Sapienza Università di Roma

Flight control of a Quadrotor Vehicle Subsequent to a Rotor Failure

Edoardo Ghini, Gianluca Cerilli, Giuseppe L'Erario

Modeling and control of multi-rotor UAVs

Table of contents

Introduction

Problem statement

Development of a control scheme in case of **rotor failure**, to allow the quadrotor to land with only three actuators.

Proposed Strategy

Full control of attitude is lost: renounce the yaw angle control, maintaining flight control of a spinning vehicle

Double-loop architecture

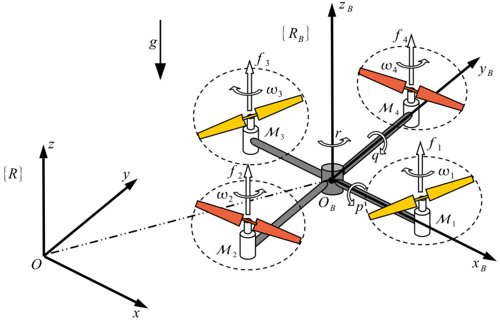
- *Outer Control*
  Yaw velocity is computed instead of the yaw angle

- *Inner Control*
  Roll angle, pitch angle and yaw velocity regulated via feedback linearization

Quadrotor Model

Control Model

- Symmetric
- Rigid propellers
- Linear drag



Two reference frames:

- Earth frame: $\{R\}\{O, x, y, z\}$
- Body-fixed frame: $\{R_b\}\{O_b, x_b, y_b, z_b\}$

Control Model

Rotations along roll (ϕ) and pitch (θ) angles are constrained:

$$(-\frac{\Pi}{2} < \phi < \frac{\Pi}{2}), \qquad (-\frac{\Pi}{2} < \theta < \frac{\Pi}{2})$$

while yaw angle ψ is unconstrained.

Inertia matrix in the body frame:

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

where $I_{xx} = I_{yy}$ due to symmetry.

Control Model

Dynamic model, defined from generalized forces, drag force and weight:

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

- k_r, k_t = rotational and linear drags, $N \cdot m \cdot s$;
- u_f = total upward lift force, N;
- $\tau_p,\ \tau_q,\ \tau_r$ = torque around roll, pitch and yaw axis, $N \cdot m$;

Control Model

System inputs:

Generalized forces $u_f, \ \tau_p, \ \tau_q, \ \tau_r$

Real actuation:

Four motors forces $f_1, \ f_2, \ f_3, \ f_4$

Relation between these is given by:

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

where

- $d =$ ratio between drag and thrust coefficient of the rotor, m
- $l =$ arm length, m

Control Architecture

Control scheme

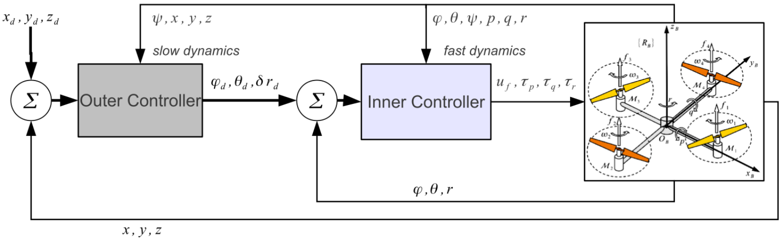


Figure 1: The double loop architecture in case of actuator loss.

Generalized force - real actuation mapping

In case of failure of rotor $\mathcal{M}_2$ the mapping (1) is:

$$\begin{bmatrix} u_f \\ \tau_q \\ \tau_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -l & l & 0 \\ d & d & -d \end{bmatrix} \begin{bmatrix} f_1 \\ f_3 \\ f_4 \end{bmatrix} \tag{2}$$

and the neglected input is $\tau_p = l \cdot f_4$

Generalized force - real actuation mapping

In case of failure of rotor $\mathcal{M}_2$ the mapping (1) is:

$$\begin{bmatrix} u_f \\ \tau_q \\ \tau_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ -l & l & 0 \\ d & d & -d \end{bmatrix} \begin{bmatrix} f_1 \\ f_3 \\ f_4 \end{bmatrix} \tag{2}$$

and the neglected input is $\tau_p = l \cdot f_4$

The inverse of (2) is:

$$\begin{bmatrix} f_1 \\ f_3 \\ f_4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & -2/l & l/d \\ 1 & 2/l & l/d \\ 2 & 0 & -2/d \end{bmatrix} \begin{bmatrix} u_f \\ \tau_q \\ \tau_r \end{bmatrix} \tag{3}$$

