



SAPIENZA
UNIVERSITÀ DI ROMA

FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE, INFORMATICA E STATISTICA

MODELING AND CONTROL OF MULTI-ROTOR UAVS

Flight control of a Quadrotor Vehicle Subsequent to a Rotor Failure

Professor

Marilena Vendittelli

Students

Edoardo Ghini

Gianluca Cerilli

Giuseppe L'Erario

Academic Year 2017/2018

Contents

1	Quadrotor model	3
2	Controller architecture	6
2.1	Quadrotor model in case of rotor loss	7
2.1.1	Control inputs constraints	7
2.1.2	State-space model	8
2.2	Outer control loop	9
2.3	Inner control loop	12
2.3.1	Feedback Linearization	13
3	Simulations	15
4	Conclusions	23

Introduction

Quadrotors are small aerial vehicles propelled by four rotors. These are widely used thanks to the recent advances in sensors, small size and low cost.

However, the small structure makes them brittle and vulnerable.

In this regard, our project deals with the development of a control scheme that can be applied in case of loss of one of the rotors, so as to allow the vehicle to land only with three actuators, starting from the study of the work of *Lanzon et al.*[1].

In case of loss of one actuator, full control of attitude states is lost. A solution to this loss is to renounce the yaw angle control, maintaining flight control of a spinning vehicle.

One of the most common control strategy is based upon a double-loop architecture:

- an outer controller whose aim is to modify the attitude angles by small values in order to perform the trajectory;
- an inner controller whose task is to regulate these attitude angles acting on the generalized forces.

In [1] it's proposed a strategy slightly different from the classical one:

- in the outer control, the yaw velocity is computed instead of the yaw angle;
- the inner control regulates the roll angle, the pitch angle and the yaw velocity via a feedback linearization strategy[2][3].

The theoretical arguments were implemented and validated using MATLAB and SIMULINK.

The report is organized as follows. In the first part will be presented the quadrotor model. Then are described the control architecture and the relative results obtained via simulations. The conclusions are presented at the end of the report.

1 Quadrotor model

A quadrotor consists of four rotors on which propellers are fixed. The rotors are positioned at the ends of a X-shaped frame. The front and rear motors \mathcal{M}_1 and \mathcal{M}_3 spin in the clockwise direction, the other two motors \mathcal{M}_2 and \mathcal{M}_4 spin in the counter-clockwise direction, as shown in fig.1.

The mathematical is based on three assumptions:

- the quadrotor structure is symmetrical. This means that the inertias along the longitudinal axis and the lateral one are equal;

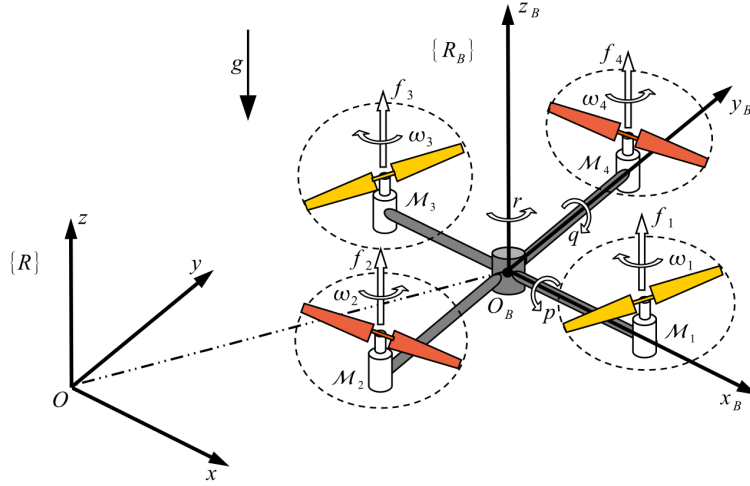


Figure 1: Quadrotor scheme used for the development of the mathematical model.

- propellers are considered rigid: no blade flapping;
- drag is assumed linear.

As one can see in fig.1, we define two reference frames:

- the first one, representing the Earth frame: $\{R\}\{O, x, y, z\}$;
- the second, representing the body-fixed frame: $\{R_b\}\{O_b, x_b, y_b, z_b\}$.

The body fixed frame is connected to $\{R\}$ by a position vector $[x, y, z]$, which describes the center of gravity in $\{R_b\}$ relative to $\{R\}$, and by a vector of angles $[\phi, \theta, \psi]$, i.e. the roll, pitch and yaw angles, which represent the orientation of the body frame w.r.t the Earth frame.

The rotation along roll and pitch angles are constrained¹:

$$\left(-\frac{\Pi}{2} < \phi < \frac{\Pi}{2}\right), \quad \left(-\frac{\Pi}{2} < \theta < \frac{\Pi}{2}\right)$$

while the yaw angle ψ is unconstrained.

The inertia matrix in the body frame is:

$$\mathbf{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

¹Consequently, no acrobatic maneuvers and inverted flight are allowed.

and we can assume $I_{xx} = I_{yy}$ due to the symmetric structure.

The dynamic model of the quadrotor is defined starting from the forces acting on it:

- the generalized forces, computed starting from the forces generated by every rotor;
- the drag force, proportional to the linear and angular velocity;
- the weight, applied to the center of gravity along the negative z axis of Earth frame.

The complete model of a quadrotor is:

$$\begin{cases} \ddot{x} = \frac{1}{m}[(C_\phi S_\theta C_\psi + S_\phi S_\psi)u_f - k_t \dot{x}] \\ \ddot{y} = \frac{1}{m}[(C_\phi S_\theta S_\psi - S_\phi C_\psi)u_f - k_t \dot{y}] \\ \ddot{z} = \frac{1}{m}[(C_\phi C_\theta)u_f - mg - k_t \dot{z}] \\ \dot{p} = \frac{1}{I_{xx}}[-k_r p - qr(I_{zz} - I_{yy}) + \tau_p] \\ \dot{q} = \frac{1}{I_{yy}}[-k_r q - pr(I_{xx} - I_{zz}) + \tau_q] \\ \dot{r} = \frac{1}{I_{zz}}[-k_r r - pq(I_{yy} - I_{xx}) + \tau_r] \\ \dot{\phi} = p + q S_\phi T_\theta + r C_\phi T_\theta \\ \dot{\theta} = q C_\phi - r S_\phi \\ \dot{\psi} = \frac{1}{C_\theta}[q S_\phi + r C_\phi] \end{cases} \quad (1)$$

where:

- p, q, r = roll, pitch and yaw velocity, rad/s ;
- k_r = rotational drag, $N \cdot m \cdot s$;
- k_t = linear drag, $N \cdot m/s$;
- m = quadrotor mass, kg ;
- g = gravity acceleration, $9.81m/s^2$
- u_f = total upward lift force, N ;
- τ_p, τ_q, τ_r = torque around roll, pitch and yaw axis, $N \cdot m$;

and $C(\cdot), S(\cdot), T(\cdot)$ stand for $\cos(\cdot), \sin(\cdot), \tan(\cdot)$.

