Quad Rotorcraft Switching Control: An Application for the Task of Path Following

Luis-Rodolfo Garcia Carrillo, Gerardo Ramon Flores Colunga, Guillaume Sanahuja, Rogelio Lozano

To cite this version:

Luis-Rodolfo Garcia Carrillo, Gerardo Ramon Flores Colunga, Guillaume Sanahuja, Rogelio Lozano. Quad Rotorcraft Switching Control: An Application for the Task of Path Following. IEEE Transactions on Control Systems Technology, 2014, 22 (4), pp.1255-1267. 10.1109/TCST.2013.2284790 . hal-00918169

HAL Id: hal-00918169
https://hal.science/hal-00918169

Submitted on 30 Dec 2013

Quad Rotorcraft Switching Control: An Application for the Task of Path Following

Luis Rodolfo García Carrillo, Gerardo R. Flores Colunga, Guillaume Sanahuja, and Rogelio Lozano

Abstract

This paper addresses the problem of road following using a quad rotorcraft equipped with an onboard image processing system. The main objective consists of estimating and tracking the road without a priori knowledge of the path to be tracked. Special interest is also given to the development of efficient estimation and control strategies for dealing with situations when the road is not detected in the camera image. Aiming at this goal, two operational regions are defined: one for the case when the road is detected, and another for the case when it is not. A switching between the measurements of imaging and inertial sensors enables estimation of the required vehicle parameters in both operational regions. Additionally, for dealing with both aforementioned cases, a switching control strategy for stabilizing the vehicle lateral position is proposed. The system stability is verified not only in the two operational regions, but also in the switching boundaries between them. The performance of the switching estimation and control strategies is tested in numerical simulations and real time experiments, successfully validating the effectiveness of the proposed approaches.

I. INTRODUCTION

This work is motivated by the theory of switching systems and the need to develop effective Unmanned Aerial Vehicle (UAV) controllers and state estimators, not only for attitude and

---

This work was partially supported by the Mexican National Council for Science and Technology (CONACYT) and the Institute for Science & Technology of Mexico City (ICyTDF).

L.R. García Carrillo is with the College of Science and Engineering, Texas A&M University - Corpus Christi, TX 78412-5797, USA, e-mail: (luis.garcia@tamucc.edu).

G.R. Flores Colunga and G. Sanahuja are with Heudiasyc UMR CNRS 7253 Laboratory, Université de Technologie de Compiègne, FRANCE, G.R. Flores Colunga (email: gfloresc@hds.utc.fr), G. Sanahuja (email: gsanahuj@hds.utc.fr).

R. Lozano is with Heudiasyc UMR CNRS 7253 Laboratory, UTC, FRANCE, and UMI-LAFMIA CNRS 3175, Cinvestav, MEXICO, (email: rlozano@hds.utc.fr).

position stabilization, but also for successfully executing a more complex predefined mission. Enabling a UAV to perform a task in a completely autonomous fashion is perhaps one of the most important and challenging control problems. Furthermore, estimating the vehicle position with sufficient accuracy is one of the important problems related with this subject.

The task addressed in this paper consists of performing an autonomous navigation mission. The objective is to enable a UAV to estimate and track a road with no prior knowledge of the path that must be followed. In addition, the mission must be successfully performed in the presence of external disturbances, which are present not only in the attitude angles, but also in the vehicle translational dynamics. To further improve the autonomy of the vehicle, the mission is complemented with an autonomous take-off, as well as with a landing at some specific point near the end of the road. With the purpose of simplifying the inherent complexity of the whole aforementioned task, the complete mission is divided into five modes of operation: *take-off*, ψ *alignment*, *lateral position* (y-state) *alignment*, *road tracking*, and *landing*.

The problem of stabilizing a quad rotorcraft using visual feedback has been addressed by several research groups, see for example [1], [2], [3] and the references therein. From previous experiences performed, it has been found that proportional derivative (PD) controllers work efficiently, in practice, for stabilizing the UAV attitude dynamics [4]. In addition, these kind of controllers have proved to be robust enough for applications having objectives similar to the one being addressed in the present research [5]. In both previously mentioned approaches, the imaging system was used for estimating the vehicle translational dynamics. However, the strategies proposed there are not robust for dealing with external disturbances which may cause the imaging system to temporarily lose the artificial landmarks used for obtaining the vehicle states.

Switching system ideas have been previously implemented to solve UAV-related problems. The implementation of different controllers with different gains is addressed in [6], this approach is adopted by the fact that the sole use of a PD controller in the position dynamics is not enough for attenuating the disturbances caused by, for example, wind gusts. In [7], the authors propose a switching strategy for estimating the states of a UAV equipped with imaging, inertial and air data sensing systems, capable of dealing with instants of time when the visual detection of the landmark fails. Following a similar reasoning, in [8] the authors present the analysis and control of a vertical take-off and landing (VTOL) aircraft at simulation level. By using a common Lyapunov function, the stability of the complete system, divided into three modes, is proved.

In this paper, we aim at estimating and controlling the vehicle relative position, orientation, and velocity with respect to (w.r.t.) a road that must be followed. An onboard camera allows estimation of the vehicle heading angle w.r.t. the longitudinal orientation of the road. Similarly, the imaging sensor is used for stabilizing the lateral distance of the vehicle in order to navigate exactly over the road. With the objective of developing a switching control strategy for estimation and tracking purposes, two operational regions are defined: one region for the situation when the road is in the camera field of view (FOV), and another region for when this is not the case. The system stability is shown not only in the two operational regions, but also in the switching boundaries between them. The performance of the switching control was tested in real time experiments, successfully validating the effectiveness of the proposed approach.

The paper is organized as follows. Section II presents the problem statement. The methods proposed to estimate the vehicle states are described in Section III. The switching control strategy and a discussion of the system stability are presented in Section IV. Section V presents the quad rotorcraft experimental platform. Numerical simulations, as well as the performance of the UAV during real-time experiments are shown in Section VI. Finally, Section VII presents the conclusions.

II. PROBLEM STATEMENT

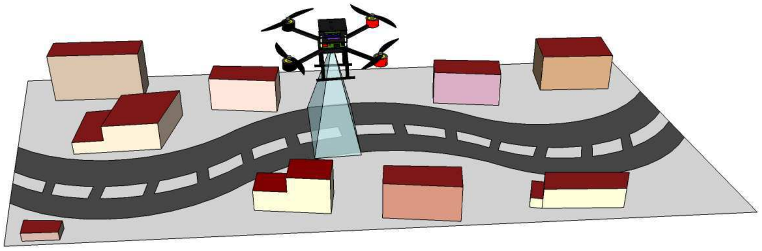


Fig. 1. Road following setup: The quad rotorcraft UAV under consideration is equipped with inertial sensors, an ultrasonic range finder, and a calibrated camera. The objective is to enable the UAV to estimate and track the road, with no prior knowledge of the path that must be followed.

In order to perform the road-following task, the quad rotorcraft must be capable of measuring its angular dynamics, as well as its relative 3-dimensional position and heading w.r.t. the road. An image describing the scenario considered in the present research is shown in Figure 1. The quad rotorcraft UAV platform is equipped with inertial sensors, an ultrasonic range finder, and a calibrated camera. The inertial system provides the vehicle angular dynamics, allowing the development of a control strategy for attitude stabilization. The ultrasonic range finder points downwards, directly measuring the altitude of the aerial vehicle during flight. The camera is installed on-board pointing downwards, in a manner that the plane formed by the helicopter (B_x, B_y) plane and the camera (C_x, C_y) image plane are parallel and have the same orientation, see Figure 2. The camera and the aerial vehicle move together as a rigid body, therefore, by using the image provided by the camera, vision-based strategies could be implemented for estimating the vehicle states required during the navigation mission.

The present study addresses the problem of stabilizing the quad rotorcraft UAV in 6 degrees of freedom (DOF) during a road-following task. Such a procedure can be detailed as follows. In the first stage, the vehicle performs an autonomous take-off, reaching the desired altitude z_{d} over the road. At this point, the heading of the vehicle, expressed by ψ, is driven to yield a parallel positioning between the helicopter x-axis (represented by B_x) and the longitudinal direction of the road (expressed by R_x). The helicopter forward speed, expressed by $\dot{x}$, is kept to a constant value while the distance between B_x and R_x, expressed by ${}^{B}e_y$ (see Figure 2), is regulated and kept to a minimum value, achieving a flight path well aligned and centered w.r.t. the road.

The objective of this paper is to design a road following strategy based on computer vision and switching controllers, with the purpose of stabilizing the vehicle altitude, heading angle, and lateral position (z, ψ, and y states, respectively) w.r.t. the road, while traveling at constant forward speed $(\dot{x} = c)$. Overcoming external disturbances while performing the navigation task is also a subject of interest. In general, the traveling speed may be time-varying. However, for simplicity we will only consider here the case of constant speed.

