

Backstepping Controller Applied to a Foldable Quadrotor for 3D Trajectory Tracking

Saddam Hocine Derrouaoui, Yasser Bouzid, Mohamed Guiatni, Halfaoui Kada, Islam Dib and Nouredine Moudjari

Complex Systems Control & Simulators Laboratory, Ecole Militaire Polytechnique, Algiers, Algeria

Keywords: Foldable Quadrotor, Robot, Design, Rotating Arms, Variable Geometry, Adaptive Morphology, Generic Modeling, Control, Backstepping, PID.

Abstract: This paper presents a novel design and architecture control of a foldable quadrotor. This design is based on a variable geometry that can be changed during the flight. It is able to modify the orientation of its arms independently, thanks to its special morphology. This quadrotor, exploits simple mechanisms i.e. rotating arms. We stress that the control of this category of robots is not obvious compared to the conventional ones. So, a detailed generic model and backstepping control that take into account the variation of the center of gravity and the inertia are presented. Simulation results are also provided in order to illustrate the performances of this controller.

1 INTRODUCTION

1.1 Literature Overview

Foldable quadrotors have gained extensive popularity among scientists in recent years, thanks to their capacities to adapt quickly their morphologies to different flight conditions. They can perform critical tasks, difficult to perform with conventional quadrotors such as, negotiation of vertical and horizontal narrow gaps, flying in places more dangerous and difficult to access, the inspection of sensitive points, discovery of the caves and the transport of objects (Mintchev and Floreano, 2016), (Floreano and Wood, 2015).

In (Ryll et al., 2015), (Nemati, 2016), (Kamel et al., 2018), the authors concentrated their studies on morphing aerial vehicles in which the propellers and rotors tilt around their axes in order to overcome the limitation of their mobility, improve their stability and their control in flight. In works (Desbiez et al., 2017), (Sanket et al., 2018), transformable quadrotors in flight were studied in order to negotiate small openings or to transport objects. In paper (Riviere et al., 2018), a quadrotor that can fold its structure through a vertical or tilted window is proposed. The PID controller was implemented to control the position of the aerial robot. In article (Brescianini and D'Andrea, 2016), the authors presented an omnidirectional configuration allowing the quadrotor to ex-

plot the dynamics of translation and rotation separately. It is based on reversible actuators, which have proven the feasibility of the design approach and the ability of the vehicle to generate thrust and torque independently and in different directions. A multi-link aerial robot has been presented in (Zhao et al., 2018) where the joints are actuated slowly. A quadrotor, with two independent rotating arms, is designed in (Desbiez et al., 2017). A linear adaptive control is implemented to deal with the changes of the inertia and the center of mass. Bio-inspired design of quadrotors has been investigated in reference (Mintchev et al., 2018). This vehicle can deform during a collision with obstacles to avoid damage by the use of specific structural materials. Another design was presented by (Avant et al., 2018) considering the rotation and extension of the quadrotor arms dependently, which hold in the event of a rotor failure in order to ensure stable flight with three rotors. The PD and LQR controllers were tested in simulation. Considering the rotation of the quadrotor arms, (Xiong et al., 2019) are interested in optimizing energy consumption during the flight, which the attitude and the trajectory of the prototype were controlled using two PID controllers. The same principle is adopted in (Mintchev et al., 2015), (Falanga et al., 2018) to focus later on special configurations that are requested to cross vertical and horizontal gaps. To ensure a stable flight, an adaptive Linear Quadratic Regulation was applied in (Falanga

et al., 2018).

1.2 Contribution of This Work

The proposed quadrotor (see Fig. 1), having for particularity the versatility in the execution of the tasks, and especially those, which are dangerous and difficult such as, the mapping, the cross of small openings, the navigation in congested spaces, the transport objects, the inspection of cluttered environments and for minimizing the consumed energy. This quadrotor, can be used also to flight with three rotors when it has an anomaly in the one of the four rotors, which it will be able to change quickly its configuration and continuing the assigned mission.

We opted in the design on a simple mechanism of rotation of the arms, which makes our quadrotor less heavy and not complicate. The modeling and the control of this quadrotor is more challenging. Herein, we seek to present a detailed generic model and backstepping control strategy taking into account the dynamics, the variation of the center of gravity, the variation of the inertia, the variation of the allocation matrix and aerodynamic effects during the flight.

2 MECHANICAL DESIGN AND GEOMETRIC DESCRIPTION

As shown in Figures 1 and 2, our quadrotor is different to the classical platform. It can fold its arms around a central body using very powerful and fast servomotors, which allow a quick passage between the various configurations.



Figure 1: Design of our proposed quadrotor, able to change its configuration while flying. Each rotor is connected to a rotating arm. Each rotating arm is connected to a servomotor and it can rotate independently.