In (1) the system inputs are the generalized forces $u_f, \tau_p, \tau_q, \tau_r$, while the real actuation is given by the four motors forces f_1, f_2, f_3, f_4 .

The relation between these is given by:

$$\begin{bmatrix} u_f \\ \tau_p \\ \tau_q \\ \tau_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -l & 0 & l \\ -l & 0 & l & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (2)$$

in which:

- d = the ratio between drag and thrust coefficients of the rotor, m ;
- l = the arm length, m .

Every f_i is applied to the center of the i -rotor and is directed along the z_b axis. From (2) is clear that the total lift u_f is equal to $\sum_i f_i$. The torques around roll and pitch axis are due to the torque of every motor, $l \cdot f_i$. The torque around the yaw axis, instead, is caused by unbalanced clockwise and counterclockwise drag reaction torques.

2 Controller architecture

As mentioned in the introduction, usually the control strategy of a quadrotor is based on a double-loop architecture: an outer controller that takes the reference trajectory and computes the attitude angles needed to achieve it and an inner controller whose task is to regulate these attitude angles acting on the generalized forces.

In case of failure of one motor, the quadrotor loses the capability to control one of the variable among roll, pitch, yaw and altitude, or, in other words, it loses the ability to control independently the three torques required for the control of the attitude.

The controller developed in [1] is based on the assumption that one rotor has failed, while the others are full working. In this case, the controller task is to safely landing, instead of achieving the mission.

In [1] and [4], the authors claim that controlling a subset of attitude dynamics is enough to make the vehicle fly, although it is spinning around the yaw axis. The most important variables to control are roll, pitch and altitude: roll and pitch are essential because a change in their values results in a horizontal displacement; while altitude is fundamental in order to avoid collision with the ground.

For these reasons the yaw state is the one sacrificed.

Using only three rotors, the unbalance in propeller drag makes the quadrotor spinning around the yaw axis. To make the quadrotor increase its altitude it is needed to increase

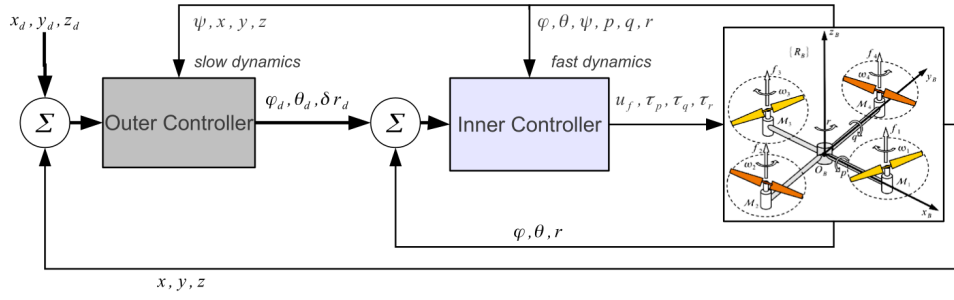


Figure 2: The double-loop architecture in case of actuator loss.

the total lift u_f : the consequence is an acceleration around the yaw axis. The rotational speed around the vertical axis and the altitude variation depends on the same input.

For this reason the inner control loop is designed to control the roll and pitch angles and yaw velocity, while the outer controller sets the desired values of these variables. In particular, the desired altitude is obtained changing the set point of the yaw velocity.

To control the quadrotor all the states must be accessible.

2.1 Quadrotor model in case of rotor loss

2.1.1 Control inputs constraints

In case of failure of rotor \mathcal{M}_2 the mapping (2) becomes:

$$\begin{bmatrix} u_f \\ \tau_q \\ \tau_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -l & l & 0 \\ d & d & -d \end{bmatrix} \begin{bmatrix} f_1 \\ f_3 \\ f_4 \end{bmatrix} \quad (3)$$

In this case the control inputs are u_f , τ_q , τ_r , while τ_p is no longer used as input. The constraints on the real inputs f_i are:

$$\begin{cases} f_i \geq 0 \\ f_i \leq f_{max} \end{cases} \quad i = 1, 3, 4 \quad (4)$$

We need to obtain the constraints on the control inputs u_f , τ_q , τ_r . From (3)

$$\begin{bmatrix} f_1 \\ f_3 \\ f_4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & -2/l & l/d \\ 1 & 2/l & l/d \\ 2 & 0 & -2/d \end{bmatrix} \begin{bmatrix} u_f \\ \tau_q \\ \tau_r \end{bmatrix} \quad (5)$$

and imposing the first of (4)

$$\begin{cases} u_f - \frac{2\tau_q}{l} + \frac{\tau_r}{d} \geq 0 \\ u_f + \frac{2\tau_q}{l} + \frac{\tau_r}{d} \geq 0 \\ 2u_f - \frac{2\tau_r}{l} \geq 0 \end{cases} \Rightarrow \begin{cases} |\tau_q| \leq \frac{l}{2}(u_f + \frac{\tau_r}{d}) \\ |\tau_r| \leq d \cdot u_f \end{cases} \quad (6)$$

to which we add:

$$u_f \leq 3 \cdot f_{max} \quad (7)$$

from the second of (4).

2.1.2 State-space model

The state vector is chosen as:

$$\begin{aligned} \mathbf{x} &= [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12}]^T \\ &= [\phi \quad \theta \quad \psi \quad p \quad q \quad r \quad x \quad y \quad z \quad \dot{x} \quad \dot{y} \quad \dot{z}]^T \end{aligned}$$

and the input vector is:

$$\mathbf{u} = [u_1 \quad u_2 \quad u_3]^T = [u_f \quad \tau_q \quad \tau_r]^T \quad (8)$$

Since $f_2 = 0$ and (2), the neglected input can be expressed as $\tau_p = l \cdot f_4$, and hence:

$$f_4 = \frac{1}{2}(u_1 - \frac{u_3}{d}) \Rightarrow \tau_p = \frac{l}{2}(u_1 - \frac{u_3}{d}) \quad (9)$$

