# Design and Control of a Bimodal aerial robot for locomotion in flat and inclined surfaces

Miguel Vicente Pimentel
miguel.vicente.pimentel@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Portugal

April 2021

**Abstract**

Employing aerial robots, acting as mobile airborne sensors, is extremely useful for many surveillance and inspection missions. Due to the strict constraints of commonly available aerial platforms, most aerial inspection missions are limited to distant perception of targets, without the possibility of coming into centimeter-range proximity or even in physical contact with them. This thesis describes a Bimodal aerial robot, consisting of a common quadrotor equipped with two passive wheels, that allows flying, approaching, landing and moving on planar and inclined surfaces, suitable for micro-level inspection of large areas. This work describes in detail the mathematical model of the passive two-wheeled quadrotor for its different modes of operation: flight, ground and inclined surfaces. Furthermore, two different motion controllers are proposed, for all three modes of locomotion, and their performance is evaluated through a series of simulated experiments that make use of the hybrid characteristics of the vehicle.
**Keywords:** Aerial Robots, Hybrid Locomotion, Motion Control, Multi-Modality

## 1. Introduction

The use of micro aerial robots for autonomous inspection and surveillance tasks has grown significantly over the past decade. Aerial robots can rapidly access target areas by flying over obstacles or cluttered terrain and reach regions that are inaccessible to humans or other ground robots. They can provide an elevated and birds eye view sensing of the environment and approach hard-to-reach structures, while enhancing the efficiency and safety of inspection missions. Inspection of solar panels and photo-voltaic installations [1], inspection of an industrial structure [2], detecting heat leakages in buildings [3] and inspection of a vessel [4] are among many foreseen applications for aerial robots.

While the benefits of aerial robots for distant inspection of infrastructures have been extensively demonstrated [5], limited airborne solutions exist that can perform such tasks in centimeter-range proximity to structures, or even while in physical contact with its surface. Remotely detected failures by distant aerial robots always need to be examined closely by the operators and often require additional efforts to detect the type and source of the failures. This is due to the strict limitations imposed by common aerial platforms, which are unable to get close or even come into contact with structures without the risk of collision and even crashing. They have a very limited flight time due to the small battery size they can carry, and their operation can easily be disrupted by disturbances such as wind [6]. An effective approach for increasing the versatility of an aerial robot, or any other robot in general, is to provide it with additional modes of locomotion, to obtain platforms that can adapt to different situations, moving in aerial, terrestrial or even aquatic environments [7]. For instance, a robot that is able to both fly and roll, can fly towards a structure, land and move in its surface, reducing the energy consumption and minimising the acoustic noise signature, to perform the most various of inspection tasks.

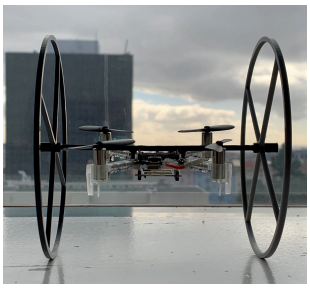

Figure 1: Prototype of the Bimodal aerial robot, consisting of a quadrotor with two passive wheels.

The broader goal of this work is to design a simple Bimodal aerial robot that can perform both macro and micro-level inspection of structures. This the-

sis describes the effort towards designing a robot which is capable of landing and moving in different surfaces, while employing the same set of actuators for all modes, and developing its fundamental control approach. Such robot is expected to enhance the efficiency of many aerial inspection missions as it can land or physically contact with the inspecting surface, to obtain a stable inspection and increase its operation time due to reduced consumption of energy.

The outline of this paper is as follows: Section 2 provides a summary of previous work on the topic of multi-modal aerial/ground robots. Section 3 describes the Bimodal robot system and provides the mathematical model for different states of operation. Section 4 describes two control strategies to control the motion of the robot. Section 5 describes the experiments and results, presenting the behaviour of the system in its different modes of operation.

## 2. Previous work

Over the past years, multi-modal robots with two or more modes of locomotion have appeared in the literature, in particular, vehicles with the ability to switch between aerial and terrestrial locomotion have received a great deal of attention as they provide a good trade-off between energy consumption and mobility. A straightforward approach to obtain a multi-modal robot is to employ different sets of actuation mechanisms for each mode of locomotion. For example, in [8, 9], where a hybrid aerial/terrestrial robot is developed by equipping a ground robot with four rotors. Such multi-actuation solutions can demonstrate an effective performance for all modes of operation, however, result in expensive and heavy hardware, that have poor energy efficiency due to the added weight.

Contrarily, in [7], an all-round two-wheeled quadrotor was designed to work in air, land and sea. It consists of an UAV with 2 rolling protective frames that also act as wheels. The robot has an extra weight attached to it in order to keep the balance in the event of a poor landing, which reduces the energy efficiency. Furthermore, in ground mode, the controller does not account for the non-holonomic nature of the system which can lead to under-performance in several situations. Likewise, in [10], a cylindrical cage is attached to a quadrotor through two revolute joints to allow aerial and terrestrial movement.

At the same time, several studies have proposed aerial robots capable of moving on vertical surfaces, from flying robots that can perch onto walls using microspine technology [11], suction cups [12] or dry-adhesive grippers [13], quadcopters that have tilt-mechanisms allowing them to change the direction of their thrusters and attach to walls through a normal force [14], to robots that can fly and perch using a compliant, underactuated gripping mechanism [15]. Such designs lead to slow locomotion on vertical surfaces and implicate additional actuators and mechanisms that increase the weight and complexity.

Examples of other distinctive concepts include a two-wheeled ground robot with a helicopter mechanism folded into its own body [16, 17, 18], a flying robot that shares its structure for different modalities, using its wings for both flying and walking [19] or the "DUCK", a quadrotor combined with passive legs that uses its thrust for dynamic walking [20]. The design employed by [21] is also singular, for using skateboard steering truck wheels below two of the rotors for turning, being successful in providing a rolling mechanism in semi-smooth surfaces without altering the avionics of the original quadrotor.

