UNIVERSITÀ DEGLI STUDI FIRENZE
Da un secolo, oltre.

School of Engineering

Computer Science Department (DINFO)

Bachelor in Electronics and Telecommunication Engineering

Analysis, Simulation and Control of a quadrotor UAV prototype

Elisa Ferrara

Context and objective of the Thesis

The Flight Controller represents in someway the brain of a drone. It allows it to create the link between the decisional aspects and the actuators. My thesis project had the goal to synthetize the controller for the DART quadrotor three-blade prototype previously built in the SysCon Lab of the University of Florence (Figure 1).

Figure 1. DART prototype

The controller has the final goal of chasing the trajectories of a planner in a realistic environment: obstacles and disturbances of the environment need to be overcome. In order to achieve this goal, the main specifications are:

1. Inner control (broadband) over yaw rate, speed of ascent and attitude
2. Outer control over yaw angle, altitude and x-y position

Materials and methods

After an accurate study of aerial vehicles, the quad-copter dynamics [5], the most common control techniques used and the methods of estimation of the parameters (especially the aerodynamic ones), the work consisted in:

1. Estimating the drone's parameters thanks to real measurements about dimensions, weights, and load tests.
2. Implementing the controller on **Matlab-Simulink**.

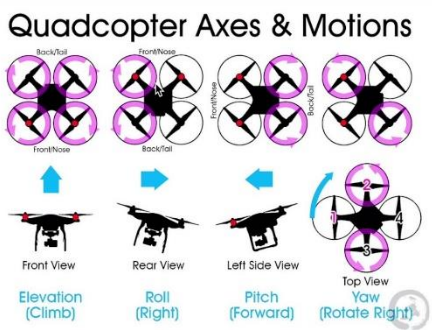


Figure 2 Movements and Rotations of the quadcopter

Plant Model

The quad-rotor follows the same dynamics of the rigid object: Newton Law on the translational side and Euler's one on the rotational one. The forces considered are the gravitational one and the total thrust of the rotors; the air resistance is neglected. The torques is the three directions are generated by modulating the four angular velocities of the rotors as shown in Figure 2. The quadrotor is under-actuated: it is needed to combine the z-thrust and the 3 torques to achieve all the possible configurations.

Parameters estimation

A model for the masses (Figure 3) of the drone was created as well as its CAD. The position of the different parts was studied for achieving a diagonal inertial matrix. The esteem of the thrust (Ct) and torque (Cq) [3] parameters used the load test of the tri-blade rotors.

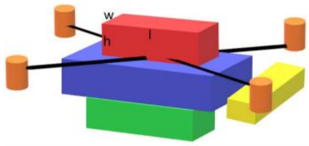


Figure 3 Model for the mass distribution of the quadrotor

MATLAB Simulink Implementation:

PLANT: In order to implement the model of the quadrotor, we used the so called *Multirotor* in cascade with the *6DOF Euler Angles Platform* Simulink blocks. The first one computes the aerodynamic forces and moments generated by multiple rotating propellers or rotors, such as quadcopters, in all three dimensions and the other one implements six-degrees-of-freedom equations of motion.

FLIGHT CONTROLLER: The synthesis followed the hierarchy of the control itself [1]: from the inner broadband controller to the outer ones (Figure 4). While altitude (or speed of ascent) and yaw (or its rate) are controlled in a single negative feedback loop, position regulation follows a cascade control: an inner loop for attitude which input is the output of the outer controller of position. Finally, z-thrust and the 3 directions torque are mixed and transformed in 4 angular velocities for the rotors by the motor mixing algorithm.