The complete quadrotor model (1) in state-space form can be written as:

$$\dot{\phi} = \dot{x}_1 = x_4 + x_5 S_{x_1} T_{x_2} + x_6 C_{x_1} T_{x_2} \quad (10a)$$

$$\dot{\theta} = \dot{x}_2 = x_5 C_{x_1} \quad (10b)$$

$$\dot{\psi} = \dot{x}_3 = \frac{1}{C_{x_2}} [x_5 S_{x_1} + x_6 C_{x_1}] \quad (10c)$$

$$\dot{p} = \dot{x}_4 = \frac{1}{I_{xx}} [-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + \frac{l}{2} (u_1 - \frac{u_3}{d})] \quad (10d)$$

$$\dot{q} = \dot{x}_5 = \frac{1}{I_{xx}} [-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + u_2] \quad (10e)$$

$$\dot{r} = \dot{x}_6 = \frac{1}{I_{xx}} (-k_r x_6 + u_3) \quad (10f)$$

$$\dot{x} = \dot{x}_7 = x_{10} \quad (10g)$$

$$\dot{y} = \dot{x}_8 = x_{11} \quad (10h)$$

$$\dot{z} = \dot{x}_9 = x_{12} \quad (10i)$$

$$\ddot{x} = \dot{x}_{10} = \frac{C_{x_1} S_{x_2} C_{x_3} + S_{x_1} S_{x_3}}{m} u_1 - \frac{k_t}{m} x_{10} \quad (10j)$$

$$\ddot{y} = \dot{x}_{11} = \frac{C_{x_1} S_{x_2} S_{x_3} - S_{x_1} C_{x_3}}{m} u_1 - \frac{k_t}{m} x_{11} \quad (10k)$$

$$\ddot{z} = \dot{x}_{12} = \frac{1}{m} [u_1 C_{x_1} C_{x_2} - k_t x_{12} - mg] \quad (10l)$$

where in (10f) I_{xx} is chosen to be equal to I_{yy} due to the symmetry of the vehicle.

2.2 Outer control loop

The horizontal motion depends on the projection of the thrust vector on the (x, y) plane. The direction of this vector depends on the roll and pitch angles.

The task of the outer control is to generate the right values of roll and pitch angles to reach a desired position in the Earth frame. The vertical dynamics, on the other side, is controlled by the variation of the yaw velocity r around a working point².

Assuming that the inner controller is stabilizing the state near the equilibrium:

$$x_1 = \phi \rightarrow \phi_d, \quad x_2 = \theta \rightarrow \theta_d, \quad x_6 = r \rightarrow r_d \quad (11)$$

the chosen desired values ϕ_d and θ_d must be small.

We compute the inputs at the steady state:

²We will discuss this working point in the inner control loop.

- from (10e):

$$\dot{q} = \dot{x}_5 = \frac{1}{I_{xx}}[-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + u_2] = 0 \quad \Rightarrow \quad u_2 \rightarrow 0 \quad (12)$$

- from (10f):

$$\dot{r} = \dot{x}_6 = \frac{1}{I_{xx}}(-k_r x_6 + u_3) = 0 \quad \Rightarrow \quad u_3 \rightarrow x_{6d} k_r \quad (13)$$

- from (10d):

$$\begin{aligned} \dot{p} = \dot{x}_4 &= \frac{1}{I_{xx}}[-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + \frac{l}{2}(u_1 - \frac{u_3}{d})] = 0 \\ \Rightarrow \quad u_1 &\rightarrow \frac{u_3}{d} = \frac{x_{6d} k_r}{d} \end{aligned} \quad (14)$$

Using the assumption of small angles for ϕ_d and θ_d , the model relative to the outer control loop can be expressed as:

$$\dot{x}_7 = x_{10} \quad (15a)$$

$$\dot{x}_8 = x_{11} \quad (15b)$$

$$\dot{x}_9 = x_{12} \quad (15c)$$

$$\dot{x}_{10} = \frac{x_{2d} C_{x_3} + x_{1d} S_{x_3}}{m} \overbrace{\frac{k_r}{d} x_{6d}}^{u_1} - \frac{k_t}{m} x_{10} \quad (15d)$$

$$\dot{x}_{11} = \frac{x_{2d} S_{x_3} + x_{1d} C_{x_3}}{m} \overbrace{\frac{k_r}{d} x_{6d}}^{u_1} - \frac{k_t}{m} x_{11} \quad (15e)$$

$$\dot{x}_{12} = \frac{1}{m} \left[\overbrace{\frac{k_r}{d} x_{6d}}^{u_1} - k_t x_{12} - mg \right] \quad (15f)$$

The state of the outer control loop is chosen as:

$$\begin{aligned} \tilde{\mathbf{x}} &= [\tilde{x}_1 \quad \tilde{x}_2 \quad \tilde{x}_3 \quad \tilde{x}_4 \quad \tilde{x}_5 \quad \tilde{x}_6]^T \\ &= [x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12}]^T \end{aligned}$$

While the control inputs vector is chosen so as to linearize the state-space equations:

$$\tilde{\mathbf{u}} = [\tilde{u}_1 \quad \tilde{u}_2 \quad \tilde{u}_3] = \left[\frac{k_r x_{6d}}{dm} x_{1d} \quad \frac{k_r x_{6d}}{dm} x_{2d} \quad \frac{k_r x_{6d}}{dm} \right] \quad (16)$$

The equations (15) can be rewritten as:

$$\dot{\tilde{x}}_1 = \tilde{x}_4 \quad (17a)$$

$$\dot{\tilde{x}}_2 = \tilde{x}_5 \quad (17b)$$

$$\dot{\tilde{x}}_3 = \tilde{x}_6 \quad (17c)$$

$$\dot{\tilde{x}}_4 = C_\psi \tilde{u}_2 + S_\psi \tilde{u}_1 - \frac{k_t}{m} \tilde{x}_4 \quad (17d)$$

$$\dot{\tilde{x}}_5 = S_\psi \tilde{u}_2 - C_\psi \tilde{u}_1 - \frac{k_t}{m} \tilde{x}_5 \quad (17e)$$

$$\dot{\tilde{x}}_6 = \tilde{u}_3 - \frac{k_t}{m} \tilde{x}_6 - g \quad (17f)$$

The dynamics of longitudinal $(\tilde{x}_1, \tilde{x}_4)$ and lateral $(\tilde{x}_2, \tilde{x}_5)$ motion are decoupled from the altitude one $(\tilde{x}_3, \tilde{x}_6)$.

The vertical dynamics is linear and can be controlled with any standard control technique.

$\tilde{x}_1, \tilde{x}_2, \tilde{x}_4, \tilde{x}_5$ have nonlinear dynamics that depend on the yaw angle ψ .

To linearize the nonlinear dynamics we use the following control inputs:

$$\begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} = \begin{bmatrix} S_\psi & -C_\psi \\ C_\psi & S_\psi \end{bmatrix} \begin{bmatrix} \tilde{\tilde{u}}_1 \\ \tilde{\tilde{u}}_2 \end{bmatrix} \quad (18)$$

where the new control inputs $\tilde{\tilde{u}}_1, \tilde{\tilde{u}}_2$ are premultiplied by the rotation matrix.

Substituting $\tilde{\tilde{u}}_1, \tilde{\tilde{u}}_2$ in (17), the lateral and longitudinal dynamics become linear and decoupled and thus can be controlled with a simple linear control technique.

$$\dot{\tilde{x}}_4 = \tilde{\tilde{u}}_1 - \frac{k_t}{m} \tilde{x}_4 \quad (19a)$$

$$\dot{\tilde{x}}_5 = \tilde{\tilde{u}}_2 - \frac{k_t}{m} \tilde{x}_5 \quad (19b)$$

The control inputs are chosen as:

$$u_i = \ddot{x}_i + 2\xi_{out} c_{out} \dot{e}_i + c_{out}^2 e_i \quad (20)$$

where \ddot{x}_i is the feed-forward term, $e_i = x_{di} - x_i$ is the error. ξ_{out} and c_{out} are positive constants.

c_{out} represents the natural frequency of the dynamical system and must be small: actually the outer control loop must be slower than the inner control loop.