III. STATES ESTIMATION USING A VISION SENSOR

Suppose the camera-equipped quad rotorcraft is flying over a road composed by straight sections and smooth curves. If the distance between the road and aerial vehicle (altitude state) is appropriate, the road will be projected in the camera image plane as a group of straight lines,

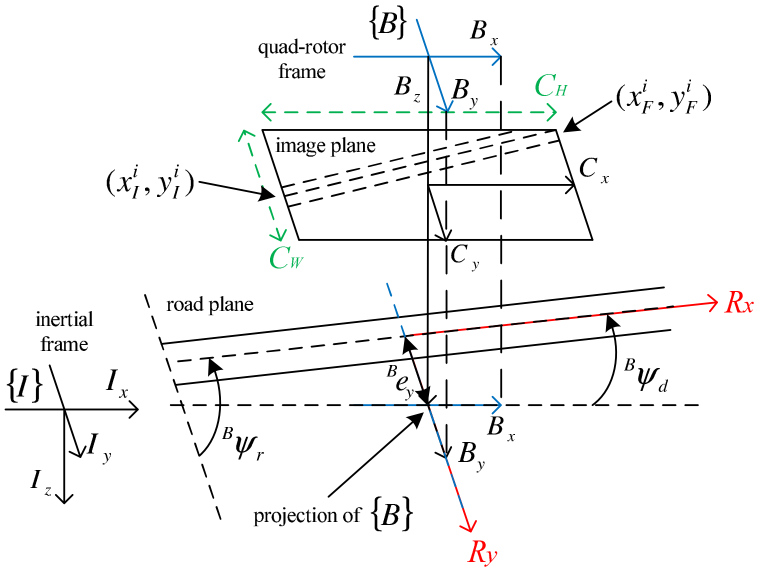


Fig. 2. Scheme of the road following mission: while flying at an appropriate altitude, the road is projected onto the image plane as a group of straight lines.

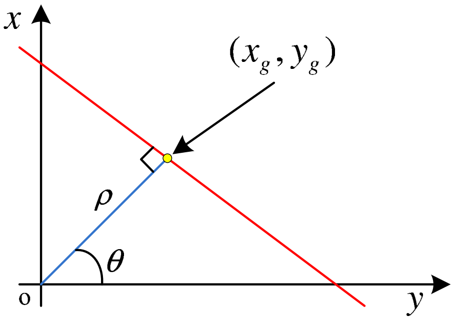


Fig. 3. Scheme of the Hough transform method for line detection.

see Figure 2. A straight line in the image plane can be seen as a segment of infinite length, whose center of gravity belongs to the straight line [9]. A scheme of such a representation can be seen in Figure 3. By implementing the Hough transform method for line detection, a straight line can be represented as [10]:

$$\rho \quad = \quad x\cos\theta + y\sin\theta \tag{1}$$

The center of gravity $(x_{\mathrm{g}}, y_{\mathrm{g}})$ of each straight line detected can be computed as

$$x_{\mathrm{g}}^i = \cos(\theta)\rho \qquad ; \qquad y_{\mathrm{g}}^i = \sin(\theta)\rho \tag{2}$$

where the super-index i stands for line i. Notice that it is possible to assign initial and final bounds to the line. Let us define $(x_{\rm I}^i, y_{\rm I}^i)$ as the initial point of the line, located in a place below the image lower margin. Similarly, let $(x_{\rm F}^i, y_{\rm F}^i)$ be the final point of the line, located in a place above the image upper margin. If the line has a parameter $\rho > 0$, the coordinates $(x_{\rm I}^i, y_{\rm I}^i)$ and $(x_{\rm F}^i, y_{\rm F}^i)$ will be defined as

$$x_{\rm I}^i = x_{\rm g} + \Gamma(-\sin(\theta)) \quad ; \quad y_{\rm I}^i = y_{\rm g} + \Gamma\cos(\theta) \tag{3}$$

$$x_{\rm F}^i = x_{\rm g} - \Gamma(-\sin(\theta)) \quad ; \quad y_{\rm F}^i = y_{\rm g} - \Gamma\cos(\theta) \tag{4}$$

where Γ is a known constant defining the bounds of the line. In the case where $\rho < 0$, the point $(x_{\rm F}^i, y_{\rm F}^i)$ will be computed as in equation (3), while the point $(x_{\rm I}^i, y_{\rm I}^i)$ will be computed as in equation (4). The set of lines obtained from the projection of the road on the image are grouped together with the purpose of obtaining just an average line. This average line will uniquely represent the road in the image with a single pair of initial and final coordinates

$$x_{\rm I} = \frac{1}{n}\sum_{i=1}^{n} x_{\rm I}^i \quad ; \quad y_{\rm I} = \frac{1}{n}\sum_{i=1}^{n} y_{\rm I}^i \tag{5}$$

$$x_{\rm F} = \frac{1}{n}\sum_{i=1}^{n} x_{\rm F}^i \quad ; \quad y_{\rm F} = \frac{1}{n}\sum_{i=1}^{n} y_{\rm F}^i \tag{6}$$

where n is the number of lines grouped together, $(x_{\rm I}, y_{\rm I})$ represents the initial (lowermost) road coordinate, and $(x_{\rm F}, y_{\rm F})$ represents the final (uppermost) road coordinate.

Wrong line detections may occur due to unexpected situations (e.g., changes in the scene illumination), therefore, a method for identifying false detections is required. In the developed algorithm, every line i detected must satisfy a safety condition in order to be considered as a *member* of the road. To achieve this, the final and initial coordinates of line i are compared with respect to the average initial and final coordinates computed in equation (5) and (6). If the coordinates of line i are far from the average values, line i is considered as a false detection and it is removed from further computations.

A. Computing the heading angle

The angle between the camera C_y axis and the line going from $(x_{\rm I}, y_{\rm I})$ to $(x_{\rm F}, y_{\rm F})$ can be computed using the coordinates $(x_{\rm I}, y_{\rm I})$ and $(x_{\rm F}, y_{\rm F})$ in the two argument arctangent function as follows

$$ {}^{B}\psi_{\mathrm{r}} \;=\; \text{arctangent}(y_{\mathrm{F}} - y_{\mathrm{I}}, x_{\mathrm{F}} - x_{\mathrm{I}}) \tag{7} $$

The angle ${}^{B}\psi_{\mathrm{r}}$ is used for obtaining the desired heading angle ${}^{B}\psi_{\mathrm{d}}$ which will align the vehicle x-axis (B_x) with the road longitudinal axis (R_x), see Figure 2. The desired heading angle ${}^{B}\psi_{\mathrm{d}}$ can be expressed as

$$ {}^{B}\psi_{\mathrm{d}} ={}^{B} \psi_{\mathrm{r}} - \frac{\pi}{2} \tag{8} $$

In equation (8) the term $\frac{\pi}{2}$ is subtracted because the angle ${}^{B}\psi_{\mathrm{r}}$ calculated using equation (7) has its origin in the position corresponding to 3:00 hours on the clock. If $\frac{\pi}{2}$ is not subtracted, the heading measurement obtained when the heading angle of the vehicle is aligned with the direction of the road would be equal to $+90^{\circ}$. With this subtraction we adjust such measurement to a value of 0°.

B. Computing the relative lateral position

Consider an image-based distance ${}^{C}e_y$ located between the road center of gravity projection $(x_{\mathrm{g}}, y_{\mathrm{g}})$ and the vehicle center of gravity projection $(x_0, y_0) = (C_{\mathrm{W}}/2, C_{\mathrm{H}}/2)$, where C_{W} and C_{H} represent the image width and height respectively, in pixel coordinates (see Figure 2). For the case, when $x_{\mathrm{I}} > x_F$, one has

$$ {}^{C}e_y \;=\; \left(\frac{x_{\mathrm{I}} - x_{\mathrm{F}}}{2} + x_{\mathrm{F}}\right) - \frac{C_{\mathrm{W}}}{2} \tag{9} $$

On the other hand, for the case when $x_{\mathrm{I}} < x_{\mathrm{F}}$, x_{I} must be replaced by x_{F} and vice-versa. The lateral position of the aerial vehicle w.r.t. the road can be estimated from ${}^{C}e_y$ as

$$ e_y \;=\; z\frac{{}^{C}e_y}{\alpha_y} \tag{10} $$

where z represents the altitude of the rotorcraft w.r.t. the road, and α_y represents the camera focal length, in terms of pixel dimensions, in the direction of C_y.

C. Translational velocities

The vehicle translational velocities in the forward and lateral directions can be estimated by implementing an optical flow algorithm in the camera image. In our work, the algorithm of Shi and Tomasi [11] is used to identify the mesh of points allowing an efficient feature tracking, i.e., the best textured points. After this step, the selected features are used by the pyramidal implementation of the Lucas-Kanade algorithm [12] to estimate the optical flow.

Consider the camera-vehicle arrangement moving in a 3-dimensional space w.r.t. a rigid scene. The optical flow computed at an image point (x_i, y_i) comprises translational and rotational parts expressed as

$$\begin{bmatrix} OF_{xi} \\ OF_{yi} \end{bmatrix} = T^i_{OF} + R^i_{OF} \tag{11}$$

where the translational part is

$$T^i_{OF} = \frac{1}{z} \begin{bmatrix} -\alpha_x & 0 & x_i \\ 0 & -\alpha_y & y_i \end{bmatrix} \begin{bmatrix} \dot{x}^c \\ \dot{y}^c \\ \dot{z}^c \end{bmatrix} \tag{12}$$

and the rotational part is

$$R^i_{OF} = \begin{bmatrix} \frac{x_i y_i}{\alpha_x} & -(\alpha_x + \frac{(x_i)^2}{\alpha_x}) & y_i \\ (\alpha_y + \frac{(y_i)^2}{\alpha_y}) & -\frac{x_i y_i}{\alpha_y} & -x_i \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{13}$$

The terms OF_{xi} and OF_{yi} are the optical flow components in the x and y coordinates, respectively, of the (x_i, y_i) feature. The terms $(\dot{x}^c, \dot{y}^c, \dot{z}^c)$ and (w_x, w_y, w_z) represent the camera translational velocities and rotation rates, respectively, while α_x and α_y express the camera focal lengths.

