Project 3

Motion control and planning

B. Guerreiro

Unmanned Aerial Vehicles – 2023/2024

Abstract

This project aims at designing basic control loops and planning algorithms for the Crazyflie drone. In a first part, linear control loops will be explored, while in a second part, a nonlinear control loop for the full 3-D motion of the vehicle will be considered. The last part we will aim at implementing a simple planning strategy for avoiding a previously unknown no-fly-zone. In this lab project, the knowledge acquired in the first two projects will be important, both for having a good model of the drone, as well as to understand how the drone is localizing itself in the arena.

Introduction

Drones, or unmanned aerial vehicles (UAVs), are nowadays a technology under intense development, where the safety and security concerns require that the systems deployed into the society have proven performance and stability. Although operating a single UAV with a Human-in-the-loop is common, using fully autonomous drones to operate in complex and GPS-denied environments is still a task under intense research in academia and in the relevant industries. In this third and last project you will consider the design of basic linear and nonlinear control loops,as weel as simple planning algorithms to drive a Crazyflie 2.1 [2], shown in Figure 1, along a desired path that may contain obstacles.

Learning outcomes

This project assignment will help you to achieve the following learning outcomes:

- Test basic linear controllers in simulation and experimentally.
- Design and test nonlinear controllers in simulation and experimentally.
- Design and test planning algorithms in simulation and experimentally.

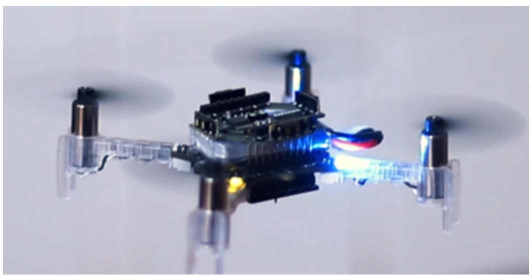


Figure 1: Crazyflie 2.1 drone.

Organization

The project assignment will take place in 4 sessions divided in 3 parts: Section 1 regarding linear control design, Section 2 dealing with nonlinear control design, and Section 3 focused on the design of a planning strategy. The evaluation of this work has three components totaling 15% of the final grade):

1. Report and in class assessment (12%): Moodle submission by 15/05/2024;
2. Peer evaluation (2%);
3. Self evaluation (1%).

Each group is expected to prepare each session before the class, by making the necessary theoretical computations and preparing the necessary Matlab/Octave code. Students are advised to write the paper as the work progresses[1], and will be asked to show the a preliminary version of the report in each session. The reasoning behind all answers **must be explained** and, if necessary, complemented with other auxiliary calculations. Nonetheless, the report should be as concise as possible. You can write in English or in Portuguese and use any editor of your choice (MS word, Latex, handwritten, etc), but the submitted document should be a **PDF file**.

The report should be submitted through Moodle, whereas the Matlab code used to obtain the

[1] A paper template in LaTeX is provided at `https://www.overleaf.com/read/zfzrrqgxqkgm`.

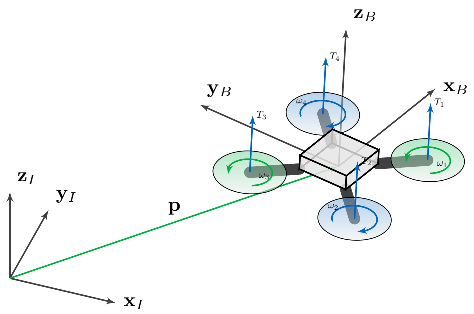


Figure 2: Crazyflie reference frames and configuration.

results should be shared in a GitHub or BitBucket repository, explicitly mentioned in the report. If this is a new tool for you, please go through some basic guide such as this or this. For beginners and Windows users, using a GUI might seem more comfortable, so consider using GitHub Destop or Sourcetree.

Ethics

In light of true academic freedom, students can and should discuss their ideas and approaches freely with other students, teachers, or the community at large. Nonetheless, the reports submitted by the students must be **original** and correspond to their **actual work**, following a widely accepted academic ethics code, also detailed in the *Código de Ética da Universidade Nova de Lisboa* (`https://www.fct.unl.pt/sites/default/files/UNL_Codigo_etica.pdf`).

Furthermore, a CRediT author statement (see details here) should be included in the report, clearly and explicitly saying what was the contribution of each student.