From (16) and having computed $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$ we have the desired values of $[\phi_d, \theta_d, r_d] = [x_{1d}, x_{2d}, x_{6d}]$ to use as set points for the inner control loop³.

2.3 Inner control loop

The inner control loop task is to control the dynamics of the attitude angles and the yaw velocity.

The equations in (10) relative to ψ, θ, p, q, r are expressed such that the origin of the system is an equilibrium point when the inputs are equal to zero.

The state is⁴:

$$\hat{\mathbf{x}} = [\hat{x}_1 \quad \hat{x}_2 \quad \hat{x}_3 \quad \hat{x}_4 \quad \hat{x}_5] = \left[x_1 \quad x_1 \quad x_2 \quad x_3 \quad x_4 \quad \left(x_6 - \frac{mgd}{k_r}\right) \right] \quad (21)$$

while the inputs are:

$$\hat{\mathbf{u}} = [\hat{u}_1 \quad \hat{u}_2 \quad \hat{u}_3] = [(u_1 - mg) \quad u_2 \quad (u_3 - mgd)] \quad (22)$$

reminding that $[u_1, u_2, u_3] = [u_f, \tau_q, \tau_r]$.

The $\hat{\mathbf{x}}$ dynamics are given by:

$$\dot{\hat{\mathbf{x}}} = \hat{\mathbf{f}}(\hat{\mathbf{x}}) + \hat{\mathbf{G}}(\hat{\mathbf{x}})\hat{\mathbf{u}} \quad (23)$$

and the output is:

$$\hat{\mathbf{y}} = [\hat{h}_1 \quad \hat{h}_2 \quad \hat{h}_3] = [\hat{x}_1 \quad \hat{x}_2 \quad \hat{x}_5] \quad (24)$$

³Note that, at equilibrium, $\tilde{u}_3 \rightarrow g$ and hence $r_d \rightarrow \frac{mgd}{k_r}$, while ϕ_d and θ_d remain small angle.

⁴In the previous section we have seen that at equilibrium $r \rightarrow \frac{mgd}{k_r}$.

where:

$$\hat{\mathbf{f}}(\hat{\mathbf{x}}) = \begin{bmatrix} \hat{x}_3 + \hat{x}_4 S_{\hat{x}_1} T_{\hat{x}_2} + (\hat{x}_5 + \frac{mgd}{k_r}) C_{\hat{x}_1} T_{\hat{x}_2} \\ \hat{x}_4 C_{\hat{x}_1} - (\hat{x}_5 + \frac{mgd}{k_r}) S_{\hat{x}_1} \\ \frac{1}{I_{xx}} [-k_r \hat{x}_3 - \hat{x}_4 (\hat{x}_5 + \frac{mgd}{k_r}) (I_{zz} - I_{yy})] \\ \frac{1}{I_{yy}} [-k_r \hat{x}_4 - \hat{x}_3 (\hat{x}_5 + \frac{mgd}{k_r}) (I_{xx} - I_{zz})] \\ \frac{1}{I_{zz}} (-k_r \hat{x}_5) \end{bmatrix} \quad (25)$$

$$\hat{\mathbf{G}}(\hat{\mathbf{x}}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{l}{2I_{xx}} & 0 & -\frac{l}{2I_{xx}} \\ 0 & \frac{1}{I_{xx}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \quad (26)$$

2.3.1 Feedback Linearization

The system fulfills the condition for the feedback linearization:

- the relative degrees are $r_1 = 2$ (h_1), $r_2 = 2$ (h_2), $r_3 = 1$ (h_3). $r_1 + r_2 + r_3 = n = 5$, the dimension of the state vector;
- the decoupling matrix:

$$\mathbf{M}(\hat{\mathbf{x}}) = \begin{bmatrix} \frac{l}{2I_{xx}} & \frac{S_{\hat{x}_1} T_{\hat{x}_2}}{\frac{I_{xx}}{C_{\hat{x}_1}}} & \frac{C_{\hat{x}_1} T_{\hat{x}_2}}{I_{zz}} - \frac{l}{2dI_{xx}} \\ 0 & \frac{C_{\hat{x}_1}}{I_{xx}} & -\frac{S_{\hat{x}_1}}{I_{zz}} \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \quad (27)$$

is always invertible.

We can compute the linearizing input as:

$$\mathbf{u}(\mathbf{x}, \mathbf{v}) = \alpha(\mathbf{x}) + \beta(\mathbf{x}) * \mathbf{v} \quad (28)$$

where $\alpha(\mathbf{x}) = [\alpha_1(\mathbf{x}) \quad \alpha_2(\mathbf{x}) \quad \alpha_3(\mathbf{x})]$ is:

$$\begin{aligned} \alpha_1(\mathbf{x}) = & (2dk_r^3\hat{x}_3C_{\hat{x}_1}C_{\hat{x}_2}^2 + k_r^3l\hat{x}_5C_{\hat{x}_1}C_{\hat{x}_2}^2 - 2I_{xx}dk_r^2\hat{x}_4^2S_{\hat{x}_1} - 4I_{xx}dk_r^2\hat{x}_4\hat{x}_5C_{\hat{x}_1}^3 \\ & + 2I_{xx}d^3g^2m^2C_{\hat{x}_1}^2S_{\hat{x}_1} - 2I_{xx}dk_r^2\hat{x}_4^2C_{\hat{x}_1}^2S_{\hat{x}_1} + 2I_{xx}dk_r^2\hat{x}_4^2C_{\hat{x}_2}^2S_{\hat{x}_1} \\ & + 2I_{xx}dk_r^2\hat{x}_5^2C_{\hat{x}_1}^2S_{\hat{x}_1} - 4I_{xx}d^2gk_rm\hat{x}_4C_{\hat{x}_1}^3 - 2I_{xx}dk_r^2\hat{x}_3\hat{x}_4C_{\hat{x}_2}S_{\hat{x}_2} + 2I_{zz}dk_r^2\hat{x}_4\hat{x}_5C_{\hat{x}_1}C_{\hat{x}_2}^2 \\ & + 2I_{zz}d^2gk_rm\hat{x}_4C_{\hat{x}_1}C_{\hat{x}_2}^2 + 4I_{xx}d^2gk_rm\hat{x}_5C_{\hat{x}_1}^2S_{\hat{x}_1}) / (dk_r^2lC_{\hat{x}_1}C_{\hat{x}_2}^2) \end{aligned} \quad (29)$$

