Computer session on Modeling and Control Design of an UAV

ASI 3A - MARS

2021-2022

Abstract

In this computer lab session, we will work on modeling and control design of UAVs (Unmanned Aerial Vehicles). It is an open project with some predefined objectives. You are free to use any control technique for your aerial drone.

1 Problem statement and objectives of the lab session

We will work with one of the Gipsa-lab quadrotor-drones (see figure 1). This aerial drone has 4 fixed rotors disposed in a cross configuration, also notated by X4.

Figure 1: Kopis quadrotor from Gipsa-lab.

The final objective of this lab is to design a controller to make the quadrotor follow a reference trajectory (that will be provided, no need to code it) as the helix one shown in Figure 2.

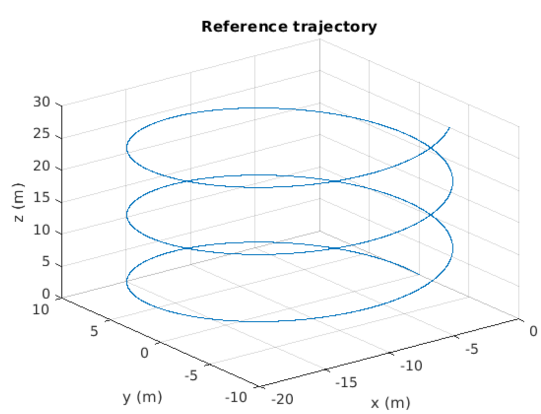

Figure 2: Helix trajectory.

As presented during the Mobile Robotics lecture on aerial vehicles by Nicolas Marchand, the X4 quadrotor is an under-actuated system with only 4 controllable inputs. Since the translation and orientation dynamics are coupled, the objective of this computer session is to apply a hierarchical control technique where the position and attitude control are designed separately.

We define two coordinates reference frames:

- Fixed inertial frame (global) $\mathcal{E} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}$
- Body frame linked to the body of the UAV (local) $\mathcal{B} = \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix}$

The dynamics of the quadrotor depend mainly on the four actuators, the physical and mechanical characteristics of the drone and the aerodynamics. Let $\vec{\xi}$ denote the position of the drone in the global frame, $\vec{v}$ its speed vector, ϕ, θ, ψ the Euler angles and $\vec{\Omega} = [p \quad q \quad r]^T$ the rotational speed expressed in the body frame of the drone. For this lab session, we assume that all the variables are measurable.

If we assume that the actuators (four electrical motors) are controlled by a local inner loop, their dynamics can be neglected with respect to the dynamics of the UAV body. In this situation, as seen during the course, the nonlinear *complete* dynamic model of the UAV body is expressed as:

$$\begin{cases} \dot{\vec{\xi}} &= \vec{v} \\ m\dot{\vec{v}} &= -mg\vec{e_3} + RT\vec{b_3} - K_v||v||\ \vec{v} \\ \dot{R} &= R\vec{\Omega^{\times}} \\ J\dot{\vec{\Omega}} &= -\Omega^{\times} J\vec{\Omega} + \vec{\Gamma_{CR}} + \vec{\Gamma_{Gyro}} + \vec{\Gamma_C} \end{cases} \tag{1}$$

where the torques $\vec{\Gamma_C} = [\Gamma_p \quad \Gamma_q \quad \Gamma_r]^T$ and the thrust T are the control variables.

The rotation matrix about axes X-Y-Z for the Euler angles ϕ, θ, ψ is denoted as:

$$R(\phi, \theta, \psi) = R(\psi)R(\theta)R(\phi)$$

$$R(\phi, \theta, \psi) = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}$$

As explained in the course, considering several assumptions to simplify this model, the dynamics of the quadrotor can be rewritten as in the following simplified model:

$$\begin{cases} \dot{\vec{\xi}} &= \vec{v} \\ m\dot{\vec{v}} &= -mg\vec{e_3} + RT\vec{b_3} \\ \dot{R} &= R\vec{\Omega^{\times}} \\ J\dot{\vec{\Omega}} &= -\Omega^{\times} J\vec{\Omega} + \vec{\Gamma_C} \end{cases} \tag{2}$$

with the control input $u = [\Gamma_p \quad \Gamma_q \quad \Gamma_r \quad T]^T$

A schema of the dynamics of the quadrotor is illustrated in Fig 3.

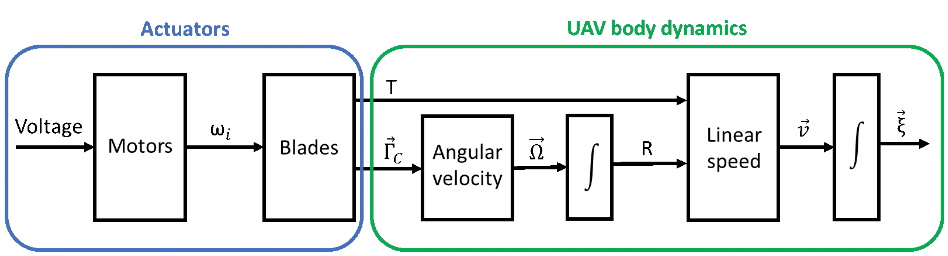

Figure 3: Schematic representation of the drone's dynamics.