State-space model

State vector

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

Input vector

$$\mathbf{u} = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T = \begin{bmatrix} u_f & \tau_q & \tau_r \end{bmatrix}^T \tag{4}$$

The neglected input is expressed as:

$$\tau_p = \frac{l}{2}(u_1 - \frac{u_3}{d}) \tag{5}$$

$$
\begin{cases}
\dot{\phi} & = \dot{x}_1 = x_4 + x_5 S_{x_1} T_{x_2} + x_6 C_{x_1} T_{x_2} \\
\dot{\theta} & = \dot{x}_2 = x_5 C_{x_1} \\
\dot{\psi} & = \dot{x}_3 = \frac{1}{C_{x_2}}[x_5 S_{x_1} + x_6 C_{x_1}] \\
\dot{p} & = \dot{x}_4 = \frac{1}{I_{xx}}[-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + \frac{l}{2}(u_1 - \frac{u_3}{d})] \\
\dot{q} & = \dot{x}_5 = \frac{1}{I_{xx}}[-k_r x_4 - x_5 x_6 (I_{zz} - I_{xx}) + u_2] \\
\dot{r} & = \dot{x}_6 = \frac{1}{I_{xx}}(-k_r x_6 + u_3) \\
\dot{x} & = \dot{x}_7 = x_{10} \\
\dot{y} & = \dot{x}_8 = x_{11} \\
\dot{z} & = \dot{x}_9 = x_{12} \\
\ddot{x} & = \dot{x}_{10} = \frac{C_{x_1} S_{x_2} C_{x_3} + S_{x_1} S_{x_3}}{m} u_1 - \frac{k_t}{m} x_{10} \\
\ddot{y} & = \dot{x}_{11} = \frac{C_{x_1} S_{x_2} S_{x_3} - S_{x_1} C_{x_3}}{m} u_1 - \frac{k_t}{m} x_{11} \\
\ddot{z} & = \dot{x}_{12} = \frac{1}{m}[u_1 C_{x_1} C_{x_2} - k_t x_{12} - mg]
\end{cases}
\tag{6}
$$

Outer control loop

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.

Task of the Outer control:

Generate the right values of:

- roll and pitch angles to reach a desired horizontal position;
- yaw velocity to control vertical dynamics.

Outer control loop

Assuming the inner control loop stabilizes the state near the equilibrium:

$$x_1 = \phi \to \phi_d, \qquad x_2 = \theta \to \theta_d, \qquad x_6 = r \to r_d \tag{7}$$

Inputs at the steady-state are:

$$\begin{aligned} u_2 &\to 0 \\ u_3 &\to x_{6d} k_r \\ u_1 &\to \frac{u_3}{d} = \frac{x_{6d} k_r}{d} \end{aligned}$$

Outer control loop

Rewrite the equations relative to (x, y, z) dynamics of (6) using:

1. assumption of small angles for ϕ_d and θ_d
2. inputs written at the steady-state
3. *new* state for outer control loop:

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

4. *new* control inputs vector:

$$\tilde{u} = [\tilde{u}_1 \quad \tilde{u}_2 \quad \tilde{u}_3] = \left[\frac{k_r x_{6d}}{dm} x_{1d} \quad \frac{k_r x_{6d}}{dm} x_{2d} \quad \frac{k_r x_{6d}}{dm} \right] \tag{8}$$

Outer control loop model:

$$
\begin{aligned}
\dot{\tilde{x}}_1 &= \tilde{x}_4 \\
\dot{\tilde{x}}_2 &= \tilde{x}_5 \\
\dot{\tilde{x}}_3 &= \tilde{x}_6 \\
\dot{\tilde{x}}_4 &= C_\psi \tilde{u}_2 + S_\psi \tilde{u}_1 - \frac{k_t}{m}\tilde{x}_4 \\
\dot{\tilde{x}}_5 &= S_\psi \tilde{u}_2 - C_\psi \tilde{u}_1 - \frac{k_t}{m}\tilde{x}_5 \\
\dot{\tilde{x}}_6 &= \tilde{u}_3 - \frac{k_t}{m}\tilde{x}_6 - g
\end{aligned}
$$

