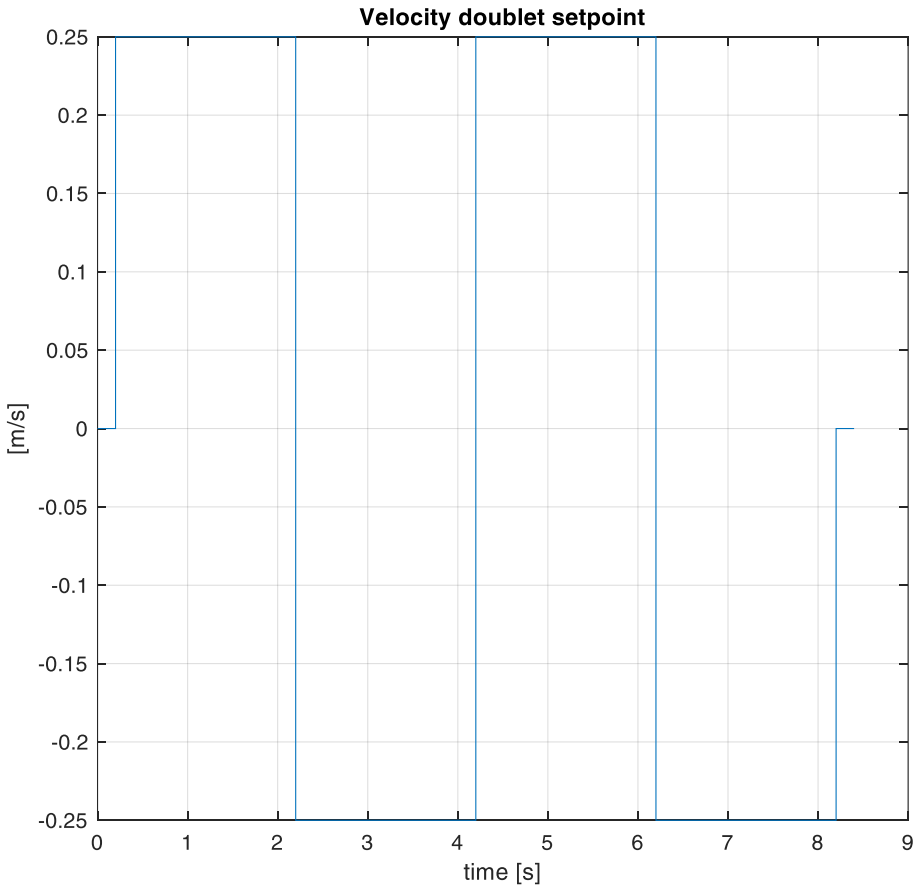
Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

- Evaluate the robustness of your controller verifying the phase and gain margins including the delay of the ZOH and the attitude loop dynamics.

- Test the performance of the control loop using a doublet reference signal

- Evaluate your controller in simulation using the proposed doublet reference signal.