Finally, it is worth mentioning the efforts made by Team CoSTAR from NASA Jet Propulsion Laboratory [22] with its Nebula autonomy solution, which address autonomous exploration of extreme environments (for planetary exploration). While this robot ecosystem is highly complex, it comprises drones with hybrid rolling/flying mechanisms, such as the Drivocopter [23], which is a quadcopter UAV with four spherical shells surrounding the propellers, that act as independent actuated wheels. However, for the development of this work, their most interesting design is presented in [24], where a control scheme was developed for a passive two-wheeled hybrid UAV. This work exploited the similarity between differential flatness mappings of quadrotors and non-holonomic vehicles, in order to design a local planner that generates feasible trajectories for both methods of mobility, using the same representation. Nonetheless, the mathematical model of the robot is not given, nor the behaviour of the transition from one mode to the other.

## 3. Robot description and mathematical model

The Bimodal robot system, shown in Figure 1, is composed of a common X-configuration quadrotor attached to two passive all-round wheels through an axle. Each wheel can rotate freely and independently around its axis, and the center of mass of the axle+wheels system coincides with the center of mass of the quadrotor. This simple design allows utilizing the same set of actuators for controlling the robot motion in all modes of locomotion. On top of the added mode of mobility, the wheels offer all-round protection for the platform, ensuring that the body and propellers of the quadrotor are well protected against impacts and collisions.

When the robot is operating in the air, its dy-

namics will follow the same model as that of a conventional quadcopter [25, 26], with only increased inertia values due to the addition of the wheels. The dynamic models for when the robot is moving on the ground and on an inclined surface are described in the following subsections.

### 3.1. Ground model

When on ground, the motion of the robot will be similar to that of a Unicycle, however, as the wheels are passive, the robot movement relies on the thrust generated by the propellers. Figure 2 shows a schematic of the robot in ground mode, describing an Inertial frame ($\mathbf{S_I}$), a Body fixed reference frame ($\mathbf{S_B}$), and a Rolling frame ($\mathbf{S_R}$) that is attached to the centre of mass but with the $x_r - y_r$ plane always parallel to the surface. The pose of the robot is given by the displacement and rotation of $S_B$ in $S_I$:

$$\mathbf{P} = (x, y, z), \qquad \mathbf{\Phi} = (\varphi, \theta, \psi), \tag{1}$$

where $\varphi, \theta, \psi$ are the roll, pitch and yaw, respectively (Euler angles).

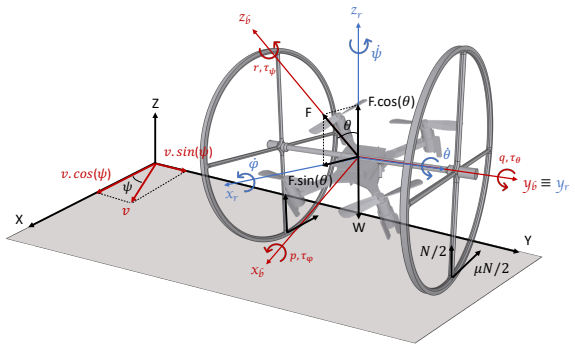


Figure 2: Model of Bimodal robot on a planar surface.

In this mode, it is required that both wheels are always kept on the ground, ensuring the robot only moves on the 2D X-Y plane. Hence, the altitude is kept constant at surface level, i.e, $\dot{z} = 0 = \dot{v_z}$, and the vertical thrust component imposed by the rotors must be smaller than its weight:

$$|F\ cos(\theta)| \leq mg. \tag{2}$$

Furthermore, it is assumed that the wheels roll on the surface without slipping, resulting in the robots velocities respecting the non-holonomic kinematics imposed by the wheels:

$$\dot{x}sin(\psi) - \dot{y}cos(\psi) = 0. \tag{3}$$

For this to be true, the robot must not experience any roll movements:

$$\varphi = \dot{\varphi} = 0. \tag{4}$$

To generate motion, the vehicle needs to pitch around the $y_r$-axis and have a thrust force satisfying constraint (2). The horizontal component of the thrust makes the robot move on the plane with velocity $v$, that can only have a component along the $x_r$ direction, while the torques produced by the rotors will be responsible for the turning motion around the $z_r$ direction. The forces acting on the Rolling frame are:

$$\begin{bmatrix} Fs_\theta \\ 0 \\ Fc_\theta \end{bmatrix} + \begin{bmatrix} -\mu N \\ 0 \\ N \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} = m \begin{bmatrix} \dot{v} \\ 0 \\ 0 \end{bmatrix}, \tag{5}$$

from where the expression for the acceleration is obtained,

$$\dot{v} = \frac{F}{m} s_\theta - \mu \frac{N}{m}. \tag{6}$$

The Normal force is calculated with $N = mg - Fc_\theta$, $\mu$ is the coefficient of friction (between the wheel and the surface) and $s_\bullet$ and $c_\bullet$ represent $sin(\bullet)$ and $cos(\bullet)$, respectively.

When moving on a horizontal plane, the robots kinematics must follow the non-holonomic constraint (3), and can thus be formulated as follows:

$$\dot{x} = v\ c_\psi, \qquad \dot{y} = v\ s_\psi, \tag{7}$$

whereas the rotational rates in $S_B$, $\omega = (p, q, r)$, are computed by taking into account the no-roll condition (4):

$$p = -\dot{\psi} s_\theta, \qquad q = \dot{\theta}, \qquad r = \dot{\psi} c_\theta. \tag{8}$$

The rate around $x_b$ compensates for the coupling between the rotational directions and guarantees that $\dot{\varphi}$ always remains zero. Hence, it is necessary to control all three torques generated by the thrust, $\tau = (\tau_\varphi, \tau_\theta, \tau_\psi)$, whose relationships with the angular accelerations are defined through:

$$\begin{aligned} \dot{p} &= \frac{I_y - I_z}{I_x} qr + \frac{\tau_\varphi}{I_x}, \\ \dot{q} &= \frac{I_z - I_x}{I_y} pr + \frac{\tau_\theta}{I_y}, \\ \dot{r} &= \frac{I_x - I_y}{I_z} pq + \frac{\tau_\psi}{I_z}, \end{aligned} \tag{9}$$

where $\mathbf{I} = diag(I_x, I_y, I_z)$ is the Inertial tensor.