During the optical flow computation process the (x_i, y_i) features share the same movement (rigid scene assumption). By using all the tracked features, the mean values for the optical flow in the forward an lateral direction can be expressed as

$$\overline{OF}_x = \overline{V}_{OF_{\mathrm{x}}} + K_x \overline{V}_{OF_{\mathrm{z}}} + \overline{R}_{OF_x} \tag{14}$$

$$\overline{OF}_y = \overline{V}_{OF_{\mathrm{y}}} + K_y \overline{V}_{OF_{\mathrm{z}}} + \overline{R}_{OF_y} \tag{15}$$

where $\overline{OF}_x$ and $\overline{OF}_y$ are the mean values of the optical flow (sensed in the image coordinate system), $\overline{V}_{OF_{\mathrm{z}}}$ represents the relative depth, and K_x and K_y are known scale factors depending on the intrinsic parameters of the camera. Applying a similar approach to the one presented in [5], the rotational optical flow terms $\overline{R}_{OF_x}$ and $\overline{R}_{OF_y}$ are compensated and the pseudo-speeds $(\overline{V}_{OF_x}, \overline{V}_{OF_y}, \overline{V}_{OF_z})$ are deduced. This procedure yields

$$-z\frac{\overline{V}_{OF_x}}{\alpha_x} = \dot{x} \tag{16}$$

$$-z\frac{\overline{V}_{OF_y}}{\alpha_y} = \dot{y} \tag{17}$$

$$z\overline{V}_{OF_z} = \dot{z} \tag{18}$$

where the fact that the camera and the vehicle share the same movement has been used, i.e., the velocity of the camera is equal to the velocity of the vehicle, or $(\dot{x}^c, \dot{y}^c, \dot{z}^c) = (\dot{x}, \dot{y}, \dot{z})$.

D. Computing the heading and the lateral position when the road is not detected

The computation of the desired heading angle and lateral position of the vehicle are compromised by the sensitivity of the Hough transform method for line detection. In addition to this, it is not possible to estimate such parameters when the vehicle is flying over a region where the road is out of the camera FOV. Then, a realistic scenario suggests that the image of the road can be temporarily lost by the onboard camera. For this reason, it is crucial to extend the results presented in Sections III-A and III-B with the main objective of dealing with time events where the road is out of sight. Aiming at this goal, let us define a binary signal $s : [0, \infty) \to \{0, 1\}$ as

$$s(t) := \begin{cases} 0 & \text{no line detection at time } t \\ 1 & \text{camera detects line at time } t \end{cases} \tag{19}$$

In equation (19), $s(t)$ allows switching between two different methods for computing the vehicle states. Furthermore, for a given binary signal s and $t > \tau > 0$, denote by $T_s(\tau, t)$ the amount of time in the interval (τ, t) for wich $s = 0$. Formally, $T_s(\tau, t) := \int_\tau^t (1 - s(l))dl$.

Let us consider first the desired heading angle ψ_{d}. Using equation (19) it is possible to define

$$\psi_{\mathrm{s}}(t) = s(t)\psi_{\mathrm{d}}(t) + (1 - s(t))\psi_{\mathrm{IMU}}(T_{\mathrm{s}}(\tau)) \tag{20}$$

where $\psi_{\mathrm{d}}(t)$ is obtained from equation (8), $\psi_{\mathrm{IMU}}(T_{\mathrm{s}}(\tau))$ is the heading angle measured by the IMU at the time when the binary signal s changes from 0 to 1, and $\psi_{\mathrm{s}}(t)$ represents the desired heading angle that will be used in the helicopter control algorithm. Given the lack of visual information when $s = 0$, we decided to keep the yaw angle at a fixed value as long as the road is not seen. Equation (20) allows switching between: (i) a desired heading provided by the imaging algorithm, and (ii) a desired heading temporarily determined by the IMU heading measured at the time when the binary signal s changes from 1 to 0.

Consider now the computation of the lateral position of the vehicle when the road is out of sight. Equation (19) allows us to compute

$$e_{y\mathrm{s}}(t) \;=\; s(t)e_y(t) + (1 - s(t))\left(e_y(T_\mathrm{s}(\tau)) + \int_\tau^t \dot{y}(t)dt\right) \tag{21}$$

where $e_y(t)$ is obtained as in equation (10), $e_y(T_\mathrm{s}(\tau))$ is the lateral position measurement at the time when the binary signal s changes from 1 to 0, and $\dot{y}(t)$ represents the vehicle lateral velocity obtained from equation (17). Notice that equation (21) allows switching between two different approaches for estimating the vehicle lateral position w.r.t. the road. In the first approach (when $s = 1$) the lateral position is directly obtained from the detection of the road in the camera image (equation (10)). The second approach (when $s = 0$) makes use of the lateral position measurement at the time when the binary signal s changes from 1 to 0, in combination with the integral of the translational velocity $\dot{y}$ during the time the road is not being detected by the vision algorithm, i.e., the amount of time in the interval (τ, t) that $s = 0$.

IV. CONTROL STRATEGY

A normalized mathematical model for the quad-rotor is implemented aiming at developing the navigation control strategy [13]:

$$\begin{aligned}
\ddot{x} &= -u_1(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)\\
\ddot{y} &= -u_1(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)\\
\ddot{z} &= 1 - u_1(\cos\phi\cos\theta)\\
\ddot{\theta} &= u_2\\
\ddot{\phi} &= u_3\\
\ddot{\psi} &= u_4
\end{aligned} \tag{22}$$

The quad-rotor model in equation (22) can be written in a state-space form by introducing $\dot{X} = (x_1, ..., x_6)^T \in \mathbb{R}^6$, $\dot{Z} = (z_1, ..., z_6)^T \in \mathbb{R}^6$, with states defined by

$$\begin{array}{l|l} x_1 = x & z_1 = \theta \\ x_2 = \dot{x} & z_2 = \dot{\theta} \\ x_3 = y & z_3 = \phi \\ x_4 = \dot{y} & z_4 = \dot{\phi} \\ x_5 = z & z_5 = \psi \\ x_6 = \dot{z} & z_6 = \dot{\psi} \end{array} \tag{23}$$

Using the non-linear model in equation (22) and the notation from equation (23), the following approximation is obtained

$$\begin{array}{rcl|rcl} \dot{x_1} & = & x_2 & \dot{z_1} & = & z_2 \\ \dot{x_2} & = & -z_1 u_1 & \dot{z_2} & = & u_2 \\ \dot{x_3} & = & x_4 & \dot{z_3} & = & z_4 \\ \dot{x_4} & = & z_3 u_1 & \dot{z_4} & = & u_3 \\ \dot{x_5} & = & x_6 & \dot{z_5} & = & z_6 \\ \dot{x_6} & = & 1 - u_1 & \dot{z_6} & = & u_4 \end{array} \tag{24}$$

In general, the vehicle never operates in areas where $|\theta| \geq \pi/2$ and $|\phi| \geq \pi/2$. For this reason, the approximation in (24) is chosen for implementation instead of its non-linear version. This kind of operation is satisfied even in research works where the non-linear model is used together with a feedback control law [14].

A. *Operating modes for the road following mission*

We divide the road following mission into four different stages which are described next:

1) Take-off mode: The objective is to achieve the desired altitude z_{d}, while avoiding drift in the x-y plane.

2) ψ-alignment mode (ψ AL): In this mode the vehicle has already achieved the desired altitude. The task to be accomplished here is to align the vehicle heading angle w.r.t. the road direction.

3) y-alignment mode (y AL): The vehicle is required to maintain a flying path well aligned w.r.t. the center of the road.

4) Tracking mode: In this mode, the forward displacement of the vehicle is stabilized in order to navigate at constant velocity.

5) Landing mode: When the vehicle has reached a previously defined position near to the end of the road, the altitude is controlled in order to perform an autonomous landing. In the current implementation of the algorithm, the end of the road is not detected autonomously, i.e., a signal for activating the "landing" mode is manually generated.

B. Control Laws for each Operating Mode

The control strategy proposed in all the operating modes is based on the idea that the global system presented in equation (24) comprises two subsystems; the attitude dynamics and the position dynamics, with a time-scale separation between them [15]. From this approach it is possible to propose a hierarchical control scheme where the positioning controller provides the reference attitude angles (θ_{d}, ϕ_{d} and ψ_{d}), which are the angles that must be tracked by the orientation controllers. For the complete system analysis, the error dynamics of the model in equation (24) are represented by the error terms $\tilde{x}_i = x_i - x_{i_{\mathrm{d}}}$ and $\tilde{z}_i = z_i - z_{i_{\mathrm{d}}}$, with $i \in \{1, ..., 6\}$.