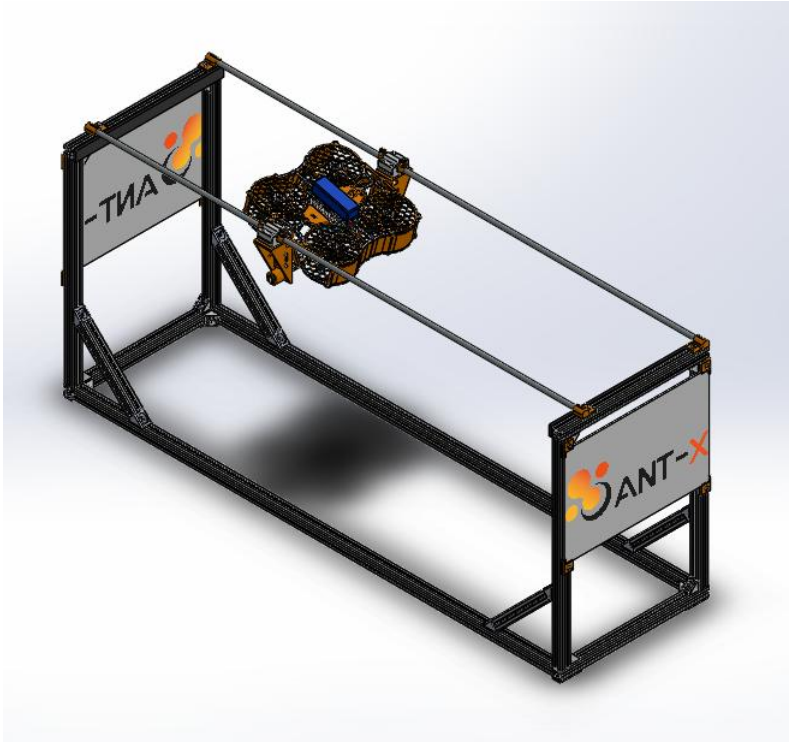
Session 4 – Position control

Hands-On test:

1. **Connect to the Wi-Fi network of the Drone**: run the Wi-Fi script located on the desktop of the PC.
2. Open Matlab
3. Go to the folder …\Antx\DroneCmd\TSCA\
4. Open the File *S4_drone2dof_velocityDoublet.m*

The procedure to activate and control the drone is as follows:

5. Run the first three sections to connect to the drone within the ROS environment, up to **drone.connect();**



Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-On test:

6. Open QGroundControl
7. Go to the vehicle setup menu located in the upper-left corner (the button with a gearbox)
8. Choose the parameters window in the lower-left corner (the gearbox symbol)
9. Adjust the corresponding gain in the controller based on the next table:

Parameter in QGC	Controller gain	Expected value	Admissible range
FC_PARAM_13	Velocity P gain		[0 – 4]
FC_PARAM_14	Velocity I gain		[0 – 4]
FC_PARAM_15	Velocity D gain	Not USED	[–]

References: https://ant-x.gitlab.io/documentation_2dofdrone/_chapters/operations.html#modifying-controller-parameters-from-qgroundcontrol

Session 4 – Position control

Once you have updated the controller gains, the drone must be started by activating the **safety switch**.

10. Continue the execution of the script by activating the control system,

```
% ARM
drone.arm();
```

11. Move the drone to the negative end of the rail (position control mode)

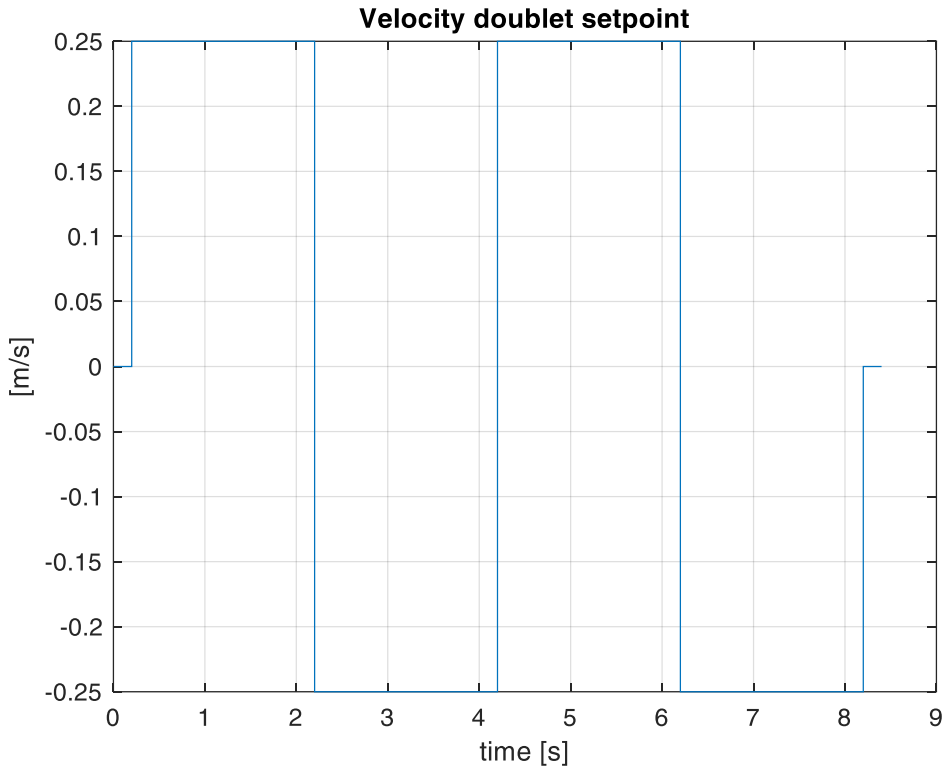
```
%% Send position setpoint
starting_point = -0.2; % m
```