Equations (6)-(9) represent the state-space model of the robot in ground mode, which is fully described by the state vector $\xi_g = (x, y, v, \theta, \psi, p, q, r)^T$.

### 3.2. Inclined Surface Model

Although the previously defined robot model can be a good approximation for surfaces with slight inclinations, it will not be a suitable choice for larger inclinations, such as when moving on the surface

of structures, as is the intended application of this work.

Figure 3 illustrates the diagram of the robot moving up a surface with an inclination angle of $\gamma \in \ ]0, \frac{\pi}{2}[$. The following conditions are imposed for this mode of navigation:

1. The inclination angle $\gamma$ and the orientation of the slope relative to $S_I$, $\psi_0$, are known.
2. The robot will only move up and down the slope without steering, i.e, $\psi(t) = \psi_0$.

The second condition is to ensure a safe navigation along any type of slope, independently of its inclination and friction properties. This is because the design of the robot only allows the thrust force to point in the direction of $z_b$, meaning if the vehicle steers to the left or right, there will be no thrust component to move the system against the gravity force (when $\psi - \psi_0 = \pi/2$). For this mode, the constraints expressed by equations (3) and (4) will remain unchanged, to ensure the wheels are kept on the surface. To satisfy the new conditions, the yaw rate must remain zero, $\dot{\psi}(t) = 0$, and to avoid take-off, constraint (2) must be re-formulated as

$$|F\ cos(\theta + \gamma)| < mg\ cos(\gamma). \tag{10}$$

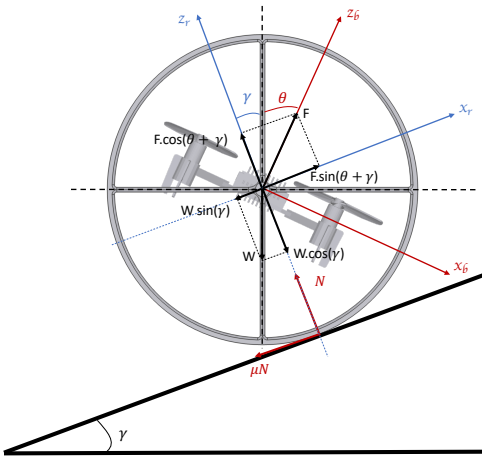


Figure 3: Model of Bimodal robot going up a slope.

Proceeding in a similar fashion as before, from the balance of forces acting on the Rolling frame,

$$\begin{bmatrix} F\ s_{(\theta+\gamma)} \\ 0 \\ F\ c_{(\theta+\gamma)} \end{bmatrix} + \begin{bmatrix} -\mu N \\ 0 \\ N \end{bmatrix} + \begin{bmatrix} mg\ s_\gamma \\ 0 \\ mg\ c_\gamma \end{bmatrix} = m \begin{bmatrix} \dot{v} \\ 0 \\ 0 \end{bmatrix}, \tag{11}$$

we can formulate the expression for the acceleration as

$$\dot{v} = \frac{F}{m} s_{(\theta+\gamma)} - g.s_\gamma - \mu \frac{N}{m}, \tag{12}$$

where the Normal force is $N = mgc_\gamma - Fc_{(\theta+\gamma)}$.

The kinematics of the robot moving along the slope direction respect the following expression:

$$\dot{x} = vc_\gamma c_\psi, \qquad \dot{y} = vc_\gamma s_\psi, \qquad \dot{z} = vs_\gamma, \tag{13}$$

and the rotational dynamics follow the same equations of ground mode, meaning that the state-space model can be fully characterized by equations (8), (9), (12) and (13). The robots state in inclined mode can, however, be described by the simplified state vector $\xi_i = (x, y, z, v, \theta, q)^T$ and the slope properties $\zeta = (\gamma, \psi_0)$.

## 4. Motion control system

Two different controllers were developed for the robot, based on a cascaded architecture, where an inner-loop regulates the attitude of the robot $(\varphi, \theta, \psi)$, through the acting torques $(\tau)$ and an outer loop determines the necessary rotation angles, that drive the position error to zero, and the thrust force $(F)$. For the three modalities of the robot, two approaches were taken:

- A linear approach that employs a proportional, integral and derivative controller (PID), to control the robot within a neighbourhood of its equilibrium conditions;
- A non-linear strategy, that applies a dynamic feedback linearization law (DFL) to the attitude dynamics, for improved robustness and agility;

Figure 4 shows the cascaded controller architecture for surface locomotion. The position controller is responsible for computing the reference pitch and thrust force to move the vehicle from the current position to the desired one. The yaw angle will be computed so that the orientation of the vehicle is aligned with the destination, for the case of flat surfaces, and it will be a constant according to the specifications of the inclination $(\zeta)$, for the case of inclined surfaces. The attitude controller regulates the pitch and yaw references and ensures there are no roll movements $(\varphi(t) = 0)$ and will be equal for both flat and inclined surfaces. What follows is a brief overview of each method, derived for the three different modalities of the platform.

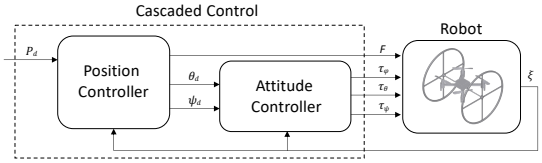


Figure 4: Architecture of a Cascaded Controller for both surface modes.

4.1. Cascaded PID Controller

As this controller is commonly described in the literature for in-flight quadrotors [25], description of the flight-mode is omitted for the sake of brevity. Derivation of the control equations for flat and inclined surfaces is reported below.

