Course notes, module 6

UAV attitude failure detection

Kjeld Jensen kjen@mmmi.sdu.dk

1 Agenda

- Short Introduction to the module theory and exercises.
- Exercises.

2 Theory presented in class

The topic of this module is the basic principles of UAV attitude failure detection. The F3322-22 standard specification is very relevant in relation to this, as it is the current industrial standard referred to by authorities concerning the requirements of a Parachute Recovery System (PRS). Data logged by the Pixhawk Flight Controller (FC) running the PX4 firmware and the analysis of this data is also part of this module.

The learning goals of this module are:

1. Get familiar with the concepts of UAV attitude failure detection principles.
2. Get familiar with the F3322-22 standard specification published by American Society for Testing and Materials (ASTM).
3. Gain knowledge about logging on the Pixhawk Flight Controller

3 Exercises

3.1 F3322-22 Standard

This exercise is about the F3322-22 standard. It is documented in the presentation available in the course materials, and here you will find the questions to be answered in this weeks' report.

3.2 PX4 log file analysis using an online tool

This exercise is about analysis of a Pixhawk PX4 log using an online tool This is also documented in the presentation available in the course materials.

3.3 UAV attitude failure detection algorithm

In this exercise you will develop an algorithm for detection of an UAV that is suddenly spinning to the ground as described in the following.

You will write the algorithm in Python by modifying an existing script in the course materials. You will then test your algorithm in a simple python simulator included in the existing script that replays data from a PX4 log recorded under a drone flight.

You will have access to 3 different Pixhawk PX4 flight log files from drone flights. For the flights a hexacopter was used. You can see it in figure 1.

During all drone flights the drone was taken to a certain height, then the motors were turned off and a parachute was ejected. More specifically the drone flights were conducted accordig to the following plan:

1. The UAV is armed.
2. The UAV takes off and fly to an altitude of 50-70 m.
3. The UAV is disarmed in air.
4. Pause for a short period while the UAV is falling.
5. Parachute is deployed.
6. The UAV descends in the parachute
7. The UAV hits the ground

This means that the drone is first taking off and flying a normal flight, then it spins down to the ground for a short while until the parachute is ejected after a fixed period of time. Your algorithm needs to detect this change from normal flight to spinning to the ground and create a "failsafe trigger" that could be used to automatically eject the parachute at the correct time.

How you do this is up to you. In the python script you can see the data from the gyro and accelerometer sensors. You also have pressure data from the barometer. For your convenience the kill switch is included, this is just to aid you in analyzing the data because you then know when the motors are turned off.

The script also plots the barometric pressure and kill switch, this is because log files often has a lot of time before and after the flight, and the time of the actual flight is easily identified by the pressure graph. You can use the two variables start_seconds and end_seconds to limit the data, you look at, to the actual flight. Feel free to extend the script to plot more data if you wish.

Below are links to youtube videos for each of the 3 flight logs. This may help you in analyzing the log data:

Test 5: `https://youtu.be/raK2fnk5ULk`
Test 8: `https://youtu.be/gJK06kHUvz8`
Test 9: `https://youtu.be/opDSC2ox-c4`

Basic physical parameters of the UAV (and parachute) can be seen in table 1.

Parameters	Symbol	Value
Gravity	g	$9.8m/s^2$
Multirotor mass	m	$1.95kg$
Air density	ρ	$1.225kg/m^3$
Effective cross-sectional area of parachute	$A_{parachute}$	$1m^2$
Coefficient of drag for parachute	$C_{parachute}$	1.6

Table 1: Multirotor and parachute parameters.

Figure 1: S550 Hexacopter with Pixhawk 2.1 Cube Flight Controller used in parachute system tests.

In the Python script in the course materials you will find practical information about installation of Python packages and running the script. In Ubuntu 24.04 Python recommends using a virtual environment when installing Python modules. This is different from what most of us are used to, but it is rather simple and you will find all commands to do this. We have tested that it works on Ubuntu 22.04 and 24.04.

3.3.1 Implement algorithm

Implement an algorithm that is able to detect the UAV attitude failure using data from accelerometers, gyros and the barometer. If you can think of two different methods, implement both and then compare them in the following exercises. Describe your algorithm(s) in the report.

3.3.2 Test algorithm

Test your algorithm(s) on the data sets from tests 5, 8, and 9. Comment on the results.

3.3.3 Visualization

Visualize your failure detection algorithm(s) together with relevant IMU data. Comment on the results.

3.3.4 Parachutes

Perform a quick state-of-the-art survey of parachute systems for drones. List the different major manufacturers and briefly describe their product in a single paragraph.