12. Then, this script sets the drone in linear speed tracking mode, with a doublet as a reference signal

```
T_doublet = 2; % s
A = .25; % m/s
```

13. Run the disconnect section

```
%% Close connection
drone.disconnect();
```



Session 3 - Attitude control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Finally, extract the experiment data for post-processing. Save the *.mat files to a USB stick for analysis on your own PC.

Post-processing:

- In your PC, save and load the *.mat file in Matlab.
- For this experiment, we are interested in the following variables:
  - v: velocity (m/s)
  - v0: velocity reference(m/s)
  - theta: pitch angle (rad)
  - theta0: pitch reference (rad)

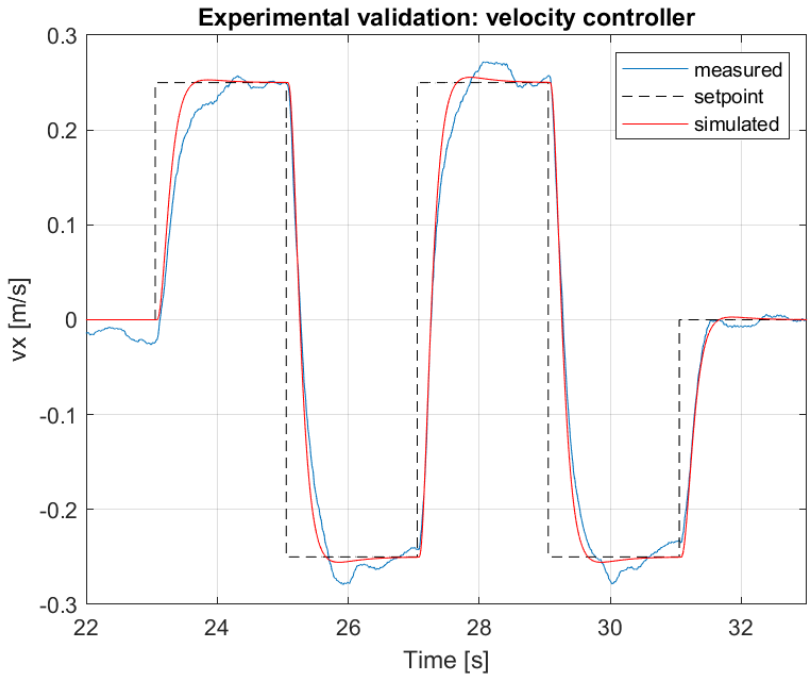
- Plot the step response of the velocity control system and characterize the performance of the loop in terms of standard indices
  - Rise time
  - Settling time
  - Overshoot
  - Oscillation period
  - pitch reference amplitude and saturation events

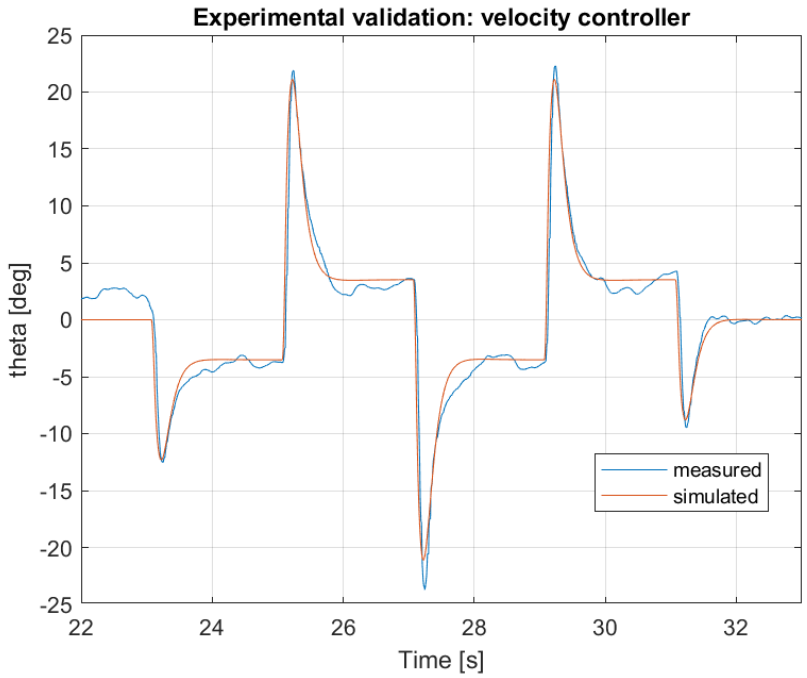
Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-on task:

Expected response of the speed control loop with $\omega_c = 4.2\ rad/s$





Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Objective

Tune and implement a P controller to guarantee closed-loop stability and desired performance for the position dynamics, using the speed PI controller designed previously.

Methodology

- P design by loop-shaping in continuous-time
- Digital implementation

Outline of the lecture

- Tuning of integral loop
- Digital implementation
- Experiment setup, data collection, and analysis of the results.

Session 4 – Position control

Control design requirements

1. Maximize the crossover frequency ω_c^o of the outer loop transfer function for good tracking performance;
2. Ensure frequency separation between the inner and outer loop: $\omega_c^i > 5\omega_c^o$, with ω_c^i the crossover frequency of the inner loop transfer function.

Since the inner loop has a crossover frequency $\omega_c^i \simeq 3.5 \frac{rad}{s}$, the maximum admissible crossover frequency to satisfy the second control requirement is

$$\omega_c^{o,d} = \frac{1}{5}\omega_c^i \approx 0.7\ rad/s.$$

In this case, the outer loop control design reduces to the design of a control law $R_o(s)$ for the system

$$x(s) = G_x(s)v_d$$

with v_d the (virtual) control input.

Session 4 – Position control

Following the loop shaping tuning approach, we start by computing the loop transfer function for the outer loop:

$$L_x(s) = R_x(s)G_x(s) = \frac{K_p^o}{s}$$

To achieve the desired crossover frequency, the value of the gain can be obtained by assigning

$$\left|L_\theta\left(j\omega_c^{o,d}\right)\right| = 1$$

From which we get

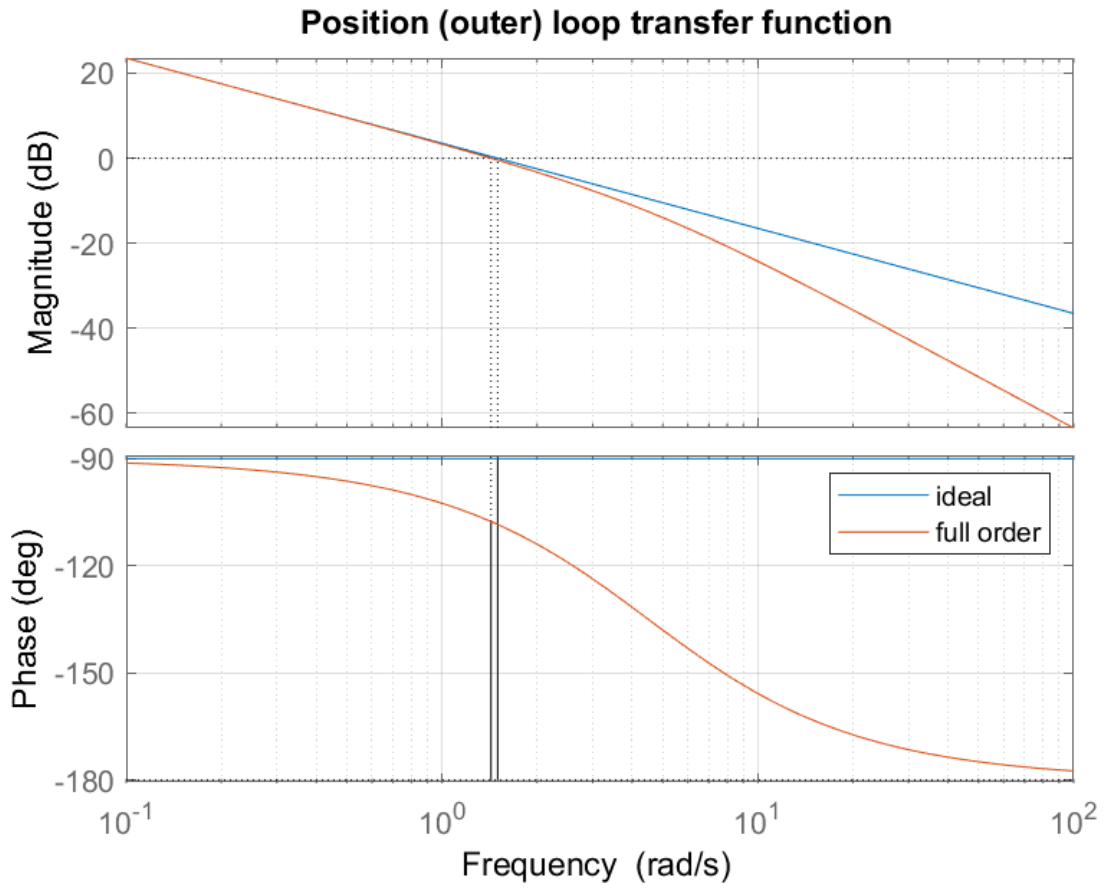
$$\frac{K_p^o}{\omega_c^{o,d}} = 1 \rightarrow K_p^o = \omega_c^{o,d}$$