Outer control loop

Outer control loop model:

$$
\begin{aligned}
\dot{\tilde{x}}_1 &= \tilde{x}_4 \\
\dot{\tilde{x}}_2 &= \tilde{x}_5 \\
\dot{\tilde{x}}_3 &= \tilde{x}_6 \\
\dot{\tilde{x}}_4 &= C_\psi \tilde{u}_2 + S_\psi \tilde{u}_1 - \frac{k_t}{m}\tilde{x}_4 \\
\dot{\tilde{x}}_5 &= S_\psi \tilde{u}_2 - C_\psi \tilde{u}_1 - \frac{k_t}{m}\tilde{x}_5 \\
\dot{\tilde{x}}_6 &= \tilde{u}_3 - \frac{k_t}{m}\tilde{x}_6 - g
\end{aligned}
$$

Not yet linear!

Outer control loop

To linearize the nonlinear dynamics we manipulate the control inputs:

$$\begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} = \begin{bmatrix} S_\psi & -C_\psi \\ C_\psi & S_\psi \end{bmatrix} \begin{bmatrix} \tilde{\tilde{u}}_1 \\ \tilde{\tilde{u}}_2 \end{bmatrix} \tag{9}$$

Obtaining:

$$\begin{aligned} \dot{\tilde{x}}_4 &= \tilde{\tilde{u}}_1 - \frac{k_t}{m}\tilde{x}_4 \\ \dot{\tilde{x}}_5 &= \tilde{\tilde{u}}_2 - \frac{k_t}{m}\tilde{x}_5 \\ \dot{\tilde{x}}_6 &= \tilde{u}_3 - \frac{k_t}{m}\tilde{x}_6 - g \end{aligned}$$

Outer control loop

To linearize the nonlinear dynamics we manipulate the control inputs:

$$\begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{bmatrix} = \begin{bmatrix} S_\psi & -C_\psi \\ C_\psi & S_\psi \end{bmatrix} \begin{bmatrix} \tilde{\tilde{u}}_1 \\ \tilde{\tilde{u}}_2 \end{bmatrix} \tag{9}$$

Obtaining:

$$\begin{aligned} \dot{\tilde{x}}_4 &= \tilde{\tilde{u}}_1 - \frac{k_t}{m}\tilde{x}_4 \\ \dot{\tilde{x}}_5 &= \tilde{\tilde{u}}_2 - \frac{k_t}{m}\tilde{x}_5 \\ \dot{\tilde{x}}_6 &= \tilde{u}_3 - \frac{k_t}{m}\tilde{x}_6 - g \end{aligned}$$

Linear and decoupled

Outer control loop

The control inputs are chosen as:

$$u_i = \ddot{x}_i + 2\xi_{out} c_{out} \dot{e}_i + c_{out}^2 e_i \tag{10}$$

where:

- $\ddot{x}_i$ is the feed-forward term;
- $e_i = x_{di} - x_i$ is the error;
- c_{out} represents the natural frequency of the system and must be small: we want the outer loop control to be slower then inner control loop;
- ξ_{out} represents the dumping of the system.

Inner control loop

Inner control loop

Task of Inner control loop:

to control the dynamics of the attitude angle and the yaw velocity.

Equation relative to ψ, θ, p, q, r expressed such that the origin of the system is an equilibrium point when the inputs are equal to zero.

State vector:

$$\hat{x} = \left[\begin{array}{ccccc} \hat{x}_1 & \hat{x}_2 & \hat{x}_3 & \hat{x}_4 & \hat{x}_5 \end{array} \right] = \left[\begin{array}{ccccc} x_1 & x_2 & x_4 & x_5 & (x_6 - \frac{mgd}{k_r}) \end{array} \right] \quad (11)$$

Inputs vector:

$$\hat{u} = \left[\begin{array}{ccc} \hat{u}_1 & \hat{u}_2 & \hat{u}_3 \end{array} \right] = \left[\begin{array}{ccc} (u_1 - mg) & u_2 & (u_3 - mgd) \end{array} \right] \quad (12)$$