$$\begin{aligned} \alpha_2(\mathbf{x}) = & (I_{xx}(C_{\hat{x}_1}(\hat{x}_5 + (dgm)/k_r) + \hat{x}_4S_{\hat{x}_1})(\hat{x}_3 + C_{\hat{x}_1}T_{\hat{x}_2}(\hat{x}_5 + (dgm)/k_r) + \hat{x}_4S_{\hat{x}_1}T_{\hat{x}_2}) \\ & + C_{\hat{x}_1}(k_r\hat{x}_4 + \hat{x}_3(\hat{x}_5 + (dgm)/k_r)(I_{xx} - I_{zz}))) / I_{xx} \\ & - (k_r\hat{x}_5S_{\hat{x}_1}/I_{zz}) / C_{\hat{x}_1} + (I_{xx}k_r\hat{x}_5S_{\hat{x}_1}) / (I_{zz}C_{\hat{x}_1}) \end{aligned} \quad (30)$$

$$\alpha_3(\mathbf{x}) = k_r\hat{x}_5 \quad (31)$$

and $\beta(\mathbf{x})$:

$$\beta(\mathbf{x}) = \begin{bmatrix} 2\frac{I_{xx}}{l} & -2\frac{I_{xx}T_{\hat{x}_1}T_{\hat{x}_2}}{l} & -\frac{2I_{xx}T_{\hat{x}_2}}{lC_{\hat{x}_1}} + \frac{I_{zz}T_{\hat{x}_1}}{d} \\ 0 & \frac{I_{xx}}{C_{\hat{x}_1}} & I_{xx}T_{\hat{x}_1} \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (32)$$

The (28) leads to the canonical Brunowski form:

$$\dot{\mathbf{x}}_{\mathbf{c}} = \mathbf{A}_{\mathbf{c}}\mathbf{x}_{\mathbf{c}} + \mathbf{B}_{\mathbf{c}}\mathbf{v} \quad (33)$$

$$\mathbf{A}_{\mathbf{c}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{B}_{\mathbf{c}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (34)$$

and the state vector $\mathbf{x}_{\mathbf{c}} = [\hat{h}_1 \quad \hat{h}_2 \quad \dot{\hat{h}}_1 \quad \dot{\hat{h}}_2 \quad \hat{h}_3] = [\hat{x}_1 \quad \hat{x}_2 \quad \dot{\hat{x}}_1 \quad \dot{\hat{x}}_2 \quad \hat{x}_5]$.

The linear input \mathbf{v} is chosen to regulate the state variables $\hat{x}_1, \hat{x}_2, \hat{x}_5$ to the desired values, coming from the outer control loop.

Usually, the desired roll and pitch angles and the variation of yaw velocity are chosen to converge to zero. As a result, the roll velocity p and the pitch velocity q will converge to zero and the yaw velocity r will converge to the constant value mgd/k_r .

3 Simulations

In this section are presented the behaviors of the outer control loop, the inner control loop and then of the overall system when both the inner and the outer control loops are activated. The model parameters used in the simulations are in Table.1.

Constant	Value	Description
m	0.5 kg	Quadrotor mass
g	9.81 m/ s ²	Gravity acceleration
I_{xx}	$5.9 \cdot 10^{-3}$ kg m ²	Inertia coefficient along x axis
I_{yy}	$5.9 \cdot 10^{-3}$ kg m ²	Inertia coefficient along y axis
I_{zz}	$1.16 \cdot 10^{-3}$ kg m ²	Inertia coefficient along z axis
l	0.255 m	Distance between rotor and c.g.
d	$2.4 \cdot 10^{-3}$ kg m ²	Blade drag / thrust ratio
k_r	0.1 N m s	Rotational drag
k_t	0.1 N s/m	Linear drag

Table 1: Quadrotor and controller parameters.

Outer control loop

In the first simulation only the outer control loop is activated. Specifically we want to regulate the position from the initial condition, expressed in Table 2, to the origin of the Earth frame.

Variable	Value	Description
\tilde{x}_1	2 m	x
\tilde{x}_2	3 m	y
\tilde{x}_3	4 m	z
\tilde{x}_4	0 m/s	\dot{x}
\tilde{x}_5	0 m/s	\dot{y}
\tilde{x}_6	0 m/s	\dot{z}

Table 2: Initial conditions for the outer control loop simulation.

Since we don't want the initial error to be too large (and hence, the inputs), a module

has been used that computes a cubic trajectory that lasts 50 seconds.

We set a constant spinning velocity equal to 0.3 rad/s. As shown in fig.3 the outer control loop tracks the reference trajectory perfectly. The reference values coming out from the outer control loop inputs in eq.(16) are shown in fig.4.

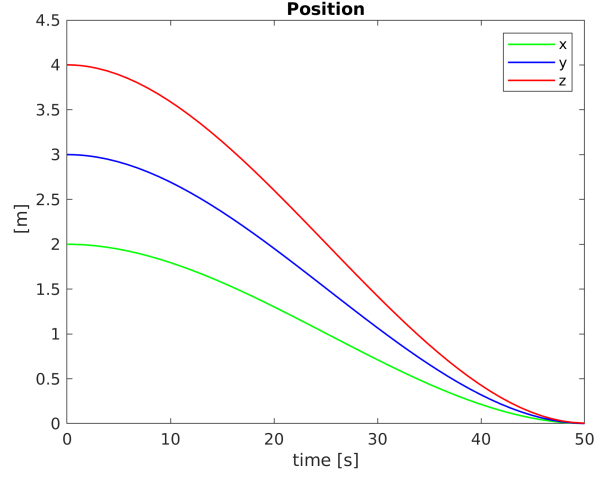
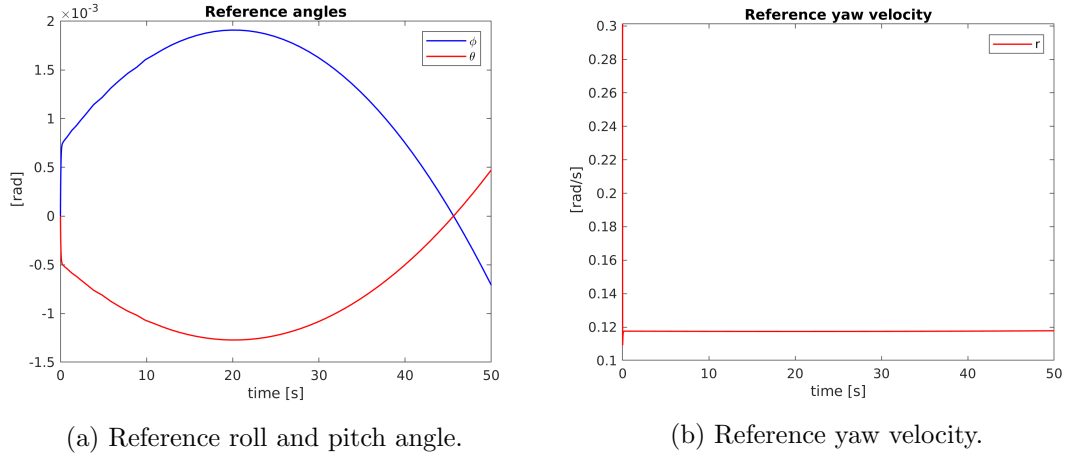


Figure 3: Position in the Earth frame.