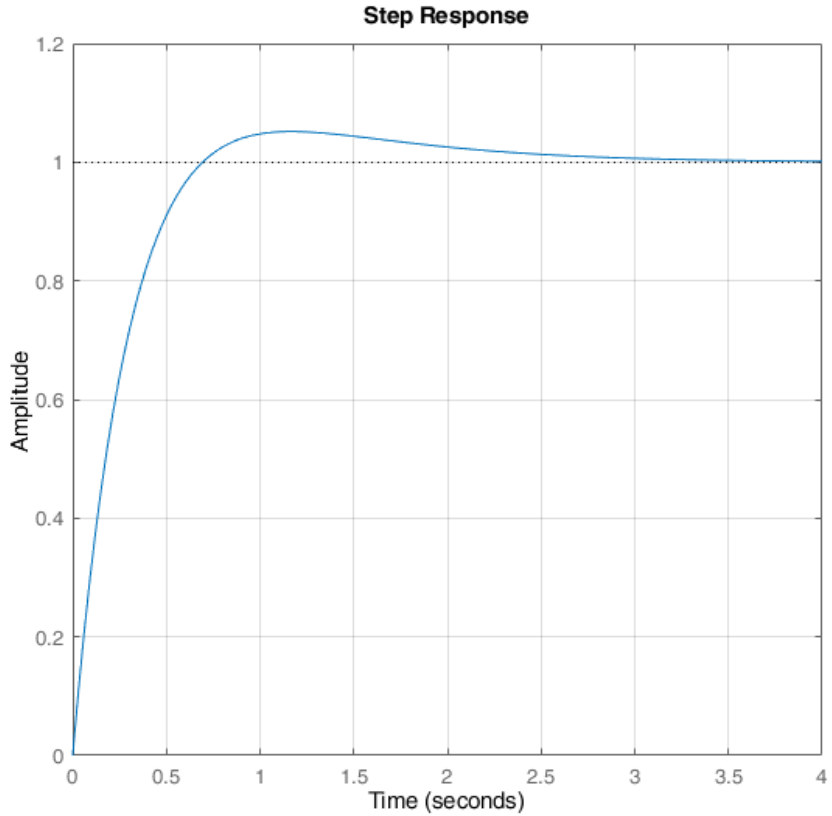
Hence, the loop gain is set equal to the desired crossover frequency.

Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Ideal and full order loop frequency response and full order closed loop step response.