The chosen sequence of rotation is: the $X Y Z$ sequence, meaning that the attitude is obtained first by the roll angle φ , then by the pitch angle θ and then by the yaw angle ψ . The obtained rotation matrix R is given by:

$$R = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\varphi - S_\psi C_\varphi & C_\psi S_\theta C_\varphi + S_\psi S_\varphi \\ S_\psi C_\theta & S_\psi S_\theta S_\varphi + C_\psi C_\varphi & S_\psi S_\theta C_\varphi - C_\psi S_\varphi \\ -S_\theta & C_\theta S_\varphi & C_\theta C_\varphi \end{bmatrix} \quad (1)$$

where $R \in SO(3) = \{R \in \mathbb{R}^{3 \times 3} | R^T R = I_{3 \times 3}, \det(R) = 1\}$. $I_{3 \times 3}$ is the identity matrix of dimension 3×3 . $S_{(\cdot)}$ and $C_{(\cdot)}$ are abbreviations for $\text{Sin}(\cdot)$ and $\text{Cos}(\cdot)$ respectively.

The designed quadrotor is composed of: a central body of mass m_0 , four rotation servomotors attached to the central body of mass $m_{1,i}$, four rotating arms attached to servomotors of mass $m_{2,i}$ and four rotor of mass $m_{3,i}$. Each component has its own body frame.

The four rotors are numbered 1, 2, 3 and 4, and each arm is numbered according to the rotor fixed on it. This quadrotor has two pairs of rotors, where two rotors (2.4) spin in the clockwise direction whilst the other two spin anticlockwise direction (1.3) (see Fig. 2). Each pair of rotors are placed on two opposite sides of the airframe. We stress that the axes of rotation of the rotors and the rotating arms are parallel to z_b axes (see Fig. 2). By changing the speed of the rotors or the angles of the four arms, the quadrotor will produce different motions (roll, pitch, yaw, altitude). The throttle input M_T is the sum of the thrusts generated by the four rotors. The direction of the thrust produced by each rotor does not depend on the variation of the geometry. The moment around the body z_b axis (M_z) does not depend on the configuration, and its expression is the same with standard quadrotor. The main configuration leads to many other possible configurations by changing either the angle of rotation of the arms $\psi_i(t)$.

3 GENERIC MODELING

To model the foldable quadrotor, we based on multi-body modeling approach taking into account all the phenomena, because our system changes its morphology while the flight according to the assigned task.

Since the majority of the configurations are not symmetrical, so we will take into consideration that the Center of Gravity (CoG) of the global system moves, the global inertia matrix of the system varies, the rotation of the arms also changes, which will push us to develop a generic model.

The relation between the velocities and the external forces and moments applied to center of gravity, expressed in body frame, may be written using Newton-Euler formalism as:

$$\begin{cases} m \ddot{\xi} = F_T + F_D + F_G \\ I_{\text{Sys}/O}(\psi_i(t)) \dot{\Omega} = -\Omega \wedge I_{\text{Sys}/O}(\psi_i(t)) \Omega + M_f - M_a - M_{gy} \end{cases} \quad (2)$$

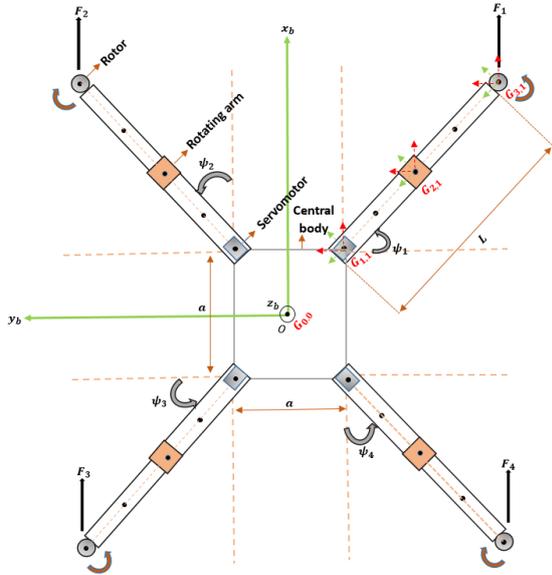


Figure 2: Schematic of our quadrotor.

where

ξ is the position of the quadrotor with respect to the inertial frame.

m is the total mass of quadrotor structure.

Ω is the angular velocity of the airframe expressed in b .

$I_{Sys/O}(\psi_i(t))$ is the quadrotor inertia during the folding process, is calculated with respect to the geometric center O of the body frame b and depends on the variation of the rotation angle $\psi_i(t)$ of the arms.

F_T is the resultant of the forces generated by the four rotors:

$$F_T = R \cdot \begin{bmatrix} 0 & 0 & \sum_{i=1}^4 F_i \end{bmatrix}^T \quad (3)$$

$$F_i = K_l \omega_i^2$$

where K_l is the lift coefficient and ω_i is the angular rotor speed.