1) Attitude Control: For the present study, the attitude dynamics controller is the same in all modes of operation An integral sliding mode control (ISMC) is proposed and implemented on the platform. This kind of controller was chosen due to its insensitivity to model errors, parametric uncertainties, and diverse disturbances [16]. Specifically, for the pitch dynamics case the error equation is defined as $\tilde{z}_1 = z_1 - z_{1_{\mathrm{d}}}$. As shown in [17], let us select the following switching function

$$s(z,t) = \dot{\tilde{z}}_1 + 2\lambda\tilde{z}_1 + \lambda^2 \int_0^t \tilde{z}_1(\tau)\,\mathrm{d}\tau \tag{25}$$

which depends on the pitch dynamics states. The parameter λ in equation (25) is the slope of the sliding curve, which should be greater than zero to ensure the asymptotic stability of the sliding mode. Computing the time derivative of equation (25) one has

$$\dot{s} = u_2 + 2\lambda z_2 + \lambda^2 \tilde{z}_1 \tag{26}$$

Considering the sliding mode condition $\dot{s} = 0$ and using equation (26), one finds the equivalent control

$$u_{2_{eq}} = -2\lambda z_2 - \lambda^2 \tilde{z}_1 \tag{27}$$

In order to obtain a control law such that the state vector $\tilde{z}_1$ remains on the sliding surface $s(z,t) = 0$, $\forall\ t > 0$, a Lyapunov function candidate is defined as

$$v(s) = \frac{1}{2}s^2 \tag{28}$$

A sufficient condition for the stability of the pitch sub-system can be satisfied if one can ensure that the following condition holds

$$\dot{v}(s) = \frac{1}{2}\frac{d}{dt}s^2 \leq -\eta|s|, \qquad \eta \geq 0 \tag{29}$$

Then, the system remains on the sliding surface and the states converge to the origin. From this, $s\dot{s} \leq -\eta|s|$ and the controller must be chosen such that

$$u_2 = u_{2_{eq}} - K\text{sign}(s) \tag{30}$$

where K is a positive real number. Following a similar approach, it is possible to obtain the controllers for the yaw and roll angle dynamics.

2) Position Control: As previously explained, the position control has well defined objectives for each operating mode. Even though the control objectives are different, the same control law structure is applied for all the operating modes.

Motion in the $x - y$ plane is accomplished by orientating the vehicle thrust vector in the direction of the desired displacement. As a consequence, the angles θ_d and ϕ_d act as virtual controllers for the position dynamics. The control laws proposed for the z and x dynamics, respectively, are expressed by the following equations

$$u_1 = k_{pz}(x_5 - x_{5_\text{d}}) + k_{vz}(x_6 - x_{6_\text{d}}) - 1 \tag{31}$$

$$\theta_d = \frac{k_{px}(x_1 - x_{1_\text{d}}) + k_{vx}(x_2 - x_{2_\text{d}})}{u_1} \tag{32}$$

where k_{vx}, k_{px}, k_{pz} and k_{vz} are positive real numbers. Notice that, in the present application, we are not making use of the term x_{1_d}, which represents a desired x position. As previously explained, we are only interested on regulating the forward velocity, therefore, equation (32) can be reduced to

$$\theta_d = \frac{k_{vx}(x_2 - x_{2_\text{d}})}{u_1} \tag{33}$$

In order to avoid singularities, the initial error between the quad-rotorcraft altitude state and its corresponding altitude reference should be small. It is worth mentioning that, when the altitude error converges to zero, the value of u_1 approaches the value -1. This means that the error converges only locally, avoiding the singularity.

The lateral position stabilization makes use of two PID controllers, one for the case when the road is being detected and the other for the case when it is not. Although both PID controllers are similar, they are designed having different parameters. When the road is inside the FOV, the gains of the control algorithm are tuned to behave as a PD controller since the vehicle needs to regulate its y coordinate at a minimum value ($y \approx 0$) as fast as possible [6]. On the other hand, when the camera loses sight of the road, a switching strategy allows using a different method for measuring the vehicle ψ angle and y position. Additionally, the parameters of the positioning controller switch to alternative values emulating a PI controller. In both cases, the control objective is to regulate the x_3 state to the origin, i.e. $x_{3\mathrm{d}} = 0$.

- Control scheme implemented when the road is detected: the feedback control law is

  $$z_3 = \frac{1}{u_1}(-k_{L3}x_3 - k_{L4}x_4 - k_{LI}\xi) \tag{34}$$

  where k_{L3}, k_{L4} and k_{LI} are positive real numbers. Here, the additional state ξ has been introduced, whose dynamics are given by $\dot{\xi} = x_{3\mathrm{d}} - x_3 = -x_3$. Using the control in equation (34), the closed-loop system of the lateral dynamics is

  $$\begin{aligned} \dot{x}_3 &= x_4 \\ \dot{x}_4 &= -k_{L3}x_3 - k_{L4}x_4 - k_{LI}\xi \\ \dot{\xi} &= -x_3 \end{aligned} \tag{35}$$

  Equation (35) can be represented as $\dot{e}_y = A_L e_y$ where $e_y = (x_3, x_4, \xi)^{\mathrm{T}}$ and

  $$A_L = \begin{pmatrix} 0 & 1 & 0 \\ -k_{L3} & -k_{L4} & k_{LI} \\ -1 & 0 & 0 \end{pmatrix} \tag{36}$$

- Control scheme implemented when the road is not detected: the control scheme proposed is the same as given by equation (34), with the only difference being a set of different gains. The closed-loop system is represented as $\dot{e}_y = A_{NL}e_y$, where

  $$A_{NL} = \begin{pmatrix} 0 & 1 & 0 \\ -k_{NL3} & -k_{NL4} & k_{NLI} \\ -1 & 0 & 0 \end{pmatrix} \tag{37}$$

and k_{NL3}, k_{NL4} and k_{NLI} are positive real numbers.

Remark 1: The controller parameters of both operational regions were adjusted independently following an heuristic approach consisting of two main steps which are described next. During the first step, special care is taken to stabilize the vehicle fast dynamics, which corresponds to the angular behavior. The parameters associated with the angular rate are adjusted first, until an appropriate angular rate response is obtained. Following a similar method, the next parameter to be adjusted corresponds to the attitude control parameter. After both parameters have been adjusted properly, an autonomous hover flight experiment is performed to verify the effectiveness of the tuning procedure. The second part of the adjustment procedure is devoted to stabilizing the vehicle translational dynamics, which involves the implementation of visual feedback in the control strategy. First, the parameter corresponding to the translational velocity is adjusted until the vehicle translational drift is almost eliminated. Next, the parameter associated with the translational position is tuned to obtain an appropriate behavior of the controller for stabilizing the 3-dimensional position of the vehicle over road area. The main objective behind the procedure just described is to minimize the tracking error, in order to obtain the most appropriate angular and translational behavior. For the first case (road detected), the gains were selected as $k_{L3} = 1000$, $k_{L4} = 850$ and $k_{LI} = 0.1$, while in the second case (no road detected) as $k_{NL3} = 1000$, $k_{NL4} = 1$ and $k_{NLI} = 9.5$. In this way, the system presents a switched-system characteristic.

C. Stability Analysis of the Lateral Position Control

We now present a study concerning the stability of the system across switching boundaries, i.e., in the regions where the control parameters of equation (34) switch to different values. In fact, it is possible to find a common Lyapunov function for the closed-loop system of the two controllers proposed for the lateral dynamics [18]. Following such approach, similar pole locations have to be chosen for both cases, i.e., when the road is detected and when it is not. However, this is not the present situation since different gain values are being applied.

Let d_c be defined as the distance measured from the vehicle center of gravity projection (in the road plane) to the point where the camera loses the image of the road (see Figure 4). From this, a state-dependent switched linear system can be defined, which will be given by the closed-loop

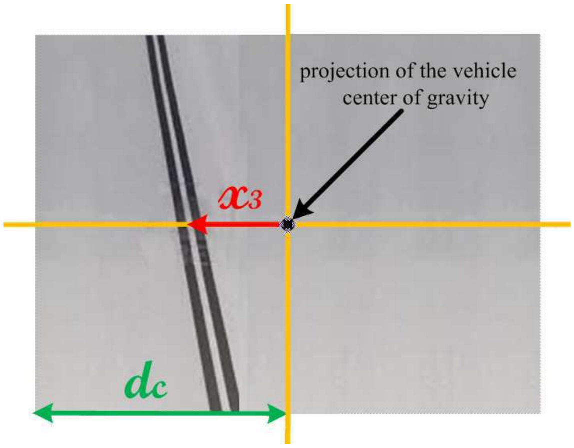


Fig. 4. The distance measured from the vehicle center of gravity projection in the image plane to the point where the camera loses the image of the road is defined as d_{c}. The distance between the vehicle center of gravity projection and the road is defined as x_3.

system (36)-(37), together with the following switching conditions

$$\dot{e}_y = \begin{cases} A_{NL}e_y & \text{if } x_3 \leq -d_{\mathrm{c}} \quad \text{or } s = 0 \\ A_L e_y & \text{if } -d_{\mathrm{c}} < x_3 < d_{\mathrm{c}} \quad \text{and } s = 1 \\ A_{NL}e_y & \text{if } x_3 \geq d_{\mathrm{c}} \quad \text{or } s = 0 \end{cases} \tag{38}$$

A switching event can occur in system (38) every time the trajectory crosses a *switching surface*. For the case under consideration, the switching surfaces are encountered when the system changes from $\dot{e}_y = A_L e_y$ to $\dot{e}_y = A_{NL}e_y$, and viceversa. In general, there is no need to associate each subsystem in (38) with a global Lyapunov function. In fact, it is enough to require that each function V_i decrease along solutions of the i-th subsystem in the region where this system is active.