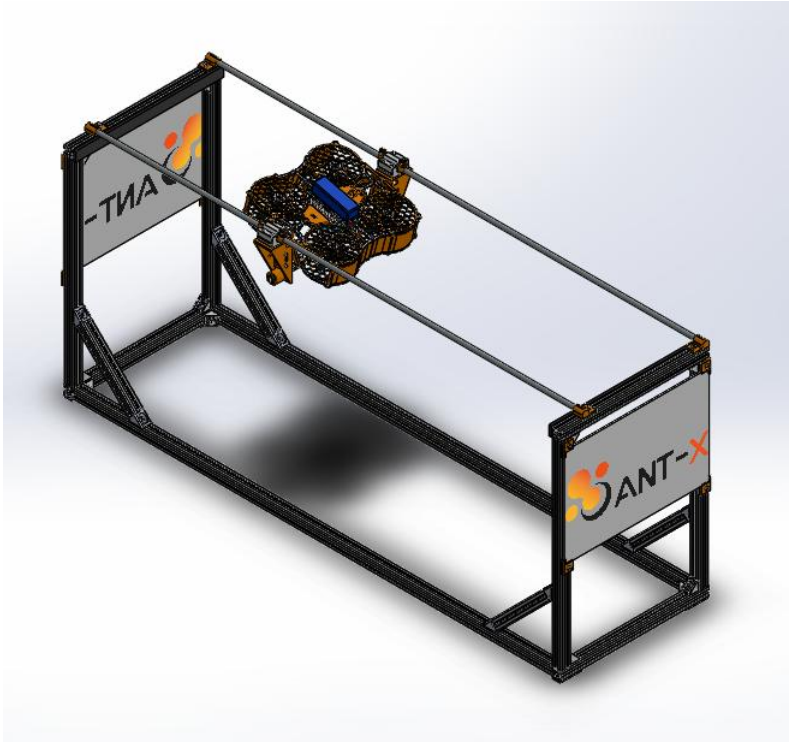
Session 4 – Position control

Hands-On test:

1. **Connect to the Wi-Fi network of the Drone**: run the Wi-Fi script located on the desktop of the PC.
2. Open Matlab
3. Go to the folder …\Antx\DroneCmd\TSCA\
4. Open the File *S4_drone2dof_positionStep_auto.m*

The procedure to activate and control the drone is as follows:

5. Run the first three sections to connect to the drone within the ROS environment, up to **drone.connect();**



Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-On test:

The controller parameters are set using the QGroundControl application.

6. IN UBUNTU Go to …\Antx\Antxtools\ and run the script QGroundControl
7. Go to the vehicle setup menu located in the upper-left corner (the button with a gearbox)
8. Choose the parameters window in the lower-left corner (the gearbox symbol)
9. Adjust the corresponding gain in the controller based on the next table:

Parameter in QGC	Controller gain	Expected value	Admissible range
FC_PARAM_13	Velocity P gain		[0 – 4]
FC_PARAM_14	Velocity I gain		[0 – 4]
FC_PARAM_15	Velocity D gain	Not USED	[–]
FC_PARAM_19	Position P gain		[0 – 2.5]

References: https://ant-x.gitlab.io/documentation_2dofdrone/_chapters/operations.html#modifying-controller-parameters-from-qgroundcontrol

Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

10. Continue the execution of the script, activating the control system applying positions setpoints in the interval [- 0.3, 0.3]m, for example,

```
%% define sequence of steps
x0_val = [0, 0.2, -0.25, 0.25, 0, 0];
```

Verify the angular position of the drone does not reach the attitude limits!

11. Apply the designed reference signal and observe the system response.

12. Run the disconnect section

```
%% Close connection
drone.disconnect();
```

13. Finally, extract the experiment data for post-processing. Save the *.mat files to a USB stick for analysis on your own PC.

Session 4 – Position control

POLITECNICO
MILANO 1863
DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

Post-processing:

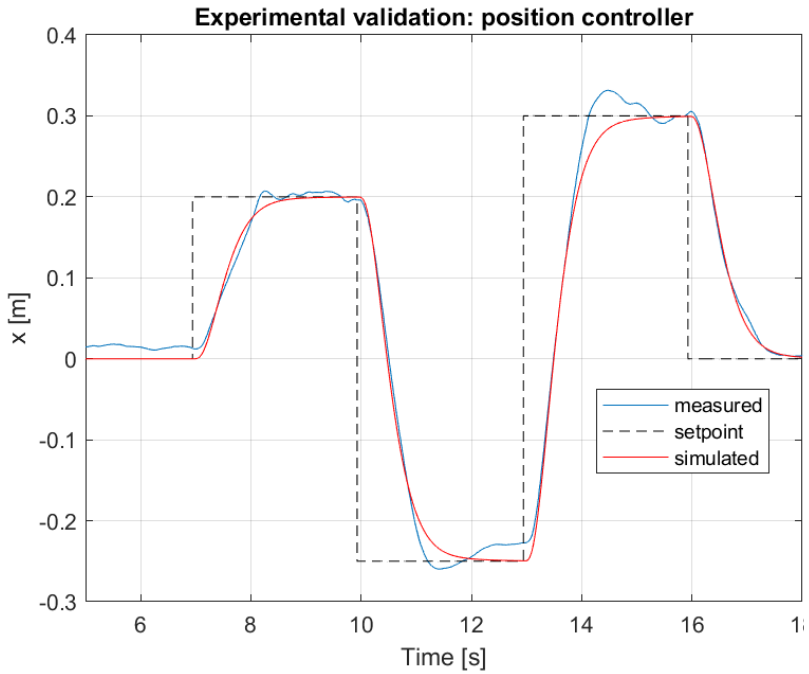
- In your PC, save and load the *.mat file in Matlab.
- For this experiment we are interested in the following variables:
  - theta: pitch angle (rad)
  - v: velocity (m/s)
  - x: position (m)
  - x0: position set point (m)
- Plot the step response of the control system and characterize the performance of the loop in terms of standard indexes
  - Rise time
  - Settling time
  - Overshoot
  - Oscillation period
  - Input peak amplitude and saturation events
- Compare the performance of your controller with the response obtained in simulation.