F_D is the resultant of the drag forces along (X, Y, Z) axes:

$$F_D = \text{diag}(-K_{Dx} \quad -K_{Dy} \quad -K_{Dz}) \dot{\xi} \quad (4)$$

where K_{Dx} , K_{Dy} , K_{Dz} are the translation drag coefficients, respectively.

F_G is the force of gravity:

$$F_G = \begin{bmatrix} 0 & 0 & -mg \end{bmatrix}^T \quad (5)$$

where g is the gravity.

M_f is the moment caused by the thrust and drag forces. It is expressed as follows:

$$M_f = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \quad (6)$$

M_a is the resultant of aerodynamic friction torques:

$$M_a = \text{diag}(K_{ax} \quad K_{ay} \quad K_{az}) \Omega^2 \quad (7)$$

where K_{ax} , K_{ay} and K_{az} are the aerodynamic friction coefficients.

M_{gy} is the resultant of torques due to the gyroscopic effects:

$$M_{gy} = \sum_{i=1}^4 \Omega \wedge J_r \begin{bmatrix} 0 & 0 & (-1)^{i+1} w_i \end{bmatrix}^T \quad (8)$$

where J_r is the rotor inertia.

3.1 Calculation of the Center of Gravity

The configurations of the quadrotor are not symmetrical. Thus, the CoG varies and has to be recomputed when the configuration is changed. It is necessary to put the general formula to calculate the CoG instantaneously, which depends on the masses and the variation of the centers of gravity of the other components that we have mentioned before (see Fig. 2).

The formula is given as follows:

$$\vec{OG} = \frac{\sum_{i=1}^4 \sum_{j=1}^3 m_{j,i} \vec{OG}_{j,i}}{m_0 + \sum_{i=1}^4 \sum_{j=1}^3 m_{j,i}} \quad (9)$$

with

$$\vec{OG} = \begin{bmatrix} x_G \\ y_G \\ z_G \end{bmatrix}, \vec{OG}_0 = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}, \vec{OG}_{1,i} = \begin{bmatrix} x_{1,i} \\ y_{1,i} \\ z_{1,i} \end{bmatrix} \quad (10)$$

$$\vec{OG}_{2,i} = \begin{bmatrix} x_{2,i} \\ y_{2,i} \\ z_{2,i} \end{bmatrix}, \vec{OG}_{3,i} = \begin{bmatrix} x_{3,i} \\ y_{3,i} \\ z_{3,i} \end{bmatrix} \quad (11)$$

and

O : the geometric center (the origin of the body frame).

$\vec{OG} \in \mathbb{R}^{3 \times 1}$: the offset between the geometric center of the system and the global center of gravity.

$\vec{OG}_0 \in \mathbb{R}^{3 \times 1}$: the vector of the center of gravity of the central body which is zero.

$\vec{OG}_{1,i} \in \mathbb{R}^{3 \times 1}$: the vector of the center of gravity of the servomotors.

$\vec{OG}_{2,i} \in \mathbb{R}^{3 \times 1}$: the vector of the center of gravity of the rotating arms.

$\vec{OG}_{3,i} \in \mathbb{R}^{3 \times 1}$: the vector of the center of gravity of the rotors.

3.2 Calculation of the Inertia

The calculation of inertia for any structure is not obvious. Therefore, some approximations should be set in order to simplify the calculations. We assume that the central body is a box of length and width a and height h_0 . The servomotors are assumed rectangular cuboid of length L_1 , width w_1 and height h_1 . The rotating arms are also supposed rectangular cuboid of length L , width w_2 and height h_2 . The rotors are supposed cylinders of radius r and height h_3 .

We use the green body frame of each element (see Fig. 2). We suppose that all the centers of gravity are in the plan known as (O, x_b, y_b) . The global inertia matrix $I_{Sys/O}(\Psi_i(t))$ depends to the angle of rotation of each arm.

Using the parallel axis theorem, the calculation of the inertia matrix is given as follows:

Inertia matrix of the central body:

$$I_{(0)/O} = m_0 \text{diag} \left(\frac{a^2 + h_0^2}{12}, \frac{a^2 + h_0^2}{12}, \frac{2a^2}{12} \right)_{G_0} \quad (12)$$

Inertia matrix of the servomotors:

$$I_{(1,i)/O} = I_{G_{1,i}} + m_{(1,i)} \begin{bmatrix} y_{1,i}^2 & -x_{1,i}y_{1,i} & 0 \\ -x_{1,i}y_{1,i} & x_{1,i}^2 & 0 \\ 0 & 0 & x_{1,i}^2 + y_{1,i}^2 \end{bmatrix} \quad (13)$$

where

$$I_{G_{1,i}} = m_{(1,i)} \text{diag} \left(\frac{L_1^2 + h_1^2}{12}, \frac{w_1^2 + h_1^2}{12}, \frac{L_1^2 + w_1^2}{12} \right)_{G_{1,i}} \quad (14)$$