The stability of the overall system (38) is accomplished when: (i) individual subsystems are stable, and (ii) the overall system remains stable when a switching surface is crossed. Indeed, it is possible to verify that individual subsystems of the switching system (38) are globally exponentially stable. Furthermore, if the switching between individual subsystems is sufficiently slow, the entire system (38) will be globally exponentially stable [19]. Aiming at this goal, let us define a dwell time τ_d satisfying $t_{n+1} - t_n \geq \tau_d$, where t_n (for $n = 1, 2, 3, \ldots$) represents a switching time. Next, it is required to investigate the lower bound on τ_d from the exponential decay bounds of the subsystems as follows.

As previously mentioned, individual subsystems are globally exponentially stable, then there exist Lyapunov functions V_L, V_{NL} which, for some positive constants a_i, b_i, and c_i satisfy [20]

$$a_i\|e_y\|^2 \leq V_i(e_y) \leq b_i\|e_y\|^2 \tag{39}$$

and

$$\frac{\partial V_i}{\partial e_y}A_i e_y \leq -c_i\|e_y\|^2 \tag{40}$$

for $i = \{L, NL\}$. Combining (39) and (40) we have

$$\frac{\partial V_i}{\partial e_y}A_i e_y \leq -\frac{c_i}{b_i}V_i(e_y) \tag{41}$$

This implies that

$$V_i(e_y(t_0+\tau_d)) \leq e^{-\frac{c_i}{b_i}\tau_d}V_i(e_y(t_0)) \tag{42}$$

Let us consider two switching times $\{t_1, t_2\}$, and let us also assume that subsystem $\dot{e}_y = A_L e_y$ is active on the interval $t \in [t_0, t_1)$, while subsystem $\dot{e}_y = A_{NL}e_y$ is active on $t \in [t_1, t_2)$. From inequalities (39)-(42) it follows that

$$V_L(t_2) \leq \frac{b_L}{a_{NL}}V_{NL}(t_2) \leq \frac{b_L}{a_{NL}}e^{-\frac{c_{NL}}{b_{NL}}\tau_d}V_{NL}(t_1) \tag{43}$$

and also

$$V_L(t_2) \leq \frac{b_{NL}b_L}{a_L a_{NL}}e^{-\left(\frac{c_L}{b_L}+\frac{c_{NL}}{b_{NL}}\right)\tau_d}V_L(t_0) \tag{44}$$

In order to find the lower bound on τ_d it is sufficient to ensure that (see Theorem 1 of [21])

$$V_L(t_2) - V_L(t_0) \leq -p\|e_y(t_0)\|^2 \tag{45}$$

where $p > 0$. It is now straightforward to compute an explicit lower bound on τ_d, which guarantees that the switched system (38) is globally asymptotically stable [22]

$$\tau_d > \frac{b_L + b_{NL}}{c_L + c_{NL}} \log \frac{b_L b_{NL}}{a_L a_{NL}} \tag{46}$$

The lower bound on τ_d can be satisfied by adjusting a_i, b_i, and c_i, for $i = \{L, NL\}$, which depend on the controller gains and the Lyapunov function.

Fig. 5. UAV experimental platform: The quad-rotor equipped with imaging, inertial and altitude sensing systems.

V. EXPERIMENTAL PLATFORM

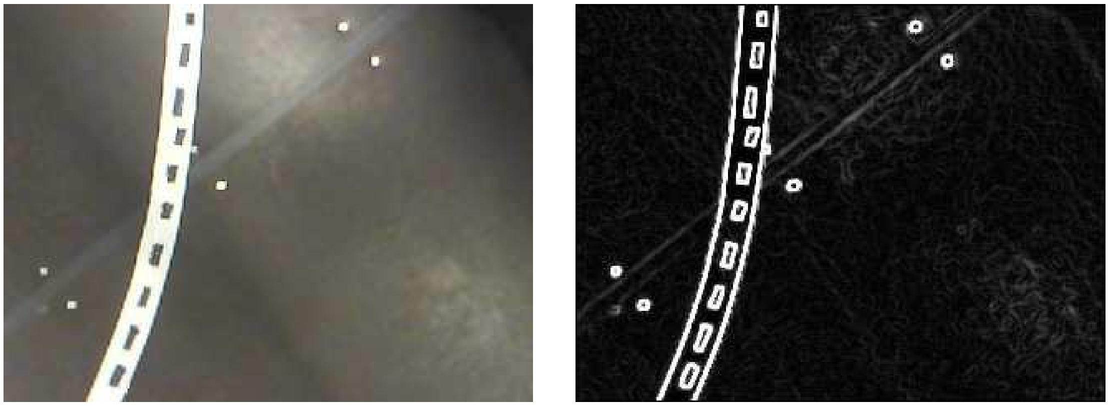
A. Quad-rotor experimental vehicle

The quad-rotor used during the real time application is shown in Figure 5. It has been built using a group of commercially available components. The body frame is an MK-40 from MikroKopter. The distance between same axis rotors is 40 cm. Motors are BL-Outrunner from Robbe ROXXY, which are driven by BlCtrl I2C electronic speed controllers. The weight of the rotorcraft is 1.1 kg. It has a 11.1V - 6000 mAh LiPo battery, allowing an autonomy of about 15 minutes. The onboard electronics are based on a IGEPv2 card, equipped with a Texas Instruments DM3730 System On Chip (SOC). The SOC benefits from having an ARM CortexA8 core running at 1 GHz, and a C64x+ DSP core running at 800 MHz. The ARM core allows execution of Linux, as well as its real-time extension Xenomai. The control law is executed in real-time at a 100 Hz frequency. The quad-rotor sensor suit consists of the next group of components. Inertial measurements are provided at 100 Hz by means of a 3DMGX3-25 IMU from Microstrain®. A SRF10 ultrasonic range finder provides the vehicle altitude at 50 Hz in a range between 0 m and 2 m. All the previously mentioned components are fully embedded onboard the vehicle.

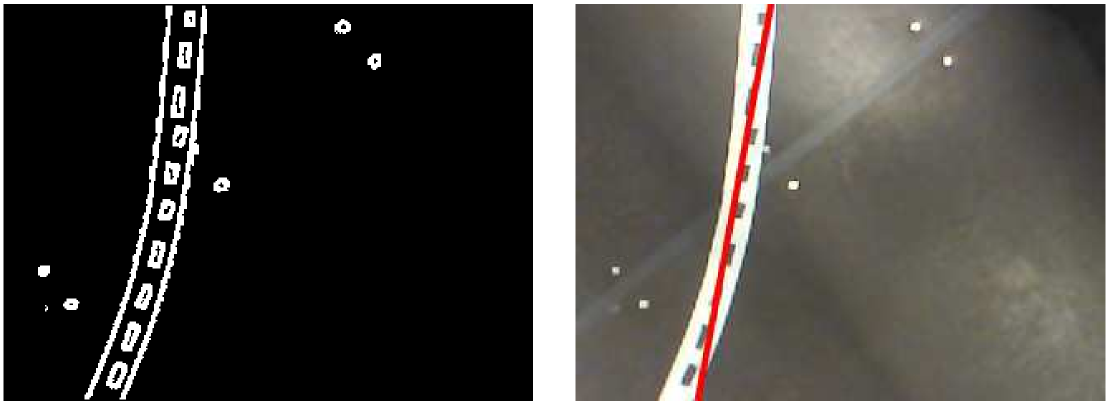
B. Embedded vision system

The vehicle is equipped with a PlayStation® Eye camera whose focal length is $(f_x, f_y) = (270.87, 271.76)$ pixels. The camera is capable of providing 120 images per second, at a resolution of 320×240 pixels. The camera points downwards which allows observing the scene below the vehicle. The images provided by the camera are processed by computer vision algorithms in order to estimate the helicopter translational velocity in the $x - y$ plane, the heading angle, as well as the lateral position w.r.t. the road direction. The translational velocity in the $x-y$ plane is obtained from an optical flow algorithm, which is based on the pyramidal Lucas-Kanade method. For this purpose, the algorithm uses two pyramid levels, searching for 64 characteristic features in the image. A Harris affine region detector was implemented to perform the characteristic features detection.

The computation of the heading angle and lateral position of the vehicle is based on a Hough transform technique for line detection. With the objective of executing the Hough transform computation in the on-board DSP, the Hough's accumulator must fit in the DSP cache, which is possible if the resolution of the ρ and θ parameters from equation (1) are properly selected. Based on the chosen resolution, a lookup table containing the different values of sines and cosines is generated and stored in the DSP cache. The image pixels are read in groups of 8 (1 octet) with the purpose of effectively filling the accumulator. It is worth mentioning that reading 1 octet or a group of 8 octets requires the same number of cycles for the DSP. Then, the computations of the Hough transform are optimized for a proper execution in the DSP, which is capable of performing several additions, subtractions, multiplications and scalar products at the same time. The embedded DSP allows computing the optical flow (translational velocities) and Hough transform (heading angle and lateral position) on-board at a 100 Hz frequency. A raw image obtained from the onboard camera while the vehicle is flying over the road is shown in Figure 6 (a). Gray scale and black and white images obtained after preprocessing steps are shown in Figure 6 (b) and Figure 6 (c), respectively. Specifically, the grayscale image is used by the optical flow algorithm, while the black and white image is used by the algorithm for line detection. Finally, the post-processed image where the road detected has been highlighted is shown in 6 (d).



(a) Raw Image. (b) Grayscale Image.



(c) Black and White Image. (d) Post Processed Image

Fig. 6. Image obtained from the onboard camera while the UAV is flying above the road: (a) the raw image provided by the camera; (b) gray scale image used for optical flow computation; (c) black and white images used in Hough transform algorithm; (d) the post-processed image where the detection of the road has been highlighted.