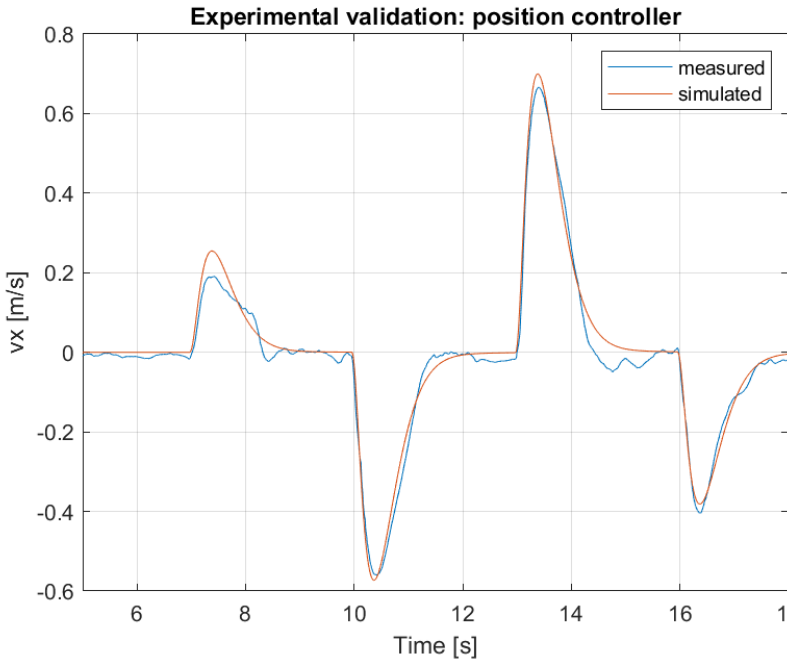
Session 4 – Position control

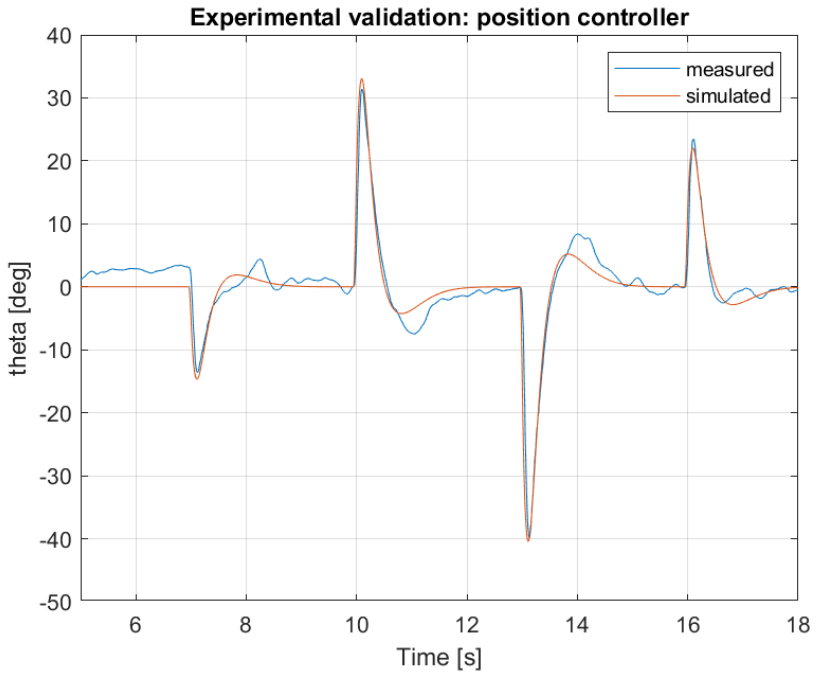
POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Hands-on task:

Expected response of a control system with $\omega_c^{pos} = 1.49\ rad/s$





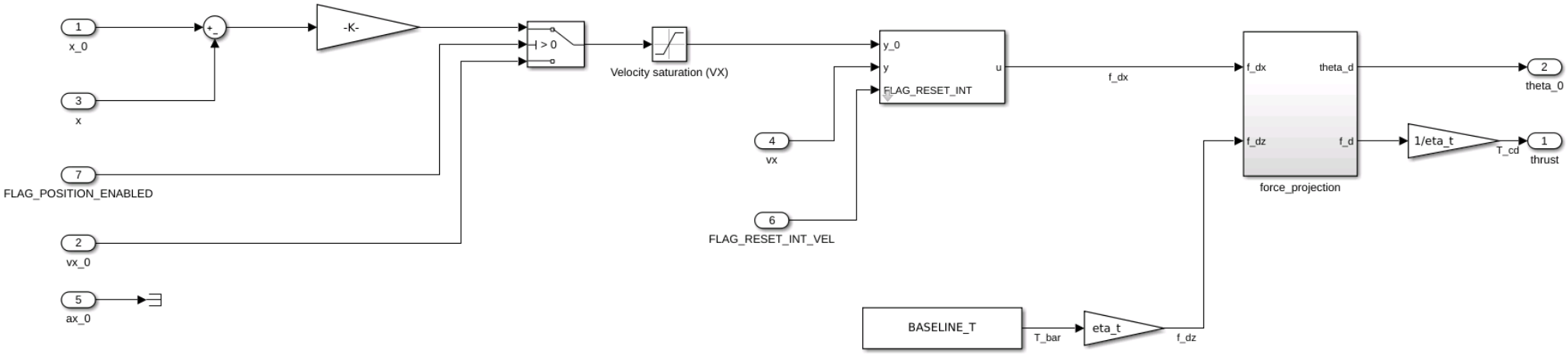


This is the end of the activity.

Session 4 – Position control

POLITECNICO MILANO 1863
DIPARTIMENTO DI ELETTRONICA INFORMAZIONE E BIOINGEGNERIA

Rapid Prototyping Systems:



- Automatic generation of control routines from Simulink schemes:

1. Open Matlab
2. Go to the folder …\Antx\DroneCmd\TSCA\2DOF_linear_tunable
3. Open the project 2DOF_linear_tunable.prj
4. Explore the controllers blocks and verify the tunable parameters

Session 4 – Position control

Rapid Prototyping Systems:

- Automatic generation of control routines from Simulink schemes:

5. In Matlab run the script *SLXtoPX4_gui_main*

6. In the GUI, add the path to the Simulink controller model

   f*light_controller.slx*

7. When the compilation process indicates it, Connect the drone to the PC using the USB and wait until the code is downloaded to the drone CPU.

8. When done, the controller contained in the Simulink project is ready to be executed in the drone embedded system.