Inertia matrix of the rotating arms:

$$I_{(2,i)/O_{(Rot)}} = R_z(\Psi_i(t)) I_{(2,i)/O} R_z(\Psi_i(t))^T \quad (15)$$

where

$$I_{(2,i)/O} = I_{G_{2,i}} + m_{(2,i)} \begin{bmatrix} y_{2,i}^2 & -x_{2,i}y_{2,i} & 0 \\ -x_{2,i}y_{2,i} & x_{2,i}^2 & 0 \\ 0 & 0 & x_{2,i}^2 + y_{2,i}^2 \end{bmatrix} \quad (16)$$

$$I_{G_{2,i}} = m_{(2,i)} \text{diag} \left(\frac{L^2 + h_2^2}{12}, \frac{w_2^2 + h_2^2}{12}, \frac{L^2 + w_2^2}{12} \right)_{G_{2,i}} \quad (17)$$

and $R_z(\Psi_i(t)) \in \mathbb{R}^{3 \times 3}$ is the rotation matrix on the z_b axis.

Inertia matrix of the rotors:

$$I_{(3,i)/O_{(Rot)}} = R_z(\Psi_i(t)) I_{(3,i)/O} R_z(\Psi_i(t))^T \quad (18)$$

where

$$I_{(3,i)/O} = I_{G_{3,i}} + m_{(3,i)} \begin{bmatrix} y_{3,i}^2 & -x_{3,i}y_{3,i} & 0 \\ -x_{3,i}y_{3,i} & x_{3,i}^2 & 0 \\ 0 & 0 & x_{3,i}^2 + y_{3,i}^2 \end{bmatrix} \quad (19)$$

$$I_{G_{3,i}} = m_{(3,i)} \text{diag} \left(\frac{r^2}{4} + \frac{h_3^2}{12}, \frac{r^2}{4} + \frac{h_3^2}{12}, \frac{r^2}{2} \right)_{G_{3,i}} \quad (20)$$

At the end, we get the global inertia matrix of the system given by:

$$I_{Sys/O}(\Psi_i(t)) = I_{(0)/O} + \sum_{i=1}^4 I_{(1,i)/O} + \sum_{i=1}^4 I_{(2,i)/O_{(Rot)}} + \sum_{i=1}^4 I_{(3,i)/O_{(Rot)}} \quad (21)$$

3.3 Allocation Matrix

Roll and pitch moments $\vec{M}_{x,y}$ are found from the cross product. They depend on the variation of the rotation angle $\Psi_i(t)$ of each arm.

$$\vec{M}_{x,y} = \sum_{i=1}^4 \vec{G}_{(3,i)} \wedge F_{i\vec{e}_z} \quad (22)$$

The relation between the thrust force total, the moments applied to CoG and velocities is given by:

$$\begin{bmatrix} M_T \\ M_x \\ M_y \\ M_z \end{bmatrix} = A \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (23)$$

where

$$A = \begin{bmatrix} K_l & K_l(y_{3,1} - y_G) & K_l(-x_{3,1} + x_G) & K_d \\ K_l & K_l(y_{3,2} - y_G) & K_l(-x_{3,2} + x_G) & -K_d \\ K_l & K_l(y_{3,3} - y_G) & K_l(-x_{3,3} + x_G) & K_d \\ K_l & K_l(y_{3,4} - y_G) & K_l(-x_{3,4} + x_G) & -K_d \end{bmatrix}^T \quad (24)$$

and

$$\begin{cases} x_{3,1} = \frac{a}{2} + (L+r) \sin \Psi_1(t) \\ x_{3,2} = \frac{a}{2} + (L+r) \cos \Psi_2(t) \\ x_{3,3} = -\frac{a}{2} - (L+r) \sin \Psi_3(t) \\ x_{3,4} = -\frac{a}{2} - (L+r) \cos \Psi_4(t) \\ y_{3,1} = -\frac{a}{2} - (L+r) \cos \Psi_1(t) \\ y_{3,2} = \frac{a}{2} + (L+r) \sin \Psi_2(t) \\ y_{3,3} = \frac{a}{2} + (L+r) \cos \Psi_3(t) \\ y_{3,4} = -\frac{a}{2} - (L+r) \sin \Psi_4(t) \end{cases} \quad (25)$$

M_T is the thrust force total, M_z is the yaw moment and K_d is the drag coefficient.

4 CONTROL OF QUADROTOR

4.1 Control Architecture

In the literature, several recent works have used the PID and LQR controllers (Desbiefz et al., 2017), (Riviere et al., 2018), (Avant et al., 2018), (Xiong et al., 2019), (Falanga et al., 2018), due to the complexity and the difficulty to control this type of quadrotors.

In this paper, our backstepping controller is designed to ensure the tracking of the desired trajectory (x_d, y_d, z_d) along the three axes and the ψ_d angle. These reference trajectories are provided on-line by a Trajectory Generator Block (TGB). The quadrotor is controlled by the speeds of the four motors, these speeds are deduced from the allocation matrix. The CoG, the inertia matrix and the allocation matrix varie and have to be recomputed on-line when the configuration is changed.