C. Ground station

The rotorcraft is wirelessly linked to a ground station PC, where a graphical user interface (GUI) based on the QT application allows monitoring and controlling the vehicle. The user can visualize, in real time, graphics representing the measurements from the on-board sensors, as well as graphics representing the control law computation. It is also possible to modify (online) the parameters of the control laws, as well as the implemented filters. The rotorcraft can also be manually controlled by a human pilot using a PlayStation®3 joystick, which is linked to the ground station by means of a BlueTooth™ link.

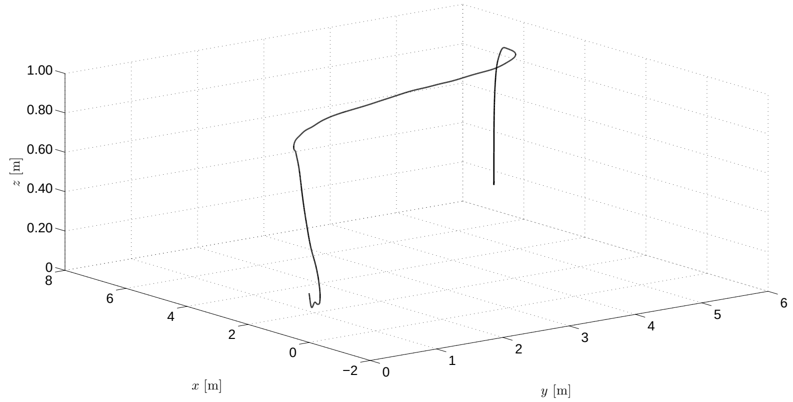


Fig. 7. Simulation of the road-following mission: The 3-dimensional trajectory flown by the vehicle. The black line represents the vehicle displacement.

VI. SIMULATIONS AND REAL TIME EXPERIMENTS

A. Numerical Simulations

The control strategy developed in Section IV has been designed for the quad rotorcraft model approximation in equation (24). In order to show that the obtained controller can also be used to control the non-linear plant in (22) as long as the pitch and roll angles are small, we have performed a set of numerical simulations. The simulation scenario follows the procedure previously explained in Subsection IV-A. A synthetic road having two smooth turns has been designed, and was used as the desired reference that the vehicle has to follow during navigation. The desired altitude is set at 0.70 m over the road. A ramp profile was used for modifying the desired velocity from $\dot{x} = 0.0$ m/s to $\dot{x} = 0.3$ m/s during the first stages of the navigation, and also for modifying the desired velocity from $\dot{x} = 0.3$ m/s to $\dot{x} = 0.0$ m/s before landing.

As can be seen in the set of set of Figures 7-11, the controller performs well in spite of the non-linearities of the plant. Figure 9 shows a disturbance in the y state reference signal (blue line) at $t = 18$ seconds. Notice how the proposed controller maintains stability of the system even in the presence of these unexpected behaviors. From Figure 11 it can be observed that the pitch and roll angles are always kept at very small values, and also that the induced disturbance produces only small effects on them.

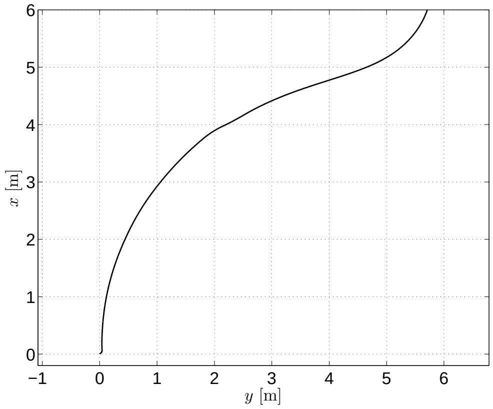


Fig. 8. A projection on the (x, y) plane of the trajectory flown by the rotorcraft.

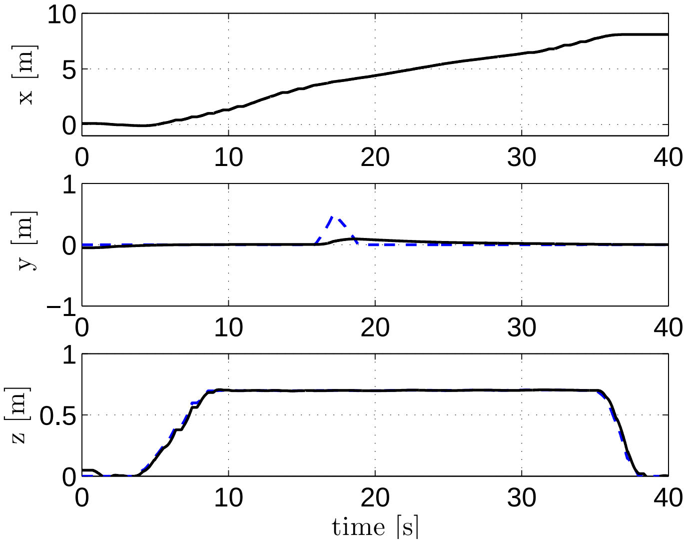


Fig. 9. The (x, y, z) states during the simulation test. The solid black line represents the behavior of the rotorcraft. Notice how the system remains stable despite a disturbance in the y state reference signal (dashed blue line) at $t = 18$ seconds. This demonstrates that the proposed controller is effective for dealing with these unexpected behaviors.

B. Real Time Experiments

The ultimate test for the proposed estimation and control strategies consists on verifying its proper operation when used during real-time experiments. Aiming at this goal, a road model having two smooth curves was built and installed in our experimental area. The experimental procedure can be described as follows.

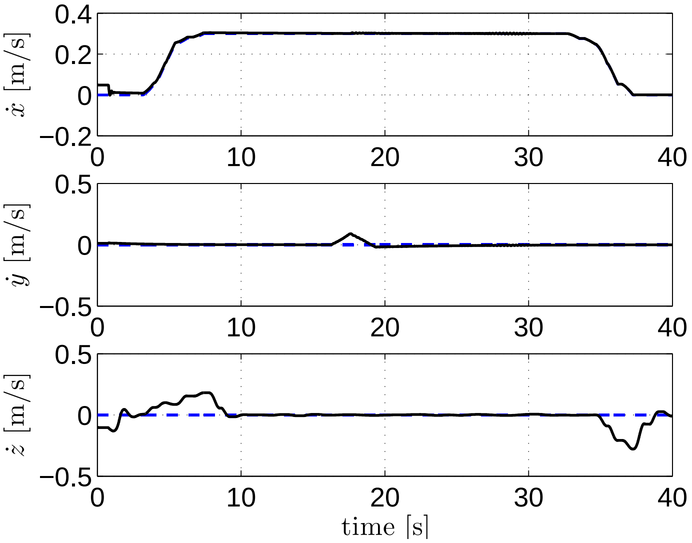


Fig. 10. Translational velocities during the simulation test. The solid black line represents the behavior of the rotorcraft. The forward velocity $\dot{x}$ is kept at 0.3 m/s, while the lateral velocity $\dot{y}$ and altitude velocity $\dot{z}$ are both kept around zero. Ramp profiles were used for increasing-decreasing the $\dot{x}$ reference value. The variation in $\dot{y}$ at $t \approx 18$ s is caused by the induced disturbance.

The rotorcraft starts its mission on the ground, exactly positioned over the road. Next, an autonomous take-off is performed, achieving a desired altitude of 0.70 m over the road. Once at this height, the embedded camera detects the road. The imaging information allows heading and lateral position stabilization for achieving a flight path well aligned w.r.t. the road. The vehicle is then required to navigate forward at a desired speed of $\dot{x} = 0.3$ m/s, while regulating its heading angle ψ and lateral position y w.r.t. the road. In order to achieve a smooth transition from a forward velocity of $\dot{x} = 0.0$ m/s to a forward velocity of $\dot{x} = 0.3$ m/s, a ramp velocity profile is implemented, in such a way that the velocity slowly increases after the take-off task. Once the vehicle is near to the end of the road segment, the landing is performed autonomously, and the rotorcraft descends slowly to the ground. Again, a ramp velocity profile is implemented, in such a way that the forward velocity slowly decreases from $\dot{x} = 0.3$ m/s to $\dot{x} = 0.0$ m/s before the landing task is engaged.

During the execution of the autonomous navigation experiment, an external force (disturbance) was applied in the lateral dynamics of the vehicle. The disturbance coerces the quad-rotor to displace arbitrarily in the y direction, making the imaging system unable to detect the road for

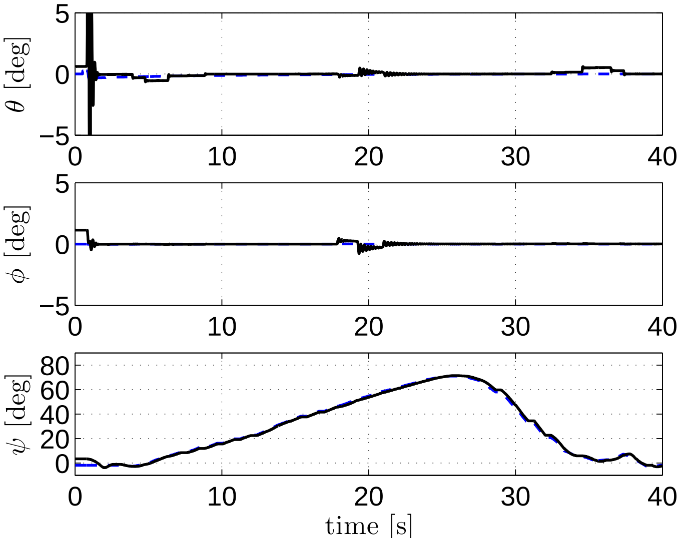


Fig. 11. The rotorcraft Euler angles during the simulation test. The solid black line represents the behavior of the rotorcraft. Notice that the pitch angle (θ) and roll angle (ϕ) are both kept small. The heading angle (ψ) is represented w.r.t. the inertial reference, and describes the heading that the vehicle must track in order to handle the two curves of the road. From the (ψ) graphic, notice that the difference between the actual behavior (black line) and expected behavior (dashed blue line) is very small.

several instants of time. Specifically, the external perturbation has the purpose of demonstrating the effectiveness of the estimation and control strategies when the switching procedure occurs. It is worth mentioning that, during the experimental application, the maximum time that the road stayed out of sight was around 3 seconds.