(a) Reference roll and pitch angle.

(b) Reference yaw velocity.

Figure 4: Reference values with cubic trajectory.

For comparison we show also the plot relative to the simulation without the generation of the cubic trajectory.

Also in this case the position is quickly regulated (fig.5). However, the values coming out from the cubic trajectory generation are smoother and smaller, and hence preferable.

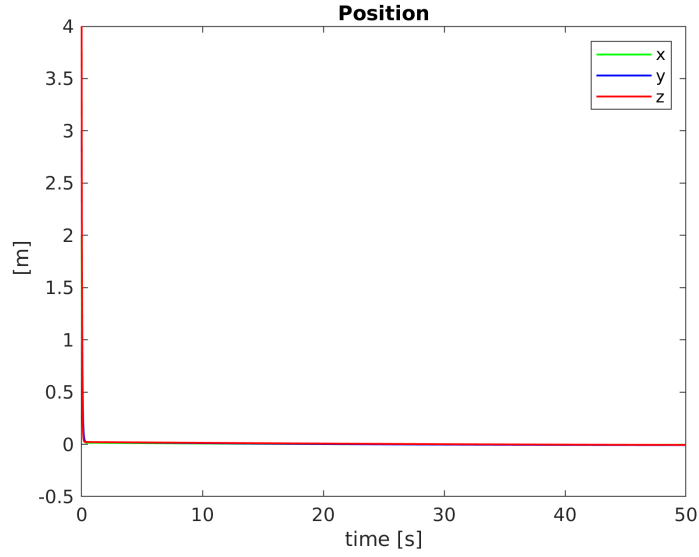


Figure 5: Position in the Earth frame.

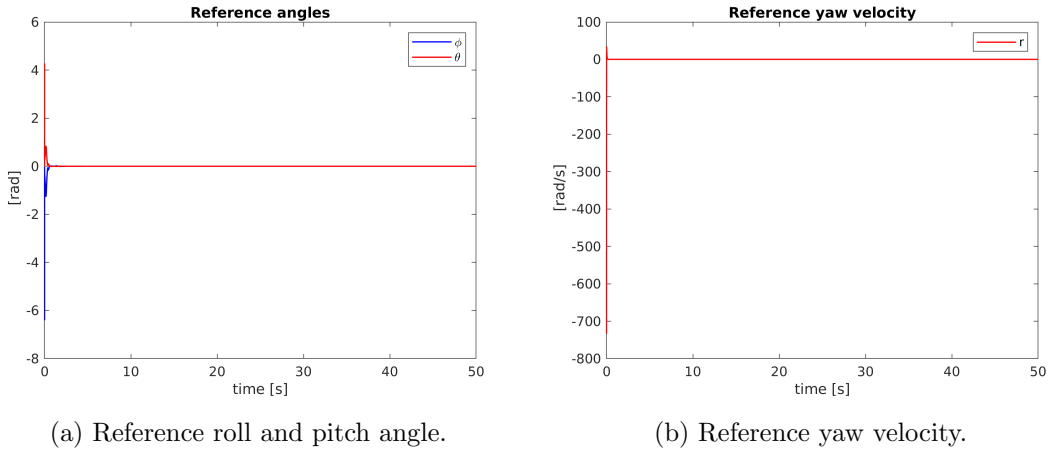


Figure 6: Reference values with no cubic trajectory.

Inner control loop

In the second simulation the task of the inner control loop is to regulate the state around a set point defined in Table 3 from the initial condition in Table 4.

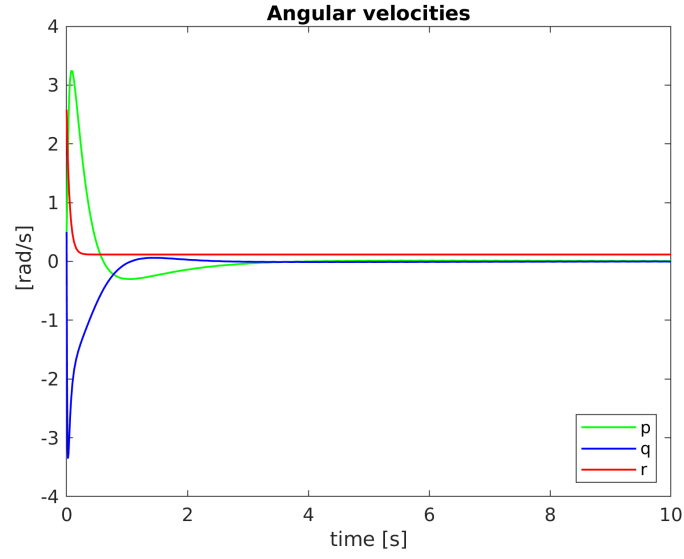
The simulation time is 5 seconds. As shown in fig.7a and fig.7b the values of both angles and velocities are perfectly regulated. In particular the spinning velocity converges to the value $\frac{mgd}{k_r}$, relative to the hovering condition.

Variable	Value	Description
x_1	0 rad	ϕ
x_2	0 rad	θ
x_3	0 rad	ψ
x_4	0 rad/s	p
x_5	0 rad/s	q
x_6	mgd/k_r rad/s	r

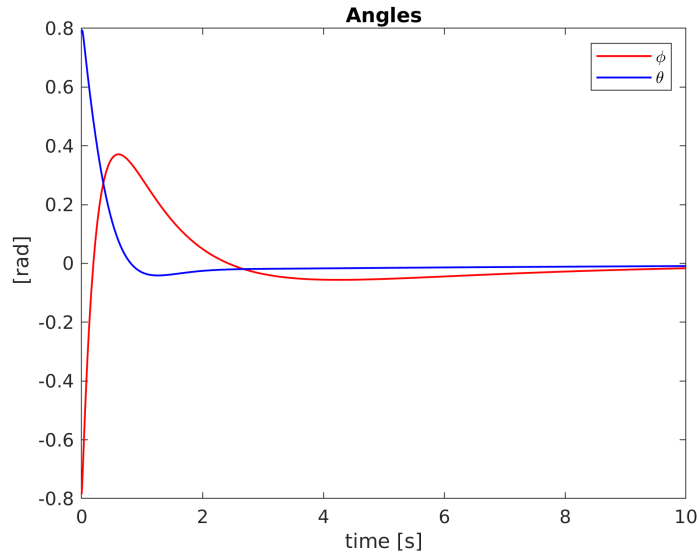
Table 3: Final set point for the inner control loop simulation.