The appropriate gains for each configuration of the Backstepping controller are identified separately. Then, we checked that the variation of the gains for the intermediate configurations is not significant and does not influence the desired dynamics. Consequently, the gains change as soon as the servomotors reach half of the desired positions. After identification the gains, they will be injected into the different controllers according to the current configuration.

The four servomotors are controlled independently of the system by the PID controller (U_5, U_6, U_7, U_8), their postions ($\psi_1(t), \psi_2(t), \psi_3(t), \psi_4(t)$) are sent to the control blocks and the allocation matrix bloc to take them into account. The control structure scheme is shown in Figure 3. In our work, we opted for six different configurations of the quadrotor, which changes them every 10 (sec) (see Table 1).

Table 1: Different configurations of our quadrotor.

	Servomotors angles
Configuration 1	$\psi_1=\pi/4, \psi_2=\pi/4, \psi_3=\pi/4, \psi_4=\pi/4$
Configuration 2	$\psi_1=\pi/2, \psi_2=0, \psi_3=\pi/2, \psi_4=0$
Configuration 3	$\psi_1=\pi, \psi_2=\pi, \psi_3=\pi, \psi_4=\pi$
Configuration 4	$\psi_1=\pi/4, \psi_2=\pi/4, \psi_3=\pi/2, \psi_4=0$
Configuration 5	$\psi_1=\pi/2, \psi_2=0, \psi_3=\pi/4, \psi_4=\pi/4$
Configuration 6	$\psi_1=0, \psi_2=\pi/2, \psi_3=\pi/2, \psi_4=0$

4.2 Controller Design

In this part, we investigate the efficiency of the proposed controller through and its application on our reconfigurable quadrotor, taking into account the variation of CoG, the inertia and the allocation matrix. The controller is designed considering the model shown in

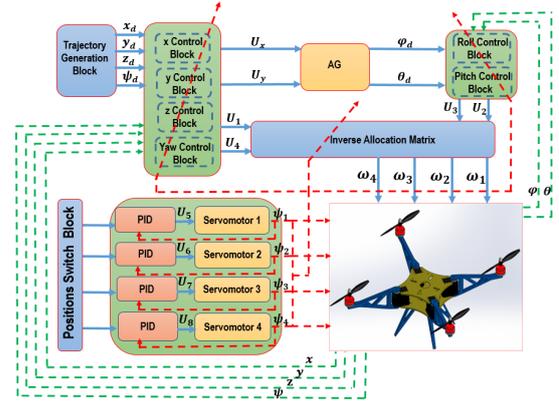


Figure 3: Control architecture.

(28) and applied to the generic model that we developed before in section 3.

The choice of the state vector is as follows:

$$X = [\varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}, z, \dot{z}, x, \dot{x}, y, \dot{y}]^T \quad (26)$$

such as:

$$X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}]^T \quad (27)$$

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = a_1 x_4 x_6 + a_2 x_4 \Omega + b_1 U_2 + c_1 x_2^2 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = a_3 x_2 x_6 + a_4 x_2 \Omega + b_2 U_3 + c_2 x_4^2 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = a_5 x_2 x_4 + b_3 U_4 + c_3 x_6^2 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = -g + U_1 \frac{\cos(x_1) \cos(x_3)}{m} + c_4 x_8 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = U_1 \frac{U_x}{m} + c_5 x_{10} \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = U_1 \frac{U_y}{m} + c_6 x_{12} \end{cases} \quad (28)$$

where

$$\begin{aligned} a_1 &= \frac{I_{yy}(\psi_i(t)) - I_{zz}(\psi_i(t))}{I_{xx}(\psi_i(t))}, \quad a_2 = \frac{-J_r}{I_{xx}(\psi_i(t))}, \quad a_3 = \frac{I_{zz}(\psi_i(t)) - I_{xx}(\psi_i(t))}{I_{yy}(\psi_i(t))} \\ a_4 &= \frac{J_r}{I_{yy}(\psi_i(t))}, \quad a_5 = \frac{I_{xx}(\psi_i(t)) - I_{yy}(\psi_i(t))}{I_{zz}(\psi_i(t))}, \quad b_1 = \frac{1}{I_{xx}(\psi_i(t))}, \quad b_2 = \frac{1}{I_{yy}(\psi_i(t))} \\ b_3 &= \frac{1}{I_{zz}(\psi_i(t))}, \quad c_1 = \frac{-K_{ax}}{I_{xx}(\psi_i(t))}, \quad c_2 = \frac{-K_{ay}}{I_{yy}(\psi_i(t))}, \quad c_3 = \frac{-K_{az}}{I_{zz}(\psi_i(t))}, \quad c_4 = \frac{-K_{Dz}}{m} \\ c_5 &= \frac{-K_{Dx}}{m}, \quad c_6 = \frac{-K_{Dy}}{m}, \quad \Omega = \omega_1 - \omega_2 + \omega_3 - \omega_4 \end{aligned}$$

The basic controller design procedure in our case is performed in two steps.