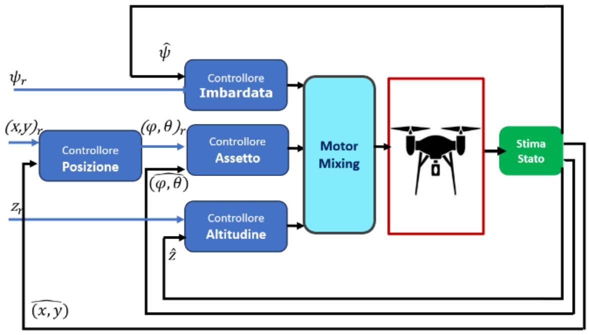


Figure 4 General scheme for the flight controller

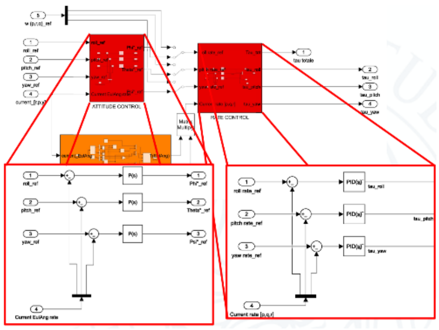


Figure 5 Attitude Controller on Simulink

The type of controller preferred for the thesis is the PID [2]. The general scheme for each controller is a position-velocity one (Figure 5 shows the example of the attitude controller): an inner loop with a PID controller for the first derivative of the subject variable and an outer proportional loop for the variable itself. This structure allows to improve the performances and to achieve a double choice control: on the variable and on its rate.

Moreover, the motor-mixing [4] block's main idea is to invert the operations of the Multirotor block. In order to do so it performs this matrixial operation:

$$\begin{bmatrix} {\omega_1}^2 \\ {\omega_2}^2 \\ {\omega_3}^2 \\ {\omega_4}^2 \end{bmatrix} = M^{-1} * \begin{bmatrix} T_Z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \quad \text{where} \quad \text{M=}\begin{bmatrix} -ct & -ct & -ct & -ct \\ dx*ct & -dx*ct & -dx*ct & dx*ct \\ dy*ct & dy*ct & -dy*ct & -dy*ct \\ cq & -cq & cq & -cq \end{bmatrix}$$

Figure 6 shows the final project on Simulink.