Assuming the rotational dynamics can be considered decoupled from one another, $I\dot{\omega} = \tau$, which is true for small pitch angles around the equilibrium point, the attitude controller receives the reference pitch and yaw and determines the necessary angular accelerations on the Rolling frame:

$$\begin{aligned} U_\theta &= K_\theta^p.e_\theta + K_\theta^d.\dot{e_\theta} + K_\theta^i.\int_0^t e_\theta \ dt, \\ U_\psi &= K_\psi^p.e_\psi + K_\psi^d.\dot{e_\psi}, \end{aligned} \tag{14}$$

where $e_\theta$ and $e_\psi$ are the smallest angular differences between the desired and the current orientation and $K_\Phi$ are positive constant gains. In the body-frame, the necessary torques to produce these angular accelerations are given by:

$$\begin{aligned} \tau_\varphi &= -I_x \ U_\psi s_\theta, \\ \tau_\theta &= I_y \ U_\theta, \\ \tau_\psi &= I_z \ U_\psi c_\theta. \end{aligned} \tag{15}$$

**Ground Mode**: The position controller for flat surfaces considers a rotated version of the position error, expressed in $S_R$, to compute a desired velocity, through a proportional control law, while respecting the non-holonomic nature of the robot,

$$v_d = K_x^p.e_x^R, \tag{16}$$

where $e_x^R = c_\psi(x_d - x) + s_\psi(y_d - y)$. Then, another feedback loop determines the longitudinal acceleration $a_d^R$ for regulating the velocity error, $e_v = v_d - v$, which is used to compute the reference pitch angle:

$$a_d^R = K_v^p.e_v^R + K_v^i.\int_0^t e_v^R dt, \tag{17}$$

$$\theta_d = sin^{-1}(\frac{a_d^R.m}{f_0}), \quad |\theta_d| < \frac{\pi}{6}, \tag{18}$$

where $f_0$ is a constant thrust force satisfying equation (2).

Conversely, the error along the $y_r$ direction converges to zero when $\psi \to \psi_d$, as the orientation of the wheels are aligned with the destination,

$$\psi_d = atan2(y_d - y, x_d - x) - k\pi, \tag{19}$$

where term $k$ is used to take into account the robot's ability to move forwards or backwards and to avoid unnecessary turns:

$$k = \begin{cases} 0 & \text{if } v_d > 0 \\ sign\big[atan2(e_y^I, e_x^I)\big] & \text{if } v_d < 0 \end{cases}. \tag{20}$$

**Inclined Surface**: The linear position controller for moving on an inclined surface forces the robot's attitude to be in compliance with the inclined surface, i.e, $\theta_d = \frac{\pi}{2} - \gamma$ and $\psi_d = \psi_0$. In this state, the thrust force vector is always parallel to the slopes direction and the translational dynamics can be simplified to $\dot{v} = \frac{F}{m} - g s_\gamma$, thus inverted to find the necessary thrust that moves the robot up or down the slope,

$$F = m(a_d^R + g \ s_\gamma). \tag{21}$$

The desired acceleration, $a_d^R$, is obtained from the altitude error projected on the Rolling frame, $e_z^R = (z_d - z)/s_\gamma$, where $z_d$ is the reference height (how high on the surface the robot should move), through two feedback control loops:

1. A PI controller to obtain $v_d$ from error $e_z^R$;
2. A PID controller to control $a_d$ from error $e_v$;

4.2. Dynamic Feedback Linearisation (DFL) Control

The linear approach considered in the previous Section only allows for small rotations around the equilibrium condition, which means it doesn't take advantage of the full manoeuvrability the robot has to offer. Alternatively, a feedback linearising controller is proposed, that transforms the closed-loop system into an equivalent, linear and controllable one, under state transformation [27]. However, performing an exact linearization on the ground dynamics would result in a singularity when the vehicle is stopped ($v = 0$), which has been proven to be structural for nonholonomic systems [28]. Instead of using a full state feedback, a dynamic inversion is used to control the attitude variables, while the position is tracked by an outer loop, with the same architecture as before (Figure 4).

**Flight Mode -** The output to be controlled is the height and the attitude, $\eta_f = [z, \varphi, \theta, \psi]^T$, which can be differentiated twice so that the input $\mathbf{u}$ appears in a nonsingular way, resulting in

$$\ddot{\eta}_f = [\ddot{z}, \ddot{\varphi}, \ddot{\theta}, \ddot{\psi}]^T = l_f(\xi) + \mathbf{J_f}(\xi).\mathbf{u}, \tag{22}$$

with:

- $\mathbf{J_f}(\xi) = \begin{bmatrix} \frac{c_\varphi c_\theta}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{I_x} & \frac{s_\varphi t_\theta}{I_y} & \frac{c_\varphi t_\theta}{I_z} \\ 0 & 0 & \frac{c_\varphi}{I_y} & -\frac{s_\varphi}{I_z} \\ 0 & 0 & \frac{s_\varphi}{I_y.c_\theta} & \frac{c_\varphi}{I_z.c_\theta} \end{bmatrix}$
- $l_{f_1}(\xi) = -g$
- $l_{f_2}(\xi) = \frac{I_y - I_z}{I_x} qr + \frac{I_z - I_x}{I_y} prs_\varphi t_\theta + \frac{I_x - I_y}{I_z} pqc_\varphi t_\theta + \dot{\varphi} t_\theta (c_\varphi q - s_\varphi r) + \dot{\theta} sec(\theta)(s_\varphi q + c_\varphi r)$
- $l_{f_3}(\xi) = \frac{I_z - I_x}{I_y} prc_\varphi - \frac{I_x - I_y}{I_z} pqs_\varphi - \dot{\varphi}(s_\varphi q + c_\varphi r)$

- $l_{f_4}(\xi) = \frac{I_z - I_x}{I_y} prs_\varphi sec(\theta) + \frac{I_x - I_y}{I_z} pqc_\varphi sec(\theta) + \dot{\varphi} sec(\theta)(c_\varphi q - s_\varphi r) + \dot{\theta} t_\theta sec(\theta)(s_\varphi q + c_\varphi r)$
- $l_f(\xi) = [l_{f_1}, l_{f_2}, l_{f_3}, l_{f_4}]^T$

Since the determinant of the decoupling matrix $\mathbf{J_f}(\xi)$ is given by

$$det(\mathbf{J_f}) = \frac{c_\varphi}{m I_x I_y I_z}, \tag{23}$$

the matrix is invertible for all $\det(\mathbf{J_f}) \neq 0 \Leftrightarrow \varphi \neq \pm\frac{\pi}{2}$, which allows to formulate the expression for the input commands as

$$\mathbf{u_f} = \mathbf{J_f^{-1}}(\xi).\big[\vartheta_f - l_f(\xi)\big], \tag{24}$$

which yields the integrator chain $\eta_f = \vartheta_f$, that is controllable [27]. Nonetheless, this controller presents a singularity when $\varphi = \pm\frac{\pi}{2}$ or $\theta = \pm\frac{\pi}{2}$, that must be taken into account.