A 3-dimensional reconstruction of the path navigated by the rotorcraft during the real-time experiment is shown in Figure 12. In this Figure, the rotorcraft trajectory is represented w.r.t. an inertial coordinate frame $\mathcal{I}$, whose origin is located at the quad rotorcraft initial position. It can be seen that the vehicle performs the road following mission inside a rectangular area of 6×6 meters. The projection of the rotorcraft displacement in the (x, y) plane can be seen in Figure 13. It is worth mentioning that the forward displacement (relative to the body fixed frame) is obtained by integrating the optical flow measured in the forward direction. On the other hand, the rotorcraft lateral position is obtained from two methods. The first of them directly applies equation (10), which provides the lateral position of the vehicle relative to the road. The second method corresponds to the integration of the optical flow, in the lateral direction, during the instants of time when the road is not detected by the vision system.

The inertial coordinates shown in Figure 12 and Figure 13 are obtained after rotating both the (x, y) states around the z-axis using a gymbal like rotation matrix. Specifically, the rotation matrix makes use of the heading angle provided by the onboard IMU. The computation, which is described in equations (47) and (48), generates the 3-dimensional trajectory of the vehicle with respect to the inertial frame $\mathcal{I}$.

$$^{\mathcal{I}}x = \int_0^t \cos(\psi(t))\dot{x}(t)dt - \sin(\psi(t))y(t) \tag{47}$$

$$^{\mathcal{I}}y = \int_0^t \sin(\psi(t))\dot{x}(t)dt + \cos(\phi(t))y(t) \tag{48}$$

The three translational states are shown individually in Figure 14. Here, $x = 10$ m represents the longitudinal dimension of the road, which is also the total distance flown by the quad-rotor in the forward direction. This measurement was based on the integration of the optical flow sensed by the camera in the x direction. The y state represents the lateral position of the quad-rotor w.r.t. the road. An external perturbation in the y direction can be seen at $t = 25$ seconds. The rotorcraft translational velocities are shown in Figure 15. Notice that, during navigation, the forward speed $\dot{x}$ is maintained at 0.3 m/s, while the lateral $\dot{y}$ and altitude $\dot{z}$ velocities are kept near zero.

The switching between the two different state estimation approaches is illustrated in Figure 16. The variable s (the graphic in the middle) shows the instants of time when the road is being detected ($s = 1$) and when it is not being detected ($s = 0$). The upper graphic represents the desired heading angle that will align the rotorcraft heading with the road direction, as obtained from equation (20). The lower graphic represents the lateral position error as computed from equation (21). A zoom to the region where the switching occurs is shown in Figure 17.

Finally, the rotorcraft Euler angles, as obtained directly from the embedded IMU are shown in Figure 18. The pitch angle (θ), shown in the upper graphic, is kept to a small value which ensures forward displacement at the required velocity. The roll angle (ϕ), shown in the middle graphic, is also kept small. However, when it is required to correct the lateral position error induced at around $t = 25$ seconds, the roll magnitude increases until reaching again the desired lateral position w.r.t. the road. The lower graphic shows the heading angle (ψ) described by the

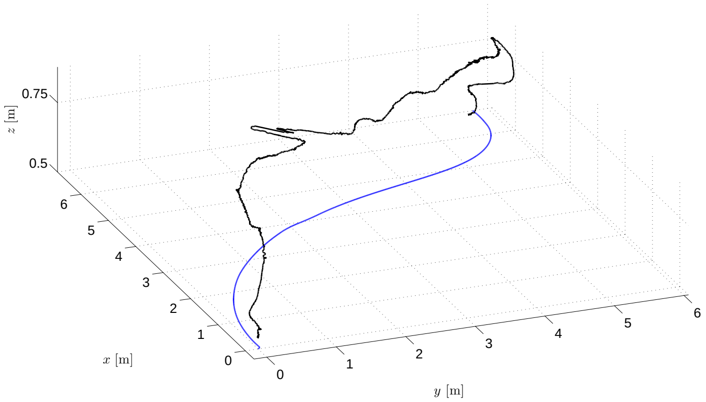


Fig. 12. Road-following mission: 3-dimensional reconstruction of the path navigated during the real-time experiment. The dashed blue line plotted in the x-y plane represents the position of the road model. The black line represents the actual vehicle displacement.

rotorcraft in order to handle the two curves of the road. This heading angle is represented w.r.t. the inertial reference frame whose origin is at the quad-rotor initial position.

Figure 19 shows an image of the rotorcraft in the experimental area during the real-time tests. In addition, a video showing the quad-rotor while performing the road following experiment can be seen at

`http://youtu.be/tmiApYi3tT4`

VII. CONCLUSION

The problem of road following using a quad rotorcraft equipped with a fully embedded imaging and control system was addressed. The goal of this research consists of estimating and tracking a road without a priori knowledge of the path to be tracked, as well as of deriving efficient estimation and control strategies for dealing with situations when the road is not detected by the vision system. Two main operational regions were defined: one for the case when the road is detected, and the other for the case when it is not. A switching between measurements coming from imaging and inertial sensors was used in order to estimate the vehicle parameters required in both regions of operation. In addition to this, a switching control strategy for stabilizing the vehicle lateral position was proposed. The system stability was verified not only in the two

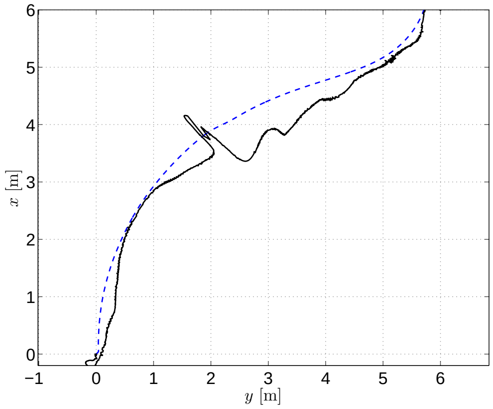


Fig. 13. The trajectory described by the rotorcraft, projected in the (x, y) plane. The dashed blue line represents the position of the road model. The black line represents the actual vehicle displacement.

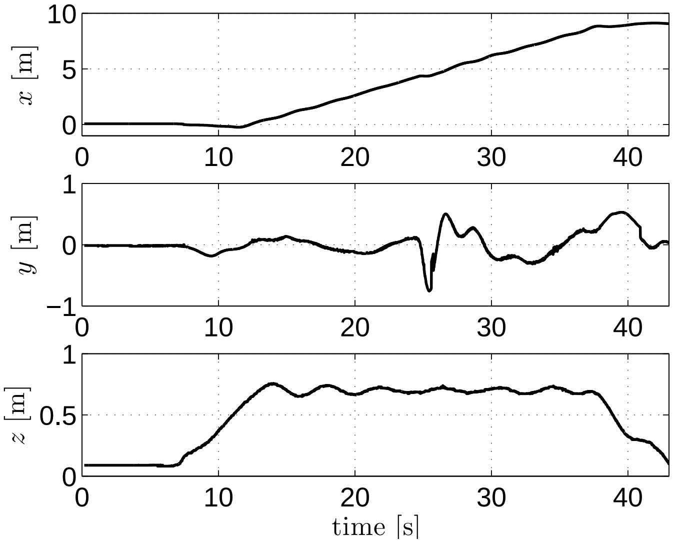


Fig. 14. The (x, y, z) rotorcraft states: The perturbation in the lateral dynamic (y state) is shown at $t = 25$ seconds.

operational regions, but also in the switching boundaries between them. The performance of the switching strategies for sensing and control was tested in a numerical simulation and a real time application. The experiment allowed to validate the effectiveness of the proposed methods for autonomously performing the task of road following in the presence of external disturbances and unexpected failures of the imaging system.