1 Linear Control

In the first part of this project you will design a simple linear controller for the position control of the Crazyflie 2.1 drone in 3-D, considering a zero yaw. Consider the same assumptions as in Project 1, where we assume the drone is a rigid-body in 3-D space with the relevant external forces acting on the center of mass, as illustrated in Figure 2, and the body frame is located at the center of mass, considering a mass m, and a moment of inertia matrix $\mathbf{J}$ as identified in Lab. 1.

Towards this end, take the following steps to obtain a working simulation before the experimental trials.

1.1 Based on the results of Project 1, obtain a simplified nonlinear model of the Crazyflie drone that receives as input the vector $\mathbf{u}_\lambda \in \mathbb{R}^4$, defined as $\mathbf{u}_\lambda = \begin{bmatrix} T & \boldsymbol{\lambda}^T \end{bmatrix}^T$, with the total thrust, T, and the Euler angle vector $\boldsymbol{\lambda} \in \mathbb{R}^3$, composed of the roll angle, ϕ, pitch angle θ, and yaw angle, ψ.

1.2 For small roll and pitch angles, and assuming zero yaw, obtain an equivalent actuation in acceleration, $\mathbf{u}_a \in \mathbb{R}^3$, combining the effects of the total thrust, gravity, and drone attitude.

1.3 Define the linear model for the linear motion of the drone, considering the control variable $\mathbf{u}_a$ and the state vector $\mathbf{x} = \begin{bmatrix} \mathbf{p}^T & \mathbf{v}^T \end{bmatrix}^T$.

1.4 Design an LQR controller for this linear model and test it in simulation. Note that you should also test the controller considering the nonlinear model.

desejado (ref)

1.5 Consider now an error state vector defined as $\tilde{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$, where we assume that the reference state, $\bar{\mathbf{x}}$, is driven by the same dynamics as $\mathbf{x}$. Obtain the equivalent state space model.

1.6 Design an LQR controller for this error model and test it in simulation. Comment on the difference between this linear model and the previous one.

2 Nonlinear Control and Trials

Building on the previous linear control algorithms and using concepts of Lyapunov theory:

2.1 Design a nonlinear controller with actuation in body accelerations, $\mathbf{u}_a \in \mathbb{R}^3$ and assuming a zero yaw, that is able to achieve asymptotic stability in the Lyapunov sense (prove this result).

2.2 Test this controller in simulation.

Considering the controllers developed in the previous questions, we will test them in the real drones. The implementation will be in the form of a Python function that has access to the current state and desired state, and defines the desired acceleration vector in 3-D for the drone inner loops to follow.

2.3 Follow the teacher instructions to use the arena PC and development environment to test your controllers in the arena, ensuring adequate saturations for the inputs.

2.4 Obtain the data logs for the experiment and compute the RMSE values of the 3-D error between the real and desired position, according to the expression:

$$p_{RMSE} = \sqrt{(1/N \sum_{k=0}^{N} \|p(k) - p_d(k)\|^2)}$$

as well as the velocity RMSE given by

$$v_{RMSE} = \sqrt{(1/N \sum_{k=0}^{N} \|v(k) - v_d(k)\|^2)}$$

The performance of the controller in terms of these values (lower values of p_{RMSE} and v_{RMSE} are better) will be taken into account.

3 Motion Planning

In this section we will build on the previous controllers and design the reference trajectories for the control loops. For this purpose, consider the flying arena usable limits defined by

$$\begin{aligned} \mathcal{P}_{Fly} = \{\mathbf{p} \in \mathbb{R}^3 | -1.2 \leq & p_x \leq 1.2, \\ -2.1 \leq & p_y \leq 2.1, \\ 0.0 \leq & p_z \leq 2.0\} \end{aligned}$$

and the no fly zones or obstacles defined by

$$\begin{aligned} \mathcal{P}_{NoFly1} = \{\mathbf{p} \in \mathbb{R}^3 | -0.6 \leq & p_x \leq 0.6, \\ 0.7 \leq & p_y \leq 0.8, \\ 0.0 \leq & p_z \leq 1.2\} \end{aligned}$$