The tracking component $\vartheta_f(\xi)$ is chosen as:

$$\vartheta_{f_1} = \ddot{z}_d + k_z^1(\dot{z}_d - \dot{z}) + k_z^0(z_d - z), \tag{25a}$$
$$\vartheta_{f_2} = \ddot{\varphi}_d + k_\varphi^1(\dot{\varphi}_d - \dot{\varphi}) + k_\varphi^0(\varphi_d - \varphi), \tag{25b}$$
$$\vartheta_{f_3} = \ddot{\theta}_d + k_\theta^1(\dot{\theta}_d - \dot{\theta}) + k_\theta^0(\theta_d - \theta), \tag{25c}$$
$$\vartheta_{f_4} = \ddot{\psi}_d + k_\psi^1(\dot{\psi}_d - \dot{\psi}) + k_\psi^0(\psi_d - \psi). \tag{25d}$$

The outer layer is responsible for computing the reference angles to stabilise the position, which can be done by inverting the translational dynamics,

$$\varphi_d = arcsin[\frac{m}{F}(U_y.c_\psi - U_x.s_\psi)], \tag{26a}$$
$$\theta_d = arcsin[\frac{m}{F c_\varphi}(U_x.c_\psi + U_y.s_\psi)], \tag{26b}$$

where the desired accelerations are found with the following PD control laws:

$$U_x = K_x^p.e_x + K_x^d.\dot{e}_x, \tag{27a}$$
$$U_y = K_y^p.e_y + K_y^d.\dot{e}_y. \tag{27b}$$

**Surface Mode -** The output vector will consist of the three attitude angles, $\eta_s = [\varphi, \theta, \psi]^T$, where the roll angle is considered to find the input values that guarantee there are no rotations around axis $x_r$ (condition 4). This vector is differentiated twice, until the input torques appear in a non-singular way, i.e,

$$\ddot{\eta}_s = [\ddot{\varphi}, \ddot{\theta}, \ddot{\psi}]^T = l_s(\xi) + \mathbf{J_s}(\xi).\tau, \tag{28}$$

where:

- $\mathbf{J_s}(\xi) = \begin{bmatrix} \frac{1}{I_x} & 0 & \frac{t_\theta}{I_z} \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{c_\theta I_z} \end{bmatrix}$, $l_s(\xi) = \begin{bmatrix} l_{s1}(\xi) \\ l_{s2}(\xi) \\ l_{s3}(\xi) \end{bmatrix}$,
- $l_{s1}(\xi) = \frac{I_y - I_z}{I_x} qr + \frac{I_x - I_y}{I_z} pqt_\theta + qrsec^2(\theta)$,
- $l_{s2}(\xi) = \frac{I_z - I_x}{I_y} pr$,
- $l_{s3}(\xi) = \frac{I_x - I_y}{I_z} pqsec(\theta) + rt_\theta sec(\theta)$.

Considering that the determinant of the decoupling matrix, $det(\mathbf{J_s}) = 1/(I_x I_y I_z.c_\theta)$, is non-singular for all $\theta \neq \pm\frac{\pi}{2}$, it can be inverted to find the dynamic compensator for surface mode,

$$\tau = \mathbf{J_s}^{-1}(\xi).\big[\vartheta_s(\xi) - l_s(\xi)\big], \tag{29}$$

where the tracking component $\vartheta_s$ is chosen by adopting a polynomial control law, that takes into account the imposed conditions (2)-(4) and is robust to disturbances in the roll direction:

$$\begin{aligned} \vartheta_{s1} &= k_\varphi^1(-\dot{\varphi}) + k_\varphi^0(-\varphi), \\ \vartheta_{s2} &= k_\theta^1(-\dot{\theta}) + k_\theta^0(\theta_d - \theta), \\ \vartheta_{s3} &= k_\psi^1(-\dot{\psi}) + k_\psi^0(\psi_d - \psi). \end{aligned} \tag{30}$$

In this way, substituting (29) in (28) yields a closed-loop integrator chain $\ddot{\eta_s} = \vartheta_s$, which is guaranteed to converge with the use of (30), except when the state of the robot comes in proximity of the singular point $\theta = \pm\frac{\pi}{2}$, which needs to be taken into account by the outer layer.

**Flat Surface -** The remaining outputs can be stabilised in a similar fashion as before, finding the desired acceleration via position feedback and determining the pitch angle with relationship (18), however, allowing the reference angle to belong to a larger subspace, $|\theta_d| < \pi/3$. This avoids the singularity but concedes extra agility to the robot. The yaw reference is again found using equation (19). Furthermore, a valve input control strategy is used to regulate the thrust input when the pitch variable gets saturated [29], allowing to increase the thrust force when necessary but keeping it in its nominal value, $f_0$, for shorter movements.