Firstly:

The tracking error of the first subsystem is given:

$$e_1 = x_{1d} - x_1 \quad (29)$$

Its dynamics is described as follows:

$$\dot{e}_1 = \dot{x}_{1d} - x_2 \quad (30)$$

Let V_1 be a candidate positive definite Lyapunov function of the first subsystem and its derivative \dot{V}_1 is negative definite:

$$V_1 = \frac{1}{2}e_1^2 \quad (31)$$

$$\dot{V}_1 = -K_1 e_1^2, K_1 > 0 \quad (32)$$

Now, the asymptotic Lyapunov stability is guaranteed. For that, the virtual control law of the first subsystem could be chosen as:

$$x_{2d} = \dot{x}_{1d} + K_1 e_1 \quad (33)$$

where e_2 is the tracking error of the second subsystem given by:

$$e_2 = x_{2d} - x_2 \quad (34)$$

Secondly:

Let us augment the Lyapunov function expressed by (31) as follows:

$$V_2 = V_1 + \frac{1}{2}e_2^2 \quad (35)$$

where V_2 is positive definite. Its derivative:

$$\dot{V}_2 = -K_1 e_1^2 + e_2 [\ddot{x}_{1d} + K_1 \dot{e}_1 - a_1 x_4 x_6 - a_2 x_4 \Omega - b_1 U_2 - c_1 x_2^2] \quad (36)$$

with \dot{V}_2 must be negative definite. To satisfy this condition and ensure the Lyapunov stability, we choose the control law U_2 as follows:

$$U_2 = \frac{1}{b_1} [\ddot{x}_{1d} + K_1 \dot{e}_1 - a_1 x_4 x_6 - a_2 x_4 \Omega + K_2 e_2 - c_1 x_2^2] \quad (37)$$

The next control laws U_1, U_3, U_4, U_x and U_y , are obtained by following the same steps presented above, therefore:

$$\left\{ \begin{array}{l} U_1 = \frac{m}{\cos(x_1)\cos(x_3)} [g + \ddot{x}_{7d} + K_7 \dot{e}_7 + K_8 e_8 - c_4 x_8] \\ U_2 = \frac{1}{b_1} [\ddot{x}_{1d} + K_1 \dot{e}_1 - a_1 x_4 x_6 - a_2 x_4 \Omega + K_2 e_2 - c_1 x_2^2] \\ U_3 = \frac{1}{b_2} [\ddot{x}_{3d} + K_3 \dot{e}_3 - a_3 x_2 x_6 - a_4 x_2 \Omega + K_4 e_4 - c_2 x_4^2] \\ U_4 = \frac{1}{b_3} [\ddot{x}_{5d} + K_5 \dot{e}_5 - a_5 x_2 x_4 + K_6 e_6 - c_3 x_6^2] \\ U_x = \frac{m}{U_1} [\ddot{x}_{9d} + K_9 \dot{e}_9 + K_{10} e_{10} - c_5 x_{10}] \\ U_y = \frac{m}{U_1} [\ddot{x}_{11d} + K_{11} \dot{e}_{11} + K_{12} e_{12} - c_6 x_{12}] \end{array} \right. \quad (38)$$

where $K_2, K_3, K_4, K_5, K_6, K_7, K_8, K_9, K_{10}, K_{11}, K_{12}$ are positive constants.

The servomotor controllers are given as follows:

$$\left\{ \begin{array}{l} U_5 = K_{P5} e_5 + K_{I5} \int_0^t e_5 dt + K_{D5} \dot{e}_5 \\ U_6 = K_{P6} e_6 + K_{I6} \int_0^t e_6 dt + K_{D6} \dot{e}_6 \\ U_7 = K_{P7} e_7 + K_{I7} \int_0^t e_7 dt + K_{D7} \dot{e}_7 \\ U_8 = K_{P8} e_8 + K_{I8} \int_0^t e_8 dt + K_{D8} \dot{e}_8 \end{array} \right. \quad (39)$$

where $e_i = \psi_{id} - \psi_i$ is the tracking error. K_{Pi}, K_{Ii} and K_{Di} denote the usual proportional integral derivative tuning gains. They are positive constants.

where $i=5, \dots, 8$.

4.3 Results and Discussion

We have developed a Matlab simulation environment to evaluate the effectiveness of the proposed control strategy applied to the quadrotor.

The results are displayed in Figures 4-5-6-7. We have plotted separately, the tracking errors, the outputs along x, y, z -axes, the attitude angles φ, θ, ψ , the servomotors outputs $\psi_1(t), \psi_2(t), \psi_3(t), \psi_4(t)$ and the control inputs U_1, U_2, U_3, U_4 .