2 Modeling (2h)

First of all, we study the model of the quadrotor. The objective of this part is to understand all the variables involved in the dynamics of the drone and the relations between them in order to be able to obtain a simplified linear model of the drone. This linear model will be the first block to be implemented in your Matlab simulator. The structure of this block should be as in Fig 3.

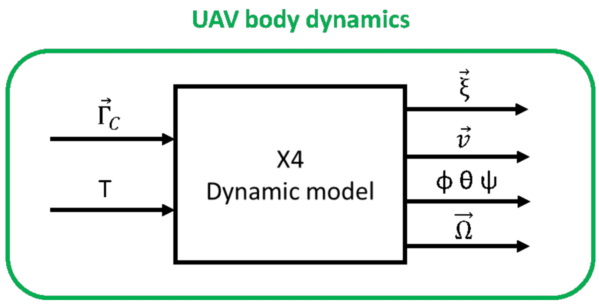

Figure 4: Schematic representation of the drone's dynamics.

1. Justify, as detailed as possible, the assumptions made to write the simplified model of the quadrotor.

2. From the compact equations (2) of the simplified model, write the extended version of these equations. Hint: write an equation for each scalar variable instead of vector equations. Use the Wronskian matrix introduced in the course defined as:
$$W = \begin{pmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{pmatrix}$$

3. In Matlab, code a file to define all the parameters of your quadrotor which will be used in the simulations (the numerical values of the parameters are provided in the appendix).

4. Considering the simplified model of the X4 drone, we define the space state variables as follows $X = \begin{bmatrix} \vec{\xi}^T & \vec{v}^T & \phi & \theta & \psi & p & q & r \end{bmatrix}^T$ and the control input as $U = \begin{bmatrix} \Gamma_p & \Gamma_q & \Gamma_r & T \end{bmatrix}^T$. Find a linear model of the form $\dot{X} = AX + BU$ for the quadrotor applying a linearization around the equilibrium point corresponding to the stationary flight state $X_{eq} = \begin{pmatrix} x_e & y_e & z_e & 0_9 \end{pmatrix}^T$ and $U_{eq} = \begin{pmatrix} mg & 0_3 \end{pmatrix}^T$.

5. Code your linear model in Simulink.

3 Control design: Hierarchical control loops approach (2h)

In the hierarchical control technique, the main idea is to design several control loops at different levels: an internal loop for the attitude control and an external loop for the position control, (see Figure 5).

In order to design appropriately each controller, we will work on one control loop at a time. We will start with the inner control and finish with the external one. Each controller must be designed taking into account that inner control loops should be quicker that external control loops.

Hint: To design a controller for each loop separately, try to linearize the dynamics of the variables involved in that loop.

1. Rotational speed control loop: design a controller (PID, LQR, ...) to stabilize the rotational speed $\vec{\Omega} = [p \quad q \quad r]^T$.

2. Attitude control loop: design a controller (PID, LQR, ...) to stabilize the Euler angles ϕ, θ, ψ.

3. Position and speed control loop: design a controller (PID, LQR, ...) to stabilize the position $\vec{\xi}$ and speed $\vec{v}$ of the quadrotor.

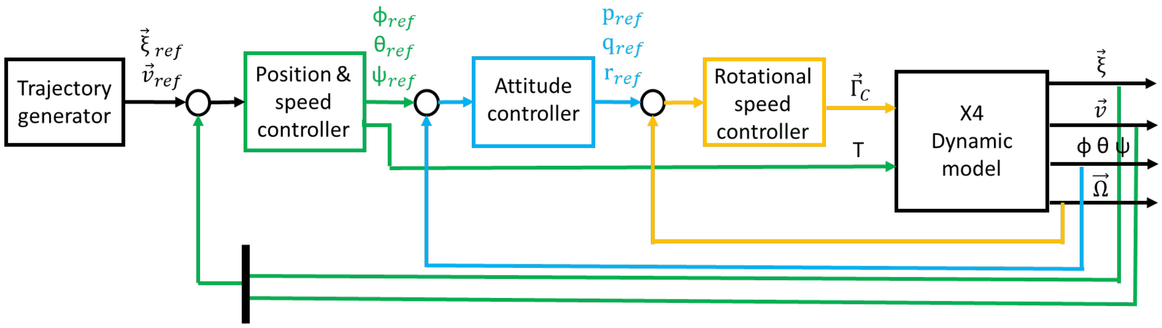

Figure 5: Three different control loops strategy.

4 Appendix

The different parameters of the drone are given in the following code

```
cT = 1.3529e-06 ;
cQ = 2.1113*1e-08 ;
l = 110*1e-3 ;
lp = sqrt(2)/2*l ;
m = 0.536 ;
g = 9.81 ;
rho_ground = 1.4 ;
rotor_radius = 6.0*1e-2 ;
hauteur_rotor = 29.3*1e-3 ;
Ix = 0.029125 ;
Iy = 0.029125 ;
Iz = 0.055225 ;
```

For the motors used we have the following parameters:

```
Kv_motor = 1750 ;
Kv_motor_rad = Kv_motor*2*pi/60 ;
km = 1/Kv_motor_rad ;
Jr = 1.2991*1e-05 ;
R = 0.07 ;
U_sat = 12 ;

Mx = [ cT cT cT cT ;
 -cT*lp cT*lp cT*lp -cT*lp ;
 -cT*lp -cT*lp cT*lp cT*lp ;
 cQ -cQ cQ -cQ ] ;
```