**Inclined Surface -** The cascaded PID controller for the inclined case relied on maintaining the vehicle on a static pose, only increasing or decreasing the thrust force to move up and down, respectively. Here, a valve scheme will be applied instead, allowing to regulate the pitch angle for the primary control and the thrust force as a secondary control, when the pitch gets saturated, as follows:

$$\theta_d = sin^{-1}\big[\frac{m}{f_0}(a_d^R + g.s_\gamma)\big] - \gamma, \tag{31}$$

$$F = f_0 + K_a^f \Delta a + K_\theta^f \Delta\theta. \tag{32}$$

The thrust nominal value, $f_0 = mg\ s_\gamma$, guarantees the robot stays at equilibrium when stopped, $K_a^f, K_\theta^f$ are positive constant gains and $\Delta a$, $\Delta\theta$ are deviation variables, that represent the amount of acceleration lost in the saturation process and are

thus defined according to:

$$\Delta a = \begin{cases} a_d - a_d^{max} & \text{if } a_d > a_d^{max} = \frac{f_0}{m} - g.s_\gamma \\ 0 & \text{if } a_d^{min} \leq a_d \leq a_d^{max} \\ a_d - a_d^{min} & \text{if } a_d < a_d^{min} = -\frac{f_0}{m} - g.s_\gamma \end{cases},$$

$$\Delta\theta = \begin{cases} \theta_d - \theta_d^{max} & \text{if } \theta_d > \theta_d^{max} = \frac{\pi}{2} - \gamma \\ 0 & \text{if } \theta_d^{min} \leq \theta_d \leq \theta_d^{max} \\ \theta_d - \theta_d^{min} & \text{if } \theta_d < \theta_d^{min} = 0 \end{cases}. \quad (33)$$

The acceleration saturation values are defined to avoid domain errors on the $sin^{-1}$ function and the pitch saturation ensures the wheels stay in contact with the surface, which could not be prevented if, for example, the robot pitched in the direction opposite to the surface.

## 5. Experiments and results

A simulation environment was created using *RotorS Simulator* [30], a well-known tool to assess the behaviour of aerial robots in the ROS and Gazebo ecosystem. The Bimodal robot was modelled in the simulator, by equipping an X-configuration quadrotor with 2 passive wheels, allowing to test the proposed controllers and evaluate the closed-loop response of the system. The PID and DFL performance was compared for the three modes of operation, through a series of step tests that allow to compare the response time, the settling time and the overshoot. These results are summarised in Table 1, where an increase in efficiency when using the feedback linearising controller is observed, as expected.

| | | Rise Time [s] | | Settling Time [s] | | Overshoot | |
|---|---|---|---|---|---|---|---|
| | | PID | DFL | PID | DFL | PID | DFL |
| Flight | $x(t)$ | 1.69 | 1.14 | 5.31 | 4.08 | 6.8% | 0.6% |
| | $y(t)$ | 1.70 | 1.19 | 5.33 | 4.06 | 6.74% | 0.60% |
| | $z(t)$ | 0.78 | 0.68 | 2.56 | 1.11 | 8.7% | 0.38% |
| | $\psi(t)$ | 1.15 | 0.58 | 2.17 | 1.02 | 0% | 0.4% |
| Ground | $x(t)$ | 3.15 | 1.88 | 5.50 | 3.38 | 0.17% | 0.03% |
| | $y(t)$ | 3.25 | 2.14 | 6.70 | 5.08 | 0.77% | 0.22% |
| Inclined | $z(t)(\gamma = 0.5)$ | 1.62 | 1.12 | 4.52 | 2.34 | 2.32% | 0% |
| | $z(t)(\gamma = 0.8)$ | 1.43 | 1.62 | 18.44 | 4.52 | 11.68% | 2.32% |

Table 1: Step response characteristics of PID and DFL controllers.

Videos from experiments comparing the controllers performance for different modes of operation are available: Fight[1], Ground[2], Inclined[3] and Hybrid[4].

### 5.1. Flight Motion

Figure 5 shows the step response of the system in flight mode, using the PID and the DFL controllers. For both cases, it is possible to see that the altitude and yaw responses are faster than the longitudinal and lateral ones, consequence of having a cascaded architecture in which position is dependent on the stabilisation of the pitch and roll angles. As displayed in Table 1, the DFL controller eliminates the overshoot seen with the linear approach, while decreasing the time it takes for the system to converge with the setpoint.

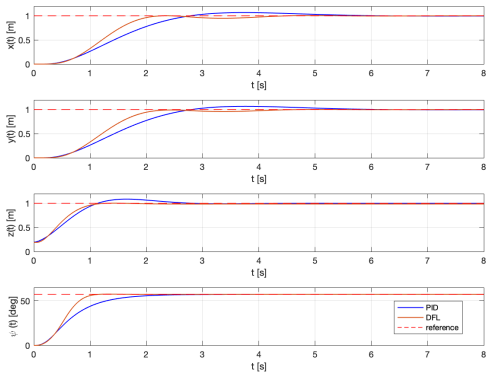


Figure 5: Step response of Flight controllers.

### 5.2. Ground Motion

For the case of rolling on the ground or on an horizontal plane, a natural increase in the response times is observed, given the constraints that moving on a surface implies, as discussed in Section 3.1.

As shown in Figure 6, a step in the X-direction is faster than a step on the Y-direction, because the latter implies a yaw rotation, to align the orientation of the wheels with the desired waypoint. Since the yaw dynamics are much slower than in flight mode, due to interaction with the ground, the system takes more time to converge. On the other hand, assuming there is no friction in the wheels joint, the pitch dynamics are as fast as in flight mode, resulting in a similar response time for the longitudinal direction.

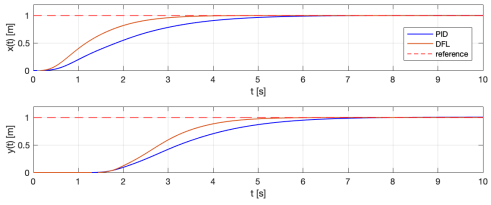


Figure 6: Step response of Ground controllers.

### 5.3. Inclined Surface Motion

Figure 7 compares the response to a step input in slopes with two different inclinations - the first with $\gamma = 0.5$rad, for which the controllers gains are tuned, and the second with $\gamma = 0.8$rad, while using

[1] https://youtu.be/eUILmzxTJdo
[2] https://youtu.be/6COnBPgDS3M
[3] https://youtu.be/yCOuWZ5nCdo
[4] https://youtu.be/VJQzz-BBlDo

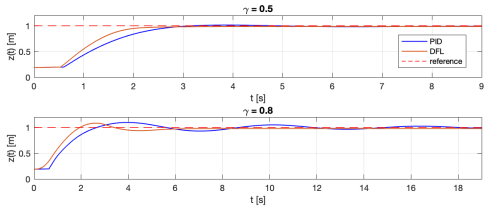


Figure 7: Step response of Inclined controllers.

the gains of the first slope. What stands out from these results is a clear distinction in performance in the two surfaces, with the first presenting a much better transient response, namely when using the tuned PID controller.