The effectiveness of this controller is investigated in the variation of the geometry of the quadrotor each 10 (sec) while flying.

From the graphics shown in Figures 4-5, the quadrotor follows its reference trajectory in a good manner and satisfactory accuracy. The overshoot is smallest on the ψ response. Moreover, it is clear that, the backstepping controller ensures the continuity of the control when switching between the different configurations, with less consumed energy (see Fig. 4-6).

From Figure 7, the outputs of the four servomotors follow the desired angles with some errors in the descent and ascent.

Through the simulation results, we note that the Backstepping controller adapts quickly to the variation of the configurations. It has proven its effectiveness taking into account the variation of the CoG, the variation of the inertia and the allocation matrix.

5 PROTOTYPE PLATFORM

In this section, we present the quadrotor platform that we made in our laboratory (see Fig. 8). This prototype is under flight testing. To have a reconfigurable structure, the quadrotor must rotate its arms. The rotation of these arms is ensured by four servomotors of type Dynamixel (see Fig. 8), which are controlled in position and speed at the same time. The body angular velocities are presented as p, q and r , which are physically measured by three gyroscopes sensors integrated in the Pixhawk flight controller. To avoid the collision of the propellers, we have proposed a solution to shift slightly each two adjacent rotors so that the propellers never coincide.

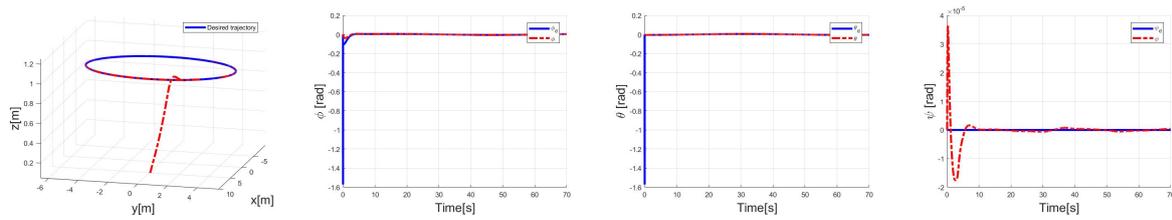


Figure 4: Quadrotor outputs.

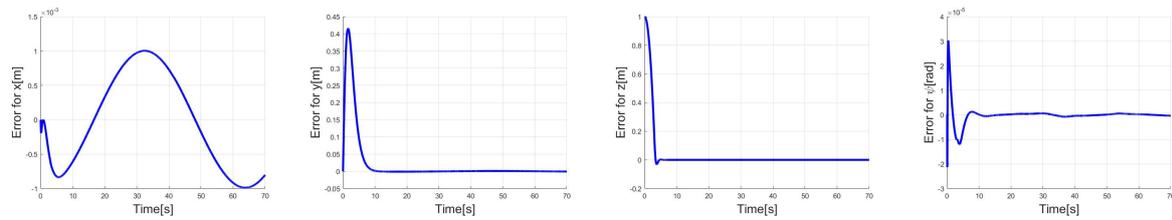


Figure 5: Quadrotor errors.

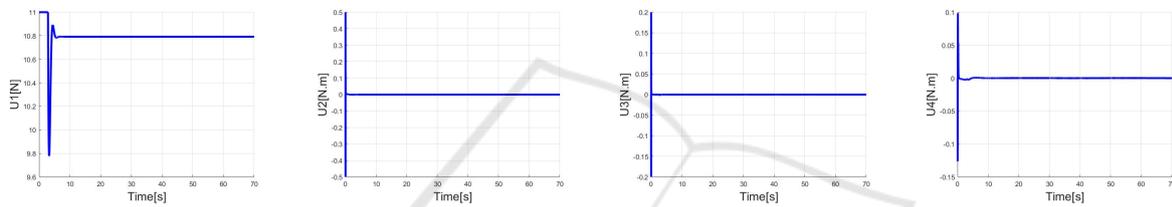


Figure 6: Control inputs.

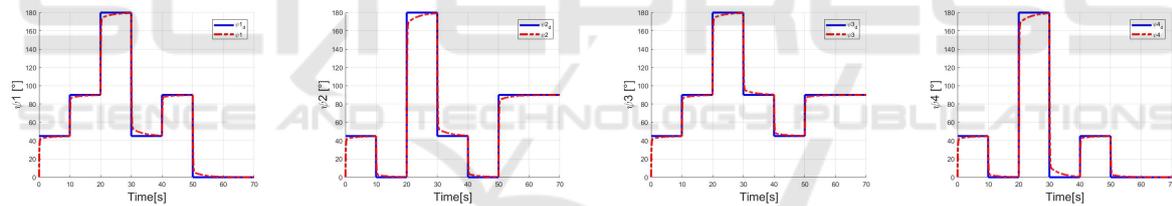


Figure 7: Servomotors outputs.