Variable	Value	Description
x_1	$-\pi/4$ rad	ϕ
x_2	$\pi/4$ rad	θ
x_3	$\pi/4$ rad	ψ
x_4	-0.5 rad/s	p
x_5	0.5 rad/s	q
x_6	2.7 rad/s	r

Table 4: Initial conditions for the inner control loop simulation.



(a)



(b)

Figure 7: Values of angles and velocities relative to the inner control loop.

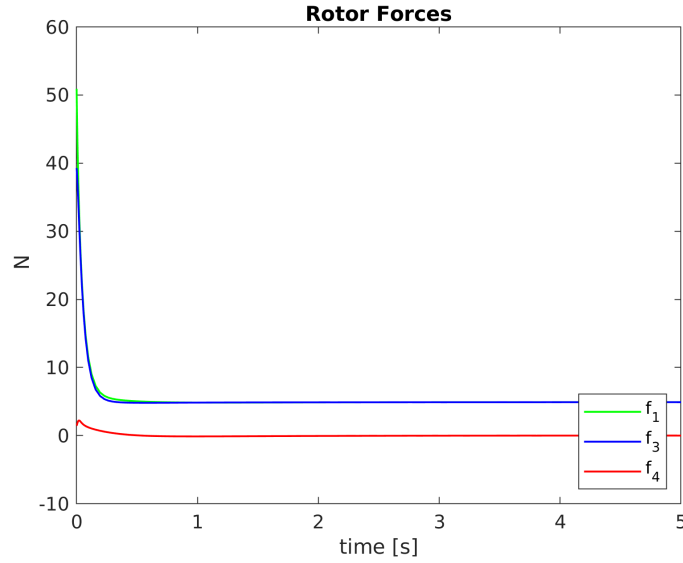


Figure 8: Forces computed from the inputs of the inner control loop.

The rotor forces, after an initial peak due to the pretty large error, remains constant and assume small values (fig.8). In particular f_1 and f_3 are used to sustain the vehicle in hovering flight and f_4 deals with the regulation of spinning velocity.

Complete simulation

In the simulations of this section, both the outer and the inner control loop are activated.

The simulation time is 100 seconds. The controller must fly the quadrotor from the initial condition in Table 5 to the origin of the Earth frame.

As shown in fig.9, the vertical trajectory of the vehicle is perfectly tracked (red line). Instead, the horizontal position (green and blue lines) oscillates around the planned trajectory: the outer loop control has small gains so as to be slow enough. It correct the error not aggressively, generating reference angles that do not change quickly.

The inner control loop regulates successfully the reference angles. In fig.10 it is clear how the error between the desired and the actual roll and pitch angles is suddenly reduced to zero.

Also in this case the forces, after an initial peak, become constant. In particular \mathcal{M}_1 and \mathcal{M}_3 rotors sustain the vehicle. The rotor \mathcal{M}_4 provides a periodical thrust (fig.12 - zoom-in): the outer controller generates a time varying references for the inner control

Variable	Value	Description
x_1	0.1 rad	ϕ
x_2	-0.1 rad	θ
x_3	0.1 rad	ψ
x_4	0.2 rad/s	p
x_5	-0.2 rad/s	q
x_6	0.15 rad/s	r
x_7	5 m	x
x_8	5 m	y
x_9	5 m	z
x_{10}	0 m/s	\dot{x}
x_{11}	0 m/s	\dot{y}
x_{12}	0 m/s	\dot{z}

Table 5: Initial conditions for the complete simulation.

loop that depends on the rotation matrix in eq.(18) in order to achieve the desired horizontal translation. To this effect, the rotor \mathcal{M}_4 tries to regulate the direction of the total thrust, allowing the translation.

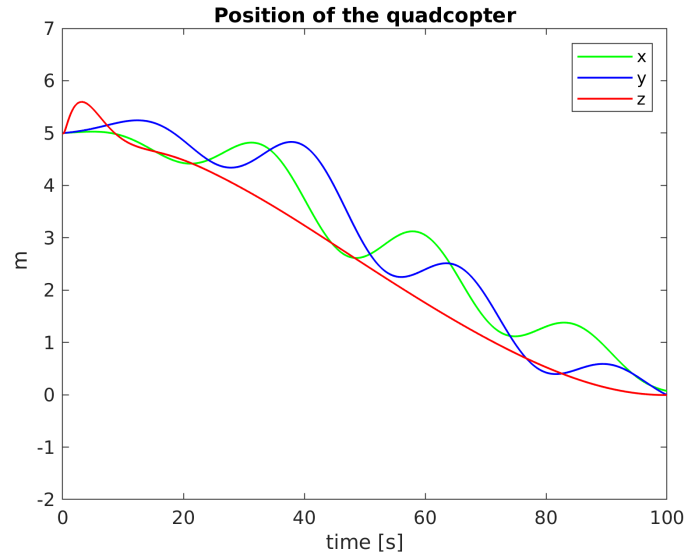


Figure 9: Position of the quadcopter in the Earth frame.

The performances of the double-loop controller decrease when the constraints on motors force saturation are applied. The controller behaves correctly when the vehicle starts with zero error conditions.

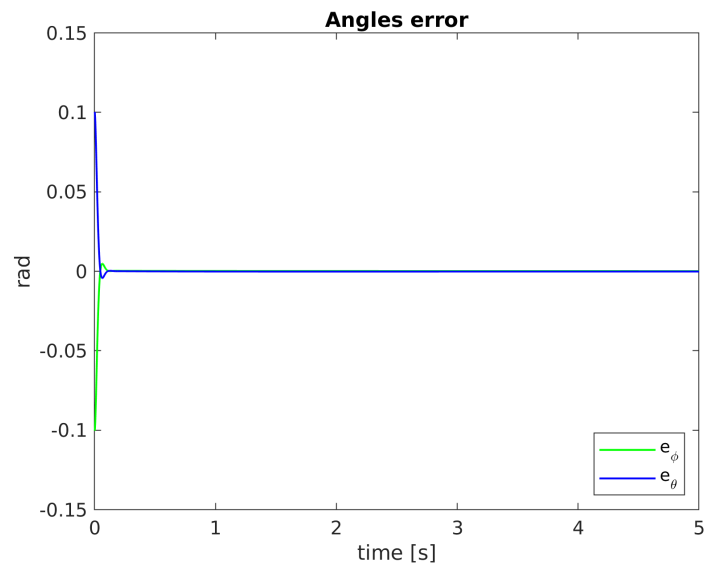


Figure 10: Error between the desired and the actual roll and pitch angles - only the first 5 seconds.