Nonlinear model

The system is described by:

$$\dot{\hat{x}} \;=\; \hat{f}\;(\hat{x}) \;+\; \hat{G}\;(\hat{x})\;\hat{u} \tag{13}$$

$$\hat{y} = [\;\hat{h}_1 \quad \hat{h}_2 \quad \hat{h}_3\;] = [\;\hat{x}_1 \quad \hat{x}_2 \quad \hat{x}_5\;] \tag{14}$$

where:

$$\hat{f}(\hat{x}) = \begin{bmatrix} \hat{x}_3 + \hat{x}_4 S_{\hat{x}_1} T_{\hat{x}_2} + (\hat{x}_5 + \frac{mgd}{k_r}) C_{\hat{x}_1} T_{\hat{x}_2} \\ \hat{x}_4 C_{\hat{x}_1} - (\hat{x}_5 + \frac{mgd}{k_r}) s_{\hat{x}_1} \\ \frac{1}{I_{xx}}[-k_r\hat{x}_3 - \hat{x}_4(\hat{x}_5 + \frac{mgd}{k_r})(I_{zz} - I_{yy})] \\ \frac{1}{I_{yy}}[-k_r\hat{x}_4 - \hat{x}_3(\hat{x}_5 + \frac{mgd}{k_r})(I_{xx} - I_{zz})] \\ \frac{1}{I_{zz}}(-k_r\hat{x}_5) \end{bmatrix} \quad \hat{G}(\hat{x}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{l}{2I_{xx}} & 0 & -\frac{l}{2I_{xx}} \\ 0 & \frac{1}{I_{xx}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix}$$

Feedback Linearization

The system fulfills the conditions for the feedback linearization:

- the sum of the relative degrees is equal to the dimension of the state vector;
- the decoupling matrix:

$$M(\hat{x}) = \begin{bmatrix} \frac{l}{2I_{xx}} & \frac{S_{\hat{x}_1} T_{\hat{x}_2}}{I_{xx}} & \frac{C_{\hat{x}_1} T_{\hat{x}_2}}{I_{zz}} - \frac{l}{2dI_{xx}} \\ 0 & \frac{C_{\hat{x}_1}}{I_{xx}} & -\frac{S_{\hat{x}_1}}{I_{zz}} \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \tag{15}$$

  is always invertible.

Feedback Linearization

The system fulfills the conditions for the feedback linearization:

- the sum of the relative degrees is equal to the dimension of the state vector;
- the decoupling matrix:

$$M(\hat{x}) = \begin{bmatrix} \frac{l}{2I_{xx}} & \frac{S_{\hat{x}_1} T_{\hat{x}_2}}{I_{xx}} & \frac{C_{\hat{x}_1} T_{\hat{x}_2}}{I_{zz}} - \frac{l}{2dI_{xx}} \\ 0 & \frac{C_{\hat{x}_1}}{I_{xx}} & -\frac{S_{\hat{x}_1}}{I_{zz}} \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \tag{15}$$

  is always invertible.

We can compute the linearizing input as:

$$u(x, v) = \alpha(x) + \beta(x) \cdot v \tag{16}$$

Feedback Linearization

The (16) leads to the canonical Brunowski form:

$$\dot{x}_c = A_c x_c + B_c v \tag{17}$$

$$A_c = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad B_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{18}$$

The linear input v is chosen to regulate the attitude angles and the yaw velocity to the desired values, coming from the *outer control loop*.

Simulations

Outer loop: Tracking

Trajectory tracking of cubic polinomial trajectory with $t \in [0, T]$

Initial values:

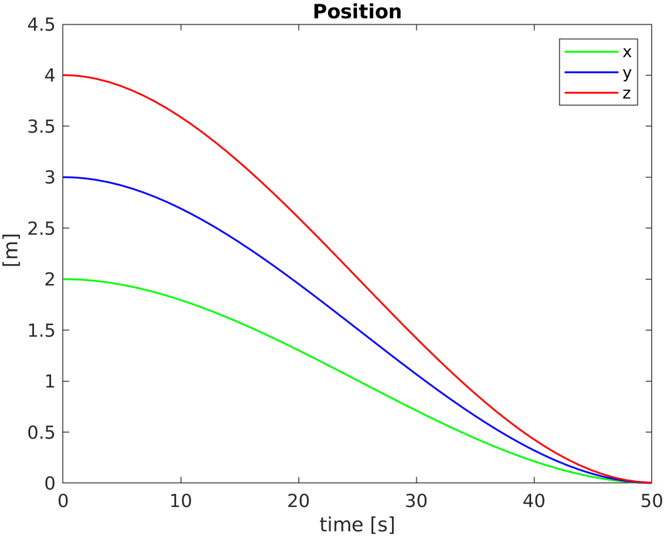
Variable	Value	Description
$\tilde{x}_{1,0}$	2 m	x
$\tilde{x}_{2,0}$	3 m	y
$\tilde{x}_{3,0}$	4 m	z
$\tilde{x}_{4,0}$	0 m/s	$\dot{x}$
$\tilde{x}_{5,0}$	0 m/s	$\dot{y}$
$\tilde{x}_{6,0}$	0 m/s	$\dot{z}$