The realized quadrotor is characterized as follows (see Table 2):

Table 2: Characteristics of our prototype.

Parameters	Value
Global platform weight	1500 g
Arm length	21 cm
Central body length	12 cm
Central body width	12 cm

6 CONCLUSIONS AND FUTURE WORK

We have presented a new architecture control and generic modeling of a foldable quadrotor and its applications. This system has the ability to control the orientation of its four arms independently. A new con-

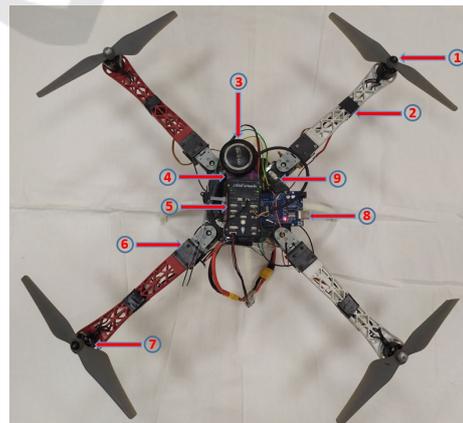


Figure 8: Realized prototype. (1) Propeller. (2) Rotating arm. (3) GPS. (4) Battery. (5) Pixhawk flight controller. (6) Junction servo-arm. (7) Rotor. (8) Arduino card. (9) Servomotor.

control strategy was applied taking into account the calculation of the CoG, the inertia and the allocation matrix instantaneously. The simulations have shown satisfactory results of the Backstepping controller. The flight test and implementation of the control laws are in the final phase.

Adaptive control laws and optimization of the energy consumption will be treated in our future works.

REFERENCES

- Avant, T., Lee, U., Katona, B., and Morgansen, K. (2018). Dynamics, hover configurations, and rotor failure restabilization of a morphing quadrotor. In *2018 Annual American Control Conference (ACC)*, pages 4855–4862.
- Brescianini, D. and D’Andrea, R. (2016). Design, modeling and control of an omni-directional aerial vehicle. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 3261–3266.
- Desbiez, A., Expert, F., Boyron, M., Diperi, J., Viollet, S., and Ruffier, F. (2017). X-morf: A crash-separable quadrotor that morfs its x-geometry in flight. In *2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pages 222–227.
- Falanga, D., Kleber, K., Mintchev, S., Floreano, D., and Scaramuzza, D. (2018). The foldable drone: A morphing quadrotor that can squeeze and fly. *IEEE Robotics and Automation Letters*, 4(2):209–216.
- Floreano, D. and Wood, R. J. (2015). Science, technology and the future of small autonomous drones. *Nature*, 521(7553):460–466.
- Kamel, M., Verling, S., Elkhatib, O., Sprecher, C., Wulkop, P., Taylor, Z., Siegwart, R., and Gilitschenski, I. (2018). The voliro omniorientational hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle. *IEEE Robotics & Automation Magazine*, 25(4):34–44.
- Mintchev, S., Daler, L., L’Eplattenier, G., Saint-Raymond, L., and Floreano, D. (2015). Foldable and self-deployable pocket sized quadrotor. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2190–2195.
- Mintchev, S. and Floreano, D. (2016). Adaptive morphology: A design principle for multimodal and multifunctional robots. *IEEE Robotics & Automation Magazine*, 23(3):42–54.
- Mintchev, S., Shintake, J., and Floreano, D. (2018). Bioinspired dual-stiffness origami. *Science Robotics*, 3(20):eaau0275.
- Nemati, A. (2016). *Designing, modeling and control of a tilting rotor quadcopter*. PhD thesis, University of Cincinnati.
- Riviere, V., Manecy, A., and Viollet, S. (2018). Agile robotic fliers: A morphing-based approach. *Soft Robotics*, 5(5):541–553.
- Ryll, M., Bühlhoff, H. H., and Giordano, P. R. (2015). A novel overactuated quadrotor uav: Modelling, control and experimental validation. *IEEE Transactions on Control Systems Technology, Institute of Electrical and Electronics Engineers*, 23(2):510–556.
- Sanket, N. J., Singh, C. D., Ganguly, K., Fermüller, C., and Aloimonos, Y. (2018). Gapflyt: Active vision based minimalist structure-less gap detection for quadrotor flight. *IEEE Robotics and Automation Letters*, 3(4):2799–2806.
- Xiong, H., Hu, J., and Diao, X. (2019). Optimize energy efficiency of quadrotors via arm rotation. *Journal of Dynamic Systems, Measurement, and Control*, 141(9).
- Zhao, M., Anzai, T., Shi, F., Chen, X., Okada, K., and Inaba, M. (2018). Design, modeling, and control of an aerial robot dragon: A dual-rotor-embedded multi-link robot with the ability of multi-degree-of-freedom aerial transformation. *IEEE Robotics and Automation Letters*, 3(2):1176–1183.