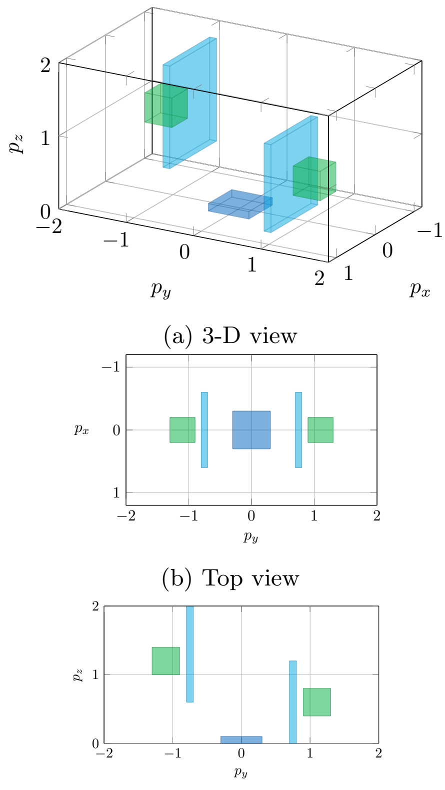
$$\begin{aligned} \mathcal{P}_{NoFly2} = \{\mathbf{p} \in \mathbb{R}^3 | -0.6 \leq & p_x \leq 0.6, \\ -0.8 \leq & p_y \leq -0.7, \\ 0.6 \leq & p_z \leq 2.0\} \,. \end{aligned}$$

Consider also the set of possible initial positions defined by

$$\begin{aligned} \mathcal{P}_{Start} = \{\mathbf{p} \in \mathbb{R}^3 | -0.3 \leq & p_x \leq 0.3, \\ -0.3 \leq & p_y \leq 0.3, \\ & p_z = 0\} \end{aligned}$$

and two sets of possible goal positions

$$\begin{aligned} \mathcal{P}_{Goal1} = \{\mathbf{p} \in \mathbb{R}^3 | -0.2 \leq & p_x \leq 0.2, \\ 0.9 \leq & p_y \leq 1.3, \\ 0.4 \leq & p_z \leq 0.8\} \end{aligned}$$



(a) 3-D view

(b) Top view

(c) Back view

Figure 3: Flight arena planning setup.

$$\begin{aligned} \mathcal{P}_{Goal2} = \{\mathbf{p} \in \mathbb{R}^3 | -0.2 \leq & p_x \leq 0.2, \\ -1.3 \leq & p_y \leq -0.9, \\ 1.0 \leq & p_z \leq 1.4\} \,. \end{aligned}$$

as depicted in Figure 3.

3.1 Implement a planning algorithm of your choice that is able to drive the drone from any initial $\mathbf{p}_{ini} \in \mathcal{P}_{Start}$ to a first goal position $\mathbf{p}_1 \in \mathcal{P}_{Goal1}$, to a second goal position $\mathbf{p}_2 \in \mathcal{P}_{Goal2}$, and back to the starting point, ensuring the drone is always within $\mathcal{P}_{Fly}$ and avoids $\mathcal{P}_{NoFly1}$ and $\mathcal{P}_{NoFly2}$.

3.2 Test this planning algorithm with your controller in simulation, noting that that $\mathbf{p}_{ini}$, $\mathbf{p}_1$, and $\mathbf{p}_2$ are randomly chosen from their respective sets.

Description of the experimental tasks.

3.3 Test your planning algorithm and controller in the arena.

Conclusions

This project addressed several methods for controller design and planning algorithms to address UAV autonomy tasks, that we hope have lead you to a better understanding about some basic techniques used in UAVs. Optionally, you can also include in your report an overview of what you learned in this work and how it may be useful for you in the future.

References

[1] Brian D.O. Anderson and John B Moore. *Optimal filtering*. Dover, 2012.

[2] Bitcraze. Crazyflie documentation. `https://www.bitcraze.io/documentation/start`. Accessed: 2024-03-14.

[3] Bruno Guerreiro. *Lecture Slides for Unmanned Aerial Vehicles*. UNL, 2024.

[4] Andrew H Jazwinski. *Stochastic processes and filtering theory*. Dover, 2007.

[5] Sigurd Skogestad and Ian Postlethwaite. *Multivariable feedback control: analysis and design*. John Wiley & Sons, 2005.

[6] Ashish Tewari. *Advanced control of aircraft, spacecraft and rockets*. John Wiley & Sons, 2011.