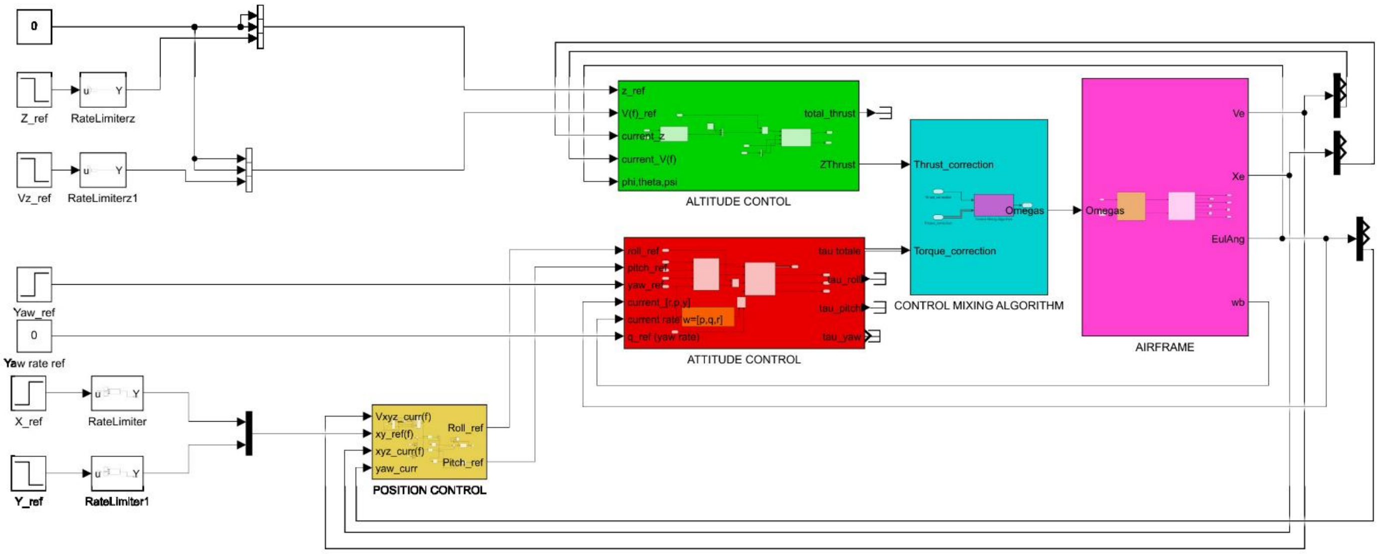
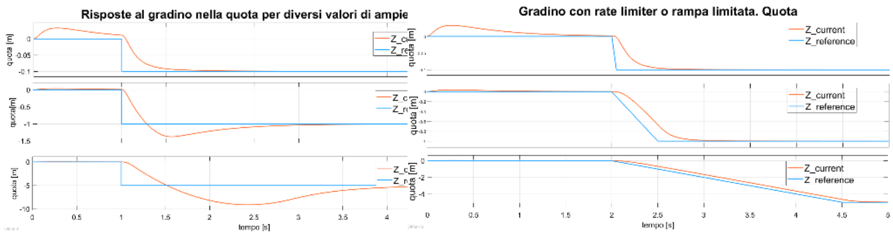
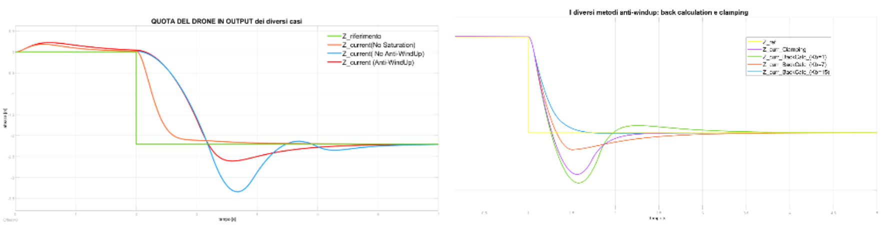


Figure 6 Controller and plant Simulink implementation

Results and discussion

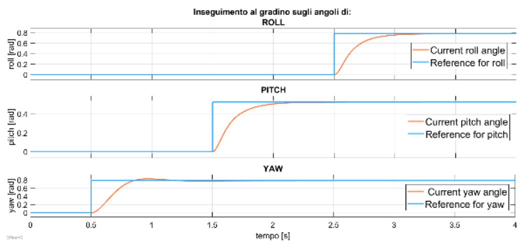
Altitude controller: Step response rise time: 0.3s. The figures on the right show the effect of the <u>actuators' saturation</u> (left one) and its resolution thanks to the Rate Limiter block (on the right).

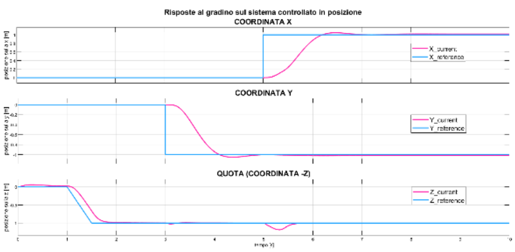




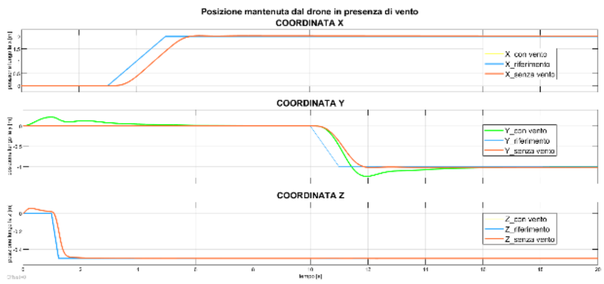
Moreover, the figures on the left show the study on the <u>wind-up problem</u> [2] of the PID controller in the case of altitude regulation. On the left is the step response in case different cases (without saturation, with and without wind-up). On the right the best ant- windup method is analysed resulting in back calculation.