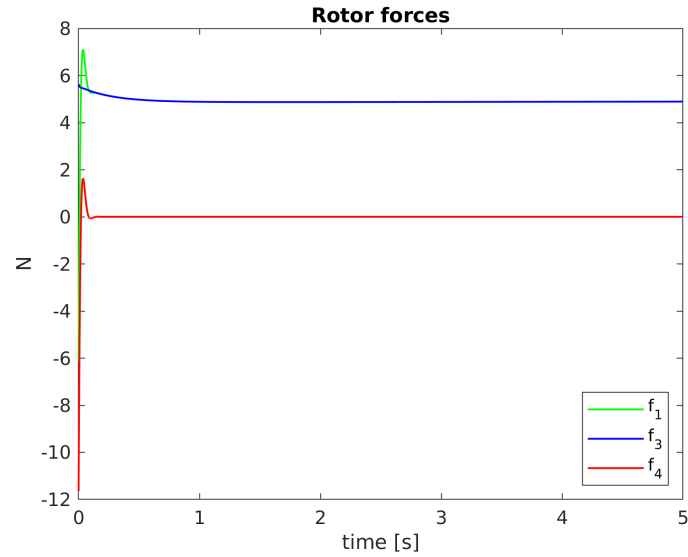


Figure 11: Forces required by the double-loop controller - only the first 5 seconds.

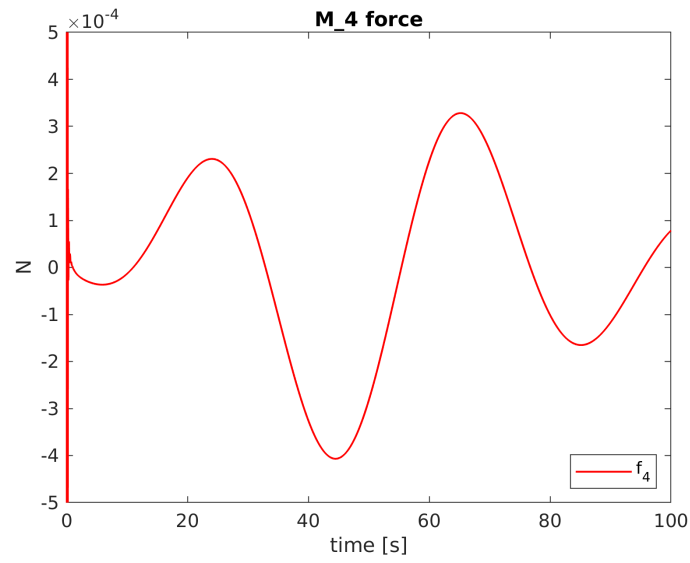


Figure 12: Force of rotor n.4.

In this case the vehicle follows the vertical trajectory (fig.13) and the forces required by the controller are always limited in the boundaries (fig.14). The explanation of this behavior is that the quadcopter has an initial small spinning velocity around the equilibrium and by consequence the motor \mathcal{M}_4 is never activated. The vertical position, instead, is regulated only by motors \mathcal{M}_1 and \mathcal{M}_3 .

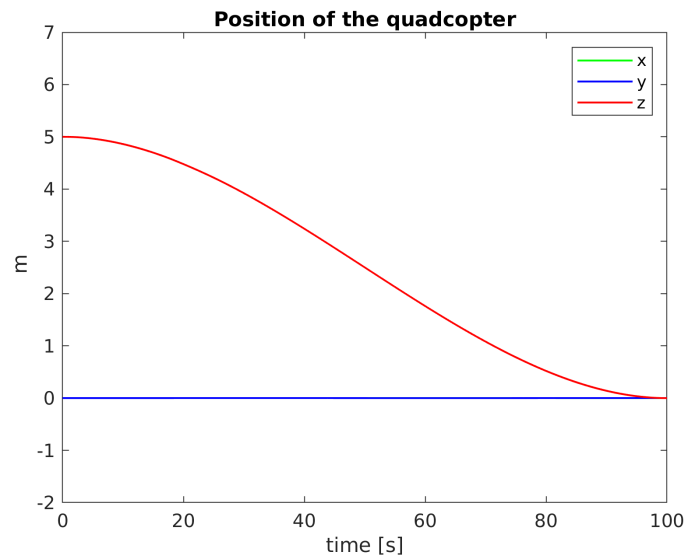


Figure 13: Position of the quadcopter in Earth frame (force saturation active).

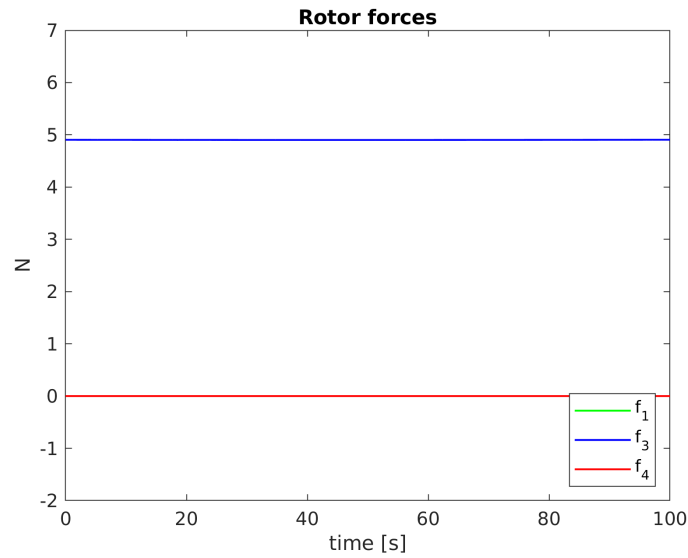


Figure 14: Forces required by the double-loop architecture (force saturation active).

4 Conclusions

In this report we addressed the problem of controlling a quadcopter in case of loss of one of the actuators. The adopted strategy is the use of a double-loop architecture.

In correspondence of the loss of one input, the state that is chosen to sacrifice is the yaw angle.

The outer control loop is based on a simple linearizing control law. The inner control loop is based on a feedback linearization control law.

The developed controllers behave appropriately singularly. When interconnected the vertical trajectory is perfectly tracked, while the horizontal position oscillates around the desired one, allowing only slow recovery maneuvers.

Another problem is represented by the motors force saturation: we can consider it as an uncertainty in the model. In this sense, the controller would require a more robust approach as proposed in [1]. Another solution is to consider saturation in the control design. For example, using an MPC one can introduce constraints relative to reference angle rate in the outer control loop and relative to forces in the inner control loop, and hence obtain inputs satisfying the boundaries imposed.

References

- [1] A. Lanzon, A. Freddi, and S. Longhi, “Flight control of a quadrotor vehicle subsequent to a rotor failure,” *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 2, pp. 580–591, 2014.
- [2] A. Isidori, *Nonlinear control systems*. Springer Science & Business Media, 2013.
- [3] J.-J. E. Slotine, W. Li, *et al.*, *Applied nonlinear control*, vol. 199. Prentice hall Englewood Cliffs, NJ, 1991.
- [4] A. Freddi, A. Lanzon, and S. Longhi, “A feedback linearization approach to fault tolerance in quadrotor vehicles,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 5413–5418, 2011.