Examining the closed-loop characteristics of the response we see the PID controller, although presenting a fairly good result for the first case, underperforms when submitted to a different inclination, taking 18 seconds to fall within 2% of the reference value, presenting a highly oscillatory behaviour and a large overshoot. This indicates a limitation of the linear controller in inclined mode, which will have to be tuned differently for different surfaces, to obtain the best response. Nevertheless, the non-linear controller is capable of complying with different specifications and still present very good results, with a steady and fast response in both situations.

5.4. Hybrid Waypoint Navigation

Together with the specified setpoint, the developed planner also accepts mobility modes - flight, ground or inclined - or it can receive a take-off or landing command, to allow transition between flying and moving on a surface:

- **Take-off Command:** Takes the robot from the ground or from a slope to flight mode, rising vertically.
- **Land Command:** Smoothly descends the robot until the planner detects it has landed on a surface, as depicted in algorithm 1.

When the robot is landing, the planner enters in an idle state until it detects a spike in the accelerometer values. Afterwards, for a small time duration, if the vehicle vertical velocity is bigger than a threshold, the planner predicts that the robot has landed on a slope, estimates the inclination angle (using the longitudinal and vertical velocity values) and activates the inclined controller automatically. Otherwise, the ground controller is switched on.

Experiments were also performed to test complete missions consisting of waypoints in all modes of operation, which showed the success of the system in navigating and switching between modes. An example from a trial can be seen in Figure 8, where the robot starts in the world frame origin

**Algorithm 1:** Landing function

```
Landed = False
Time landed = 0
while Mode == Land do
    Thrust = 0.95mg
    if Landed == True then
        while Time landed < Time treshold do
            if |v_x| > 0 and v_z < 0 then
                Gamma = atan2(v_z, vx)
                Mode = Inclined
                break
            end
            Time landed = Time landed + Δt
        end
        Mode = Ground
    end
    if a > a_min then
        Landed = True
    else
        Landed = False
    end
end
```

**Takes-off** and **Flies** to (2.5, 0, 3), then **Lands**, detecting a flat surface, moves in **Ground Mode** to (4, 0, 2) and climbs the **Inclined Surface** to $z = 3$m. Afterwards, it **Takes-off** again and **Flies** sideways to (6, 5, 4), lands again in the slope, turning on the inclined controller automatically. Finally, it descends in **Inclined Mode**, rolls to (-6, 2, 0) and returns to the origin in **Ground Mode**.

## 6. Conclusions

A Bimodal aerial robot capable of aerial and surface locomotion was described, allowing rapid micro-level inspection of flat and inclined surfaces. Consisting of a common quadrotor attached to two passive wheels, the system results in a mechanically simple and efficient solution that exploits the same actuation mechanism for all modes of operation. The mathematical models and two types of controllers were derived and tested for different locomotion modes and their performance was compared against each other. Results demonstrated the success of the systems in controlling the motion of the robot in different inclinations and full hybrid missions consisting of transitioning between modes of operations were successfully tested in simulation. Implementing the controllers on the developed prototype of Figure 1 and developing and testing surface inspection methods are some of the areas of work to pursue in the future.

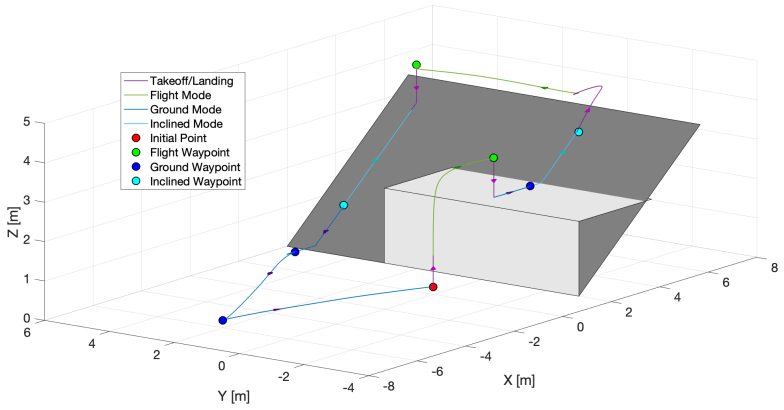


Figure 8: Waypoint Navigation test using the DFL Controller in all 3 modes.

## References


[1] Y. Zefri, A. Elkcttani, I. Sebari, and S. A. Lamallam, "Inspection of Photovoltaic Installations by Thermo-visual UAV Imagery Application Case: Morocco," in *2017 International Renewable and Sustainable Energy Conference (IRSEC)*. Tangier: IEEE, Dec. 2017, pp. 1–6.

[2] M. Burri, J. Nikolic, C. Hurzeler, G. Caprari, and R. Siegwart, "Aerial service robots for visual inspection of thermal power plant boiler systems," in *2012 2nd International Conference on Applied Robotics for the Power Industry (CARPI)*. Zurich: IEEE, Sep. 2012, pp. 70–75.

[3] H. Kayan, R. Eslampanah, F. Yeganli, and M. Askar, "Heat leakage detection and surveiallance using aerial thermography drone," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*. Izmir, Turkey: IEEE, May 2018, pp. 1–4.

[4] A. Ortiz, F. Bonnin-Pascual, and E. Garcia-Fidalgo, "Vessel Inspection: A Micro-Aerial Vehicle-based Approach," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 1, pp. 151–167, Sep. 2014.

[5] S. Jordan, J. Moore, S. Hovet, J. Box, J. Perry, K. Kirsche, D. Lewis, and Z. T. H. Tse, "State of the art technologies for UAV inspections," *IET Radar, Sonar & Navigation*, vol. 12, pp. 151–164, Feb. 2018.