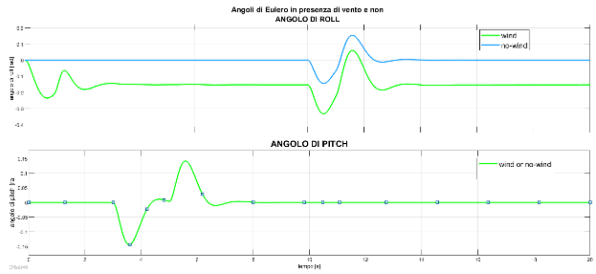
Attitude Controller: Step response rise times: 0.4s for roll and pitch, 0.25s for yaw (Figure on the right).

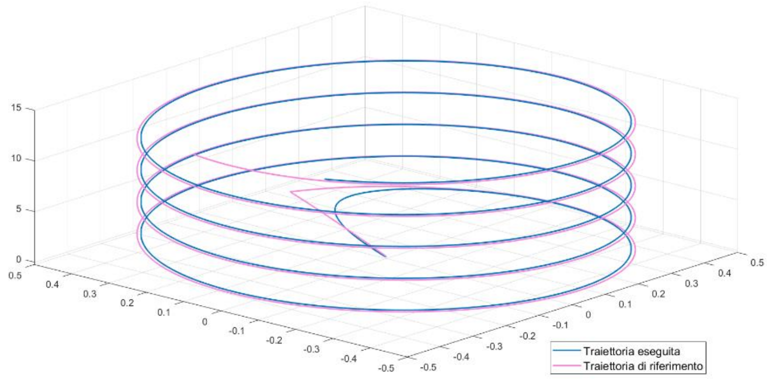




Position Controller: Step response rise times on x and y: 1.2s (figure on the left). Below there are the figures representing a plant disturbance (wind in y-direction of 10m/s). The time to come back to the equilibrium position is 3s (left figure). The figure on the right shows the attitude of the drone to overwhelm the disturbance (roll angle of 8-9 degrees).







Trajectory tracking: In the graph on the left it was simulated an upward spiral path thanks to a sinusoidal wave in phase quadrature on the x-y reference and a rate of ascent control for the z coordinate.

Conclusion and future perspectives

Step response rise times reached are adequate to the use of the controller (0.3s inner loops and 1.2s outer loop) if compared to the ones in literature. Thanks to the use of the double control position-velocities the performances are better than the flight controllers that use just a single PID feedback regulation.

The main problem concerned the saturation of the actuators and the resulting malfunctioning of the Motor Mixing block. Firstly, anti-windup techniques were implemented widening the linear functioning area. Then the use of rate limiters, which indirectly gave a limit to the maximum thrust, solved the problem at the cost of a lower movement speed.

With a view to running the real prototype the Flight Controller still needs to simulate the sensors. Simulink offers many useful blocks such as the IMU one.

Finally, the controller could be useful, in an autonomous flight perspective, to track the trajectories of a planner that works in a noisy and obstacles-full environment.

References

[1] Mahony R., Kumar V., Corke P. (2012). *Multirotor Aerial Vehicle*. 1070-9932/12/\$31.00ª2012IEEE
[2] Bolrezn, P., Scattolini, R., & Schiavoni, N. (2015). *Fondamenti di controlli automatici*. McGraw-Hill Education
[3] Kim, K. C. (1999, June). *Analytical Calculations of Helicopter Torque Coefficient (CQ) and Thrust Coefficient (Or) Values for the Helicopter Performance (HELPE) Model*. Army Research Laboratory.
[4] Tiwari, A. (2017). Position Control of an Unmanned Aerial Vehicle From a Mobile Ground Vehicle. Michigan Technological University.
[5] Vendittelli, M. (2022). *Introduction to Aerial Robotics* [pdf]. Università La Sapienza di Roma.