Future work will concern a hybrid scheme for switching not only different controller gains

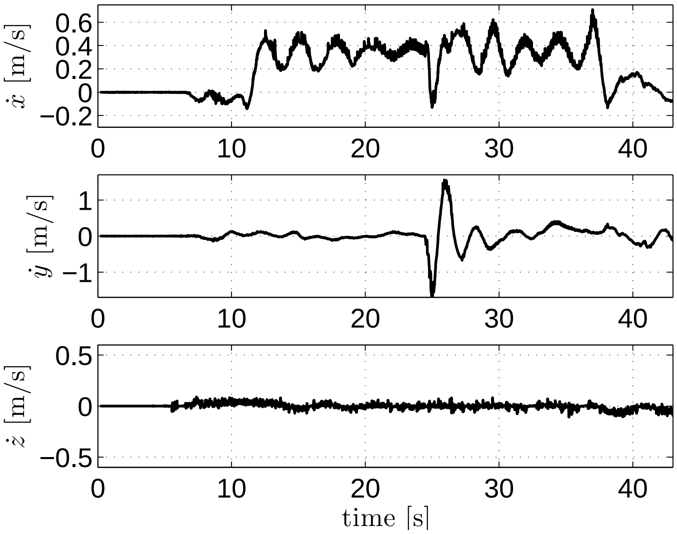


Fig. 15. Translational velocities during the experiment: The forward velocity $\dot{x}$ is kept at 0.3 m/s, while the lateral velocity $\dot{y}$ and altitude velocity $\dot{z}$ are both kept around zero. The variation in $\dot{y}$ at around 25 s is caused by the external perturbation.

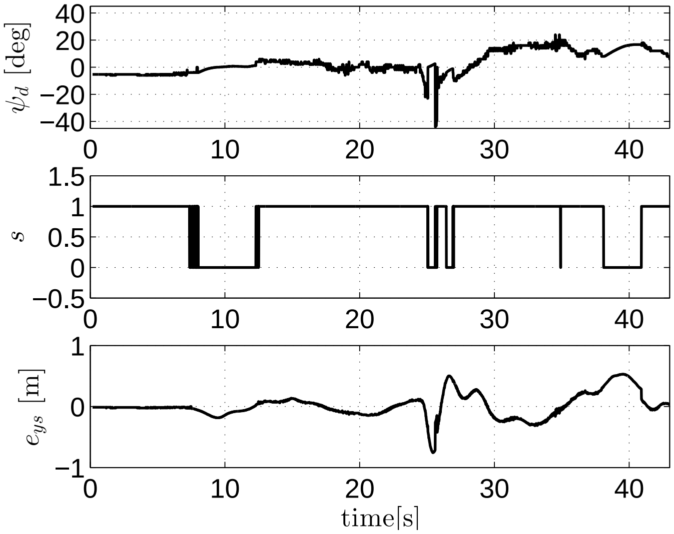


Fig. 16. Behavior of the switching signals ψ_{d}, s and e_{ys}. The graphic in the middle represents the instants of time when the line is being detected ($s = 1$) and when the line is not being detected ($s = 0$). During the take-off and landing phases the switching signal s is kept fixed at a value of $s = 1$. Once the desired height is reached, the signal s varies according to the situation encountered, i.e., $s = 1$ when the road is detected, and $s = 0$ for when the road is not detected.

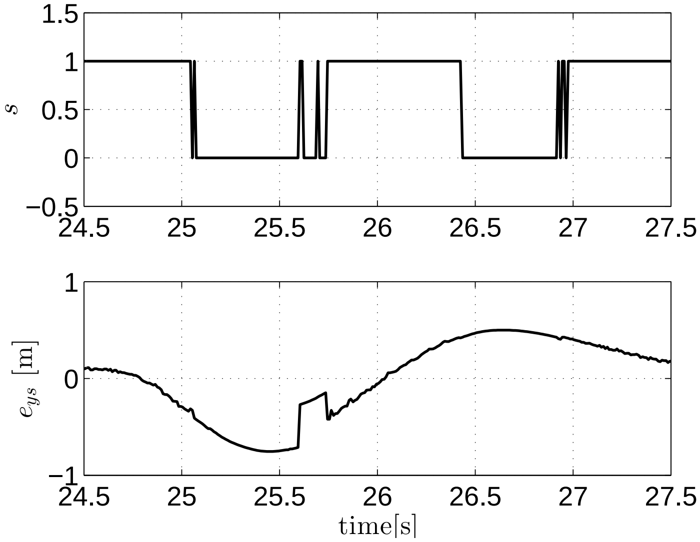


Fig. 17. A zoom to the region where the disturbance occurs: The upper graphic shows how the s signal changes between 1 and 0. The lower graphic shows how the switching strategy influences the lateral position measurement.

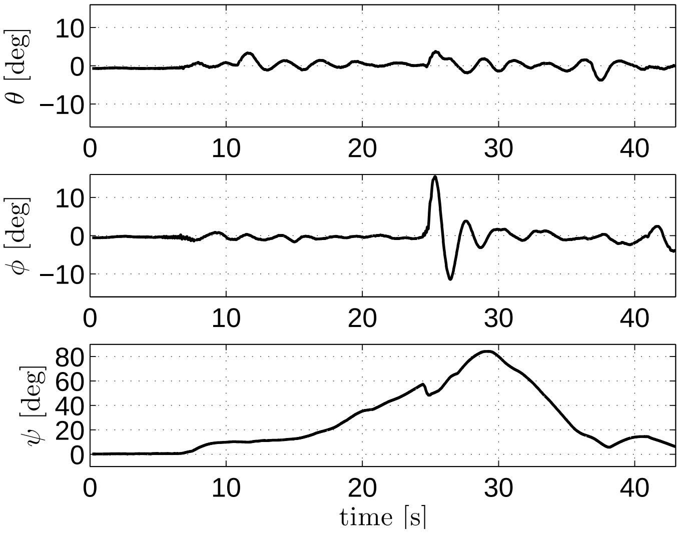


Fig. 18. The rotorcraft Euler angles as obtained directly from the embedded IMU: The pitch angle (θ) is kept to a value which ensures forward displacement at the required velocity. The roll angle (ϕ) is kept small. The heading angle (ψ) is represented w.r.t. the inertial frame $\mathcal{I}$.

but also different control strategies.

Fig. 19. Road-following mission: The rotorcraft flying in the experimental area during real-time tests.

REFERENCES

[1] E. Altug, J. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback", in *International Conference on Robotics and Automation*, Washington, DC, May 2002, pp. 72-77.

[2] S. Zingg, D. Scaramuzza, S. Weiss, and R. Siegwart, "Mav navigation through indoor corridors using optical flow", in *International Conference on Robotics and Automation*, Anchorage, Alaska, May 2010.

[3] D. Cabecinhas, C. Silvestre, and R. Cunha, "Vision-based quadrotor stabilization using a pan and tilt camera", in *Conference on Decision and Control*, Atlanta, GA, USA, December, 2010, pp. 1644-1649.

[4] L.R. García Carrillo, E. Rondón, A. Sánchez, A. Dzul, and R. Lozano, "Stabilization and trajectory tracking of a quad rotor using vision", *Journal of Intelligent and Robotic Systems*, vol. 61, no. 1-4, pp. 103-118, January, 2011.

[5] E. Rondón, L.R. García Carrillo, and I. Fantoni, "Vision-based altitude, position and speed regulation of a quad-rotor rotorcraft", in *International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, October, 2010, pp. 628-633.

[6] G. Hoffmann, H. Huang, S.L. Waslander, and C. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment", in *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, South Carolina, USA, 2007.

[7] J. Hespanha, O. Yakimenko, I. Kaminer, and A. Pascoal, "Linear parametrically varying systems with brief instabilities: An application to integrated vision/imu navigation", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, pp. 889-902, July, 2004.

[8] M. Oishi and C. Tomlin, "Switching in nonminimum phase systems: Application to vstol aircraft", in *American Control Conference*, Chicago, IL, Jun. 2000, pp. 838-843.

[9] F. Chaumette, "Image moments: a general and useful set of features for visual servoing", *Transactions on Robotics*, vol. 20, no. 4, pp. 713-723, August, 2004.

[10] L. Shapiro and G. Stockman, *Computer Vision*, 1st ed. Upper Saddle River, NJ, USA, Prentice Hall PTR, 2001.

[11] J. Shi and C. Tomasi, "Good features to track", in *Conference on Computer Vision and Pattern Recognition*, 1994, pp. 593-600.

[12] J. Bouguet, “Pyramidal implementation of the Lucas-Kanade feature tracker”, in *Technical Report*, Intel Corporation, Microprocessor Research Labs, 1999.

[13] L.R. García Carrillo, A. Dzul, R. Lozano, and C. Pégard, *Quad Rotorcraft Control: Vision-Based Hovering and Navigation*, Advances in Industrial Control, Springer, August 2012.

[14] S. Bouabdallah and R. Siegwart, “Backstepping and sliding-mode techniques applied to an indoor micro quadrotor”, in *International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 2259-2264.

[15] P. Kokotović, H.K. Khalil, and J.O’Reilly, *Singular Perturbation Methods in Control: Analysis and Design*, London Academic Press: Siam, 1999.

[16] R. Xu and U. Ozguner, “Sliding mode control of a class of underactuated systems”, *Automatica*, vol. 44, no. 1, pp. 233-241, January 2008.

[17] J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, 1990.

[18] D. Liberzon, *Switching in Systems and Control*, Boston, Birkhuser, 2003.

[19] J.P. Hespanha and A.S. Morse, “Stability of switched systems with average dwell-time”, in *Conference on Decision and Control*, December 1999, pp. 2655-2660.

[20] H.K. Khalil, *Nonlinear Systems*, New York, Prentice Hall, 2002.

[21] P. Peleties and R. DeCarlo, “Asymptotic stability of m-switched systems using Lyapunov-like functions”, in *American Control Conference*, June 1991, pp. 1679-1684.

[22] D. Liberzon, “Switched systems”, in *Handbook of Networked and Embedded Control Systems*, 2005, pp. 559-574.