[6] V. Kumar and N. Michael, "Opportunities and challenges with autonomous micro aerial vehicles," *The International Journal of Robotics Research*, vol. 31, no. 11, pp. 1279–1291, Sep. 2012.

[7] N. Takahashi, S. Yamashita, Y. Sato, Y. Kutsuna, and M. Yamada, "All-round two wheeled quadrotor helicopters with protect frames for air land sea vehicle (controller design and automatic charging equipment)," *Advanced Robotics*, vol. 29, no. 1, pp. 69–87, Jan. 2015.

[8] M. Ootsuka, C. Premachandra, and K. Kato, "Development of an air-ground operational robot and its fundamental controlling approach," in *2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*. Kita-Kyushu, Japan: IEEE, Dec. 2014, pp. 1470–1474.

[9] C. Premachandra, M. Otsuka, R. Gohara, T. Ninomiya, and K. Kato, "A study on development of a hybrid aerial terrestrial robot system for avoiding ground obstacles by flight," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 1, pp. 327–336, Jan. 2019.

[10] A. Kalantari and M. Spenko, "Modeling and Performance Assessment of the HyTAQ, a Hybrid Terrestrial/Aerial Quadrotor," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1278–1285, Oct. 2014.

[11] M. T. Pope, C. W. Kimes, H. Jiang, E. W. Hawkes, M. A. Estrada, C. F. Kerst, W. R. T. Roderick, A. K. Han, D. L. Christensen, and M. R. Cutkosky, "A Multimodal Robot for Perching and Climbing on Vertical Outdoor Surfaces," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 38–48, Feb. 2017.

[12] H. Tsukagoshi, M. Watanabe, T. Hamada, D. Ashlih, and R. Iizuka, "Aerial manipulator with perching and door-opening capability," in *2015 IEEE International Conference*

*on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 4663–4668.

[13] A. Kalantari, K. Mahajan, D. Ruffatto, and M. Spenko, "Autonomous perching and take-off on vertical walls for a quadrotor micro air vehicle," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 4669–4674.

[14] W. Myeong and H. Myung, "Development of a Wall-Climbing Drone Capable of Vertical Soft Landing Using a Tilt-Rotor Mechanism," *IEEE Access*, vol. 7, pp. 4868–4879, 2019.

[15] C. E. Doyle, J. J. Bird, T. A. Isom, J. C. Kallman, D. F. Bareiss, D. J. Dunlop, R. J. King, J. J. Abbott, and M. A. Minor, "An Avian-Inspired Passive Mechanism for Quadrotor Perching," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 2, pp. 506–517, Apr. 2013.

[16] A. Kossett, J. Purvey, and N. Papanikolopoulos, "More than meets the eye: A hybrid-locomotion robot with rotary flight and wheel modes," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. St. Louis, MO, USA: IEEE, Oct. 2009, pp. 5653–5658.

[17] A. Kossett, R. D'Sa, J. Purvey, and N. Papanikolopoulos, "Design of an improved land/air miniature robot," in *2010 IEEE International Conference on Robotics and Automation*. Anchorage, AK: IEEE, May 2010, pp. 632–637.

[18] A. Kossett and N. Papanikolopoulos, "A robust miniature robot design for land/air hybrid locomotion," in *2011 IEEE International Conference on Robotics and Automation*. Shanghai, China: IEEE, May 2011, pp. 4595–4600.

[19] L. Daler, J. Lecoeur, P. B. Hahlen, and D. Floreano, "A flying robot with adaptive morphology for multi-modal locomotion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Tokyo: IEEE, Nov. 2013, pp. 1361–1366.

[20] C. J. Pratt and K. K. Leang, "Dynamic underactuated flying-walking (DUCK) robot," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. Stockholm, Sweden: IEEE, May 2016, pp. 3267–3274.

[21] J. R. Page and P. E. I. Pounds, "The Quadroller: Modeling of a UAV/UGV hybrid quadrotor," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Chicago, IL, USA: IEEE, Sep. 2014, pp. 4834–4841.

[22] "Team CoSTAR Nebula Autonomy Solution," [Online; Accessed: 18 November 2020]. [Online]. Available: https://costar.jpl.nasa.gov

[23] A. Kalantari, T. Touma, L. Kim, R. Jitosho, K. Strickland, B. T. Lopez, and A.-A. Agha-Mohammadi, "Drivocopter: A concept Hybrid Aerial/Ground vehicle for long-endurance mobility," in *2020 IEEE Aerospace Conference*. Big Sky, MT, USA: IEEE, Mar. 2020, pp. 1–10.

[24] D. D. Fan, R. Thakker, T. Bartlett, M. B. Miled, L. Kim, E. Theodorou, and A.-a. Agha-mohammadi, "Autonomous Hybrid Ground/Aerial Mobility in Unknown Environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Macau, China: IEEE, Nov. 2019, pp. 3070–3077.

[25] M.-D. Hua, T. Hamel, P. Morin, and C. Samson, "Introduction to Feedback Control of Underactuated VTOL Vehicles," *IEEE Control Systems Magazine*, vol. 33, no. 1, pp. 61–75, Feb. 2013.

[26] R. Mahony, V. Kumar, and P. Corke, "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, Sep. 2012.

[27] Sabatino, Francesco, "Quadrotor Control: modeling, nonlinear control design and simulation," Master's thesis, KTH Electrical Engineering, Sweden, 2015.

[28] A. D. E. Luca and M. D. D. Benedetto, "CONTROL OF NONHOLONOMIC SYSTEMS VIA DYNAMIC COMPENSATION," *Kybernetika*, vol. 29, no. 6, pp. 593–608, 1993.

[29] A. Reyes-La and S. Skogestad, "Multiple-Input Single-Output Control for Extending the Steady-State Operating RangeUse of Controllers with Different Setpoints," *Processes*, vol. 7, no. 12, p. 941, Dec. 2019.

[30] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Cham: Springer International Publishing, 2016, ch. RotorS—A Modular Gazebo MAV Simulator Framework, pp. 595–625.