Final values:

Variable	Value
$\tilde{x}_{1,T}$	0 m
$\tilde{x}_{2,T}$	0 m
$\tilde{x}_{3,T}$	0 m
$\tilde{x}_{4,T}$	0 m/s
$\tilde{x}_{5,T}$	0 m/s
$\tilde{x}_{6,T}$	0 m/s

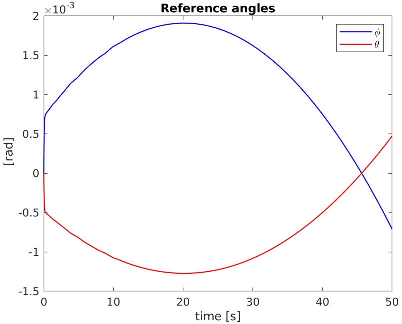
Outer loop: Tracking

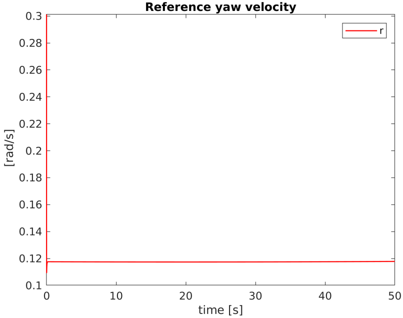
Trajectory tracking of cubic polinomial trajectory with $t \in [0, T]$



Output of the outer loop:

$$y_1 = \phi, \qquad y_2 = \theta, \qquad y_3 = r$$





Inner loop: regulation

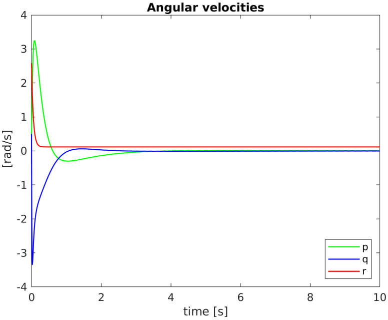
Initial values:

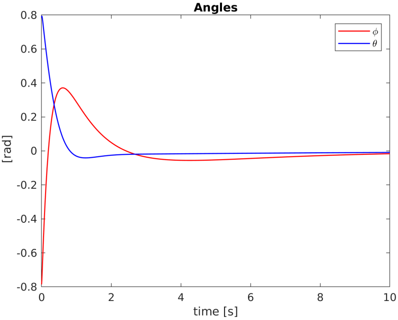
Variable	Value	Description
x_1	$-\pi/4$ rad	ϕ
x_2	$\pi/4$ rad	θ
x_3	$\pi/4$ rad	ψ
x_4	-0.5 rad/s	p
x_5	0.5 rad/s	q
x_6	2.7 rad/s	r

Final values:

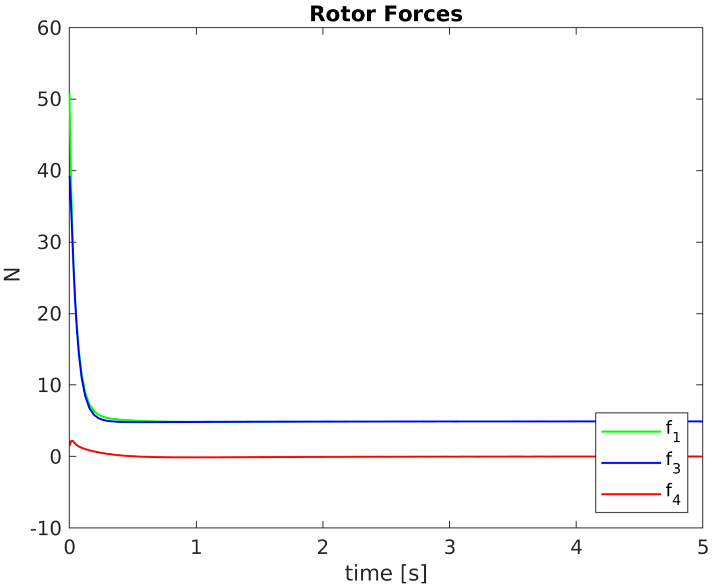
Variable	Value
x_1	0 rad
x_2	0 rad
x_3	0 rad
x_4	0 rad/s
x_5	0 rad/s
x_6	mgd/k_r rad/s

Inner loop: Regulation

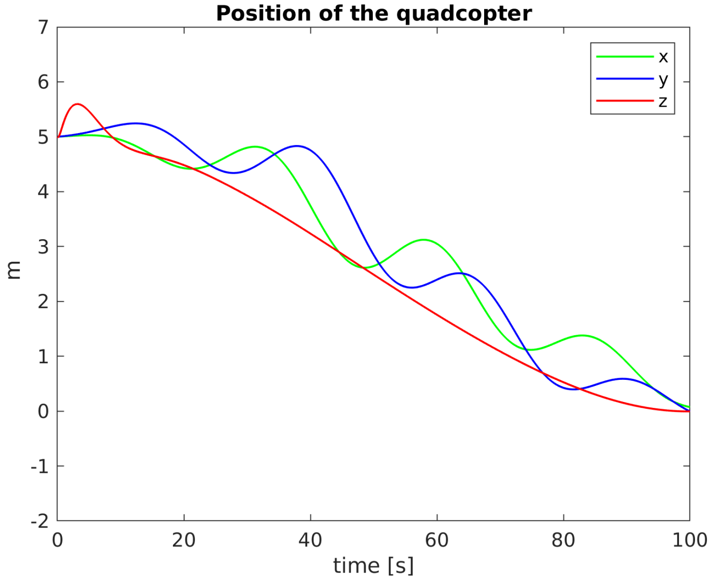




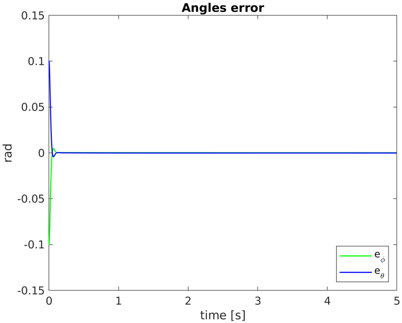
Inner loop: Regulation

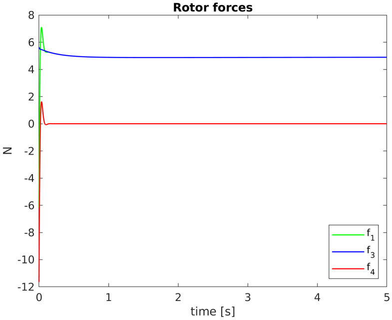


Complete loop

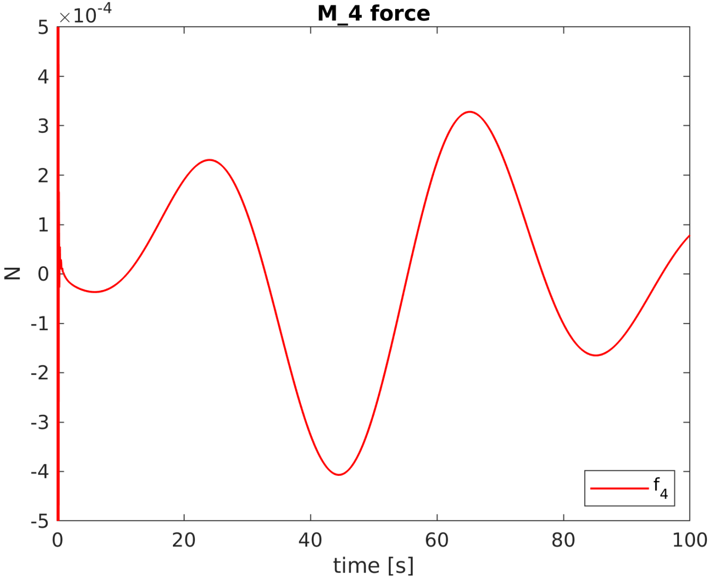


Complete loop





Complete loop



Conclusions

Conclusions

Problem statement:

- Control a quadcopter in case off loss off one of the actuators

Proposed solution:

- Adopted strategy: double loop architecture
- Sacrified state: yaw angle

Consequences:

- Horizontal positions oscillates around the desired trajectory, allowing only slow recovering manouvers

Issues:

- Motors force saturation, considered as uncertainty in the model