ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/362473490

Modeling of an unmanned aerial vehicle and trajectory tracking control using backstepping approach

Article *in* IFAC-PapersOnLine · January 2022

DOI: 10.1016/j.ifacol.2022.07.324

CITATIONS
9

READS
300

3 authors:

Asmaa Taame
Ecole Nationale Supérieure d'Electricité et de Mécanique de Casablanca
3 PUBLICATIONS 10 CITATIONS
SEE PROFILE

I. Lachkar
Ecole Nationale Supérieure d'Electricité et de Mécanique de Casablanca
110 PUBLICATIONS 712 CITATIONS
SEE PROFILE

Abdelmajid Abouloifa
Université Hassan II Casablanca
145 PUBLICATIONS 1,036 CITATIONS
SEE PROFILE

All content following this page was uploaded by Asmaa Taame on 07 August 2022.

The user has requested enhancement of the downloaded file.

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-12 (2022) 276–281

Modeling of an unmanned aerial vehicle and trajectory tracking control using backstepping approach

A.Taame*, I.Lachkar*, A.Abouloifa*

* ESE Lab, ENSEM of Casablanca, University Hassan II of Casablanca, BP 7955 Casablanca Morocco.
(e-mail:asmaataame@icloud.com)

Abstract: A quadcopter is a type of unmanned under-actuated aerial vehicle (UAV) that has four arms, each connected to a motor (Hussein, 2020). It is an excellent platform for research on control, due to its high non-linearity and the high degree of coupling in its dynamic representation of the model. This paper deals with the modeling of this nonlinear system using the Newton-Euler formalism, combining the dynamics of translation and rotation of the vehicle. The control objective is to track the reference trajectory by managing altitude and attitude at the same time. A control structure based on two cascade loops is developed using the nonlinear Backstepping approach with Lyapunov's stability theory. Simulation results in Matlab/Simulink show that the control objective is met.

Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)

Keywords: Quadcopter, Unmanned aerial vehicle, Newton-Euler method, Backstepping approach, Lyapunov.

1. INTRODUCTION

Recent technological developments, especially those related to sensors, have sparked interest in unmanned aerial vehicles (UAVs). They allowed the development of powerful systems such as mini UAVS capable of traveling with advanced autonomy performances.
Recently drones serve many purposes (Hamel et al., 2002) (Bashi et al., 2016), we can mention some disaster-related missions such as the detection and extinction of fires, forecasting volcano activity, and also risky intervention by providing strategic information in war, facial recognition, license plate recognition, and mining operations.
Despite the significant advances made in the last years, researchers continue to face significant challenges in controlling such systems, particularly in the presence of air turbulence, air friction, and magnetic field disturbance. Furthermore, drones have become a global business. Autonomous navigation has become a challenge that requires a good knowledge of vehicle dynamics and the aerodynamic effects.
Various control strategies have been proposed for quadcopter navigation in (Vibhu et al., 2015) the article suggests sliding mode and backstepping controllers to ensure trajectory tracking. Nevertheless, the effects of aerodynamic and gyroscopic are not taken into consideration. In (Sabir et al., 2018) the authors suggest in the first section a quadcopter dynamic model taking into consideration various constants and parameters that affect the navigation using Newton Euler formalism. In the second section, the paper treats trajectory tracking using a cascaded PID controllers. While the authors of (Fayzan et al., 2013) studied the effectiveness of a simple PID closed-loop and then another loop that contains PID and extended Kalman filter loop the simulations showed that the second loop offers better performances under noisy conditions.
In this paper, we configure of the system under a more full and realistic new state-space representation, that takes into consideration the noisy environment, the aerodynamic and gyroscopic effects. The control objective is to force the UAV to follow a given trajectory.
The system is under-actuated, so to reach all the six degrees of freedom, we go necessary through non-linear decoupling equations. Next, we describe a control strategy under a cascade structure, based on the backstepping approach. Simulation work in the Matlab/Simulink environment shows the effectiveness of the regulator.

2. SYSTEM DESCRIPTION AND MODELLING

2.1 Coordinate Frames

Before developing a mathematical model of the quadcopter, the two-notations $R_g(O,\vec{I},\vec{J},\vec{K})$ and $R_b(o,\vec{i},\vec{j},\vec{k})$need to be proceed as shown in Fig.1. Such that the reference R_g is bound to the ground and the reference R_b is a frame linked to the body of the UAV and its center 'O' corresponds with his mass center.

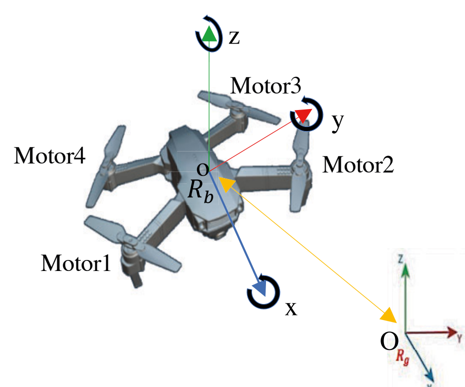

Figure 1. The quadrotor schematic and coordinate systems.

- **x, y** and **z** represents the position of the quadrotor relative to Earth frame expressed in R_g.

2405-8963 Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2022.07.324

- $\boldsymbol{\varphi}$, $\boldsymbol{\theta}$, $\boldsymbol{\psi}$ called Euler angles or roll, pitch and yaw angles, respectively. They represent the orientation of the UAV about its mass center relative to earth frame expressed in R_g.

2.2 Applied forces and torques

To have a quadcopter model closer to the reality, all the aerodynamic effects must be included in the system model. It movement are governed by the mechanical motions of the propellers.
To obtain the total system model of the UAV, we consider the following assumptions:

Assumptions:

1. The quadcopter is a rigid body and it has a symmetrical structure.
2. Center of gravity and center of mass coincides with a quadcopter geometrical center.

$$-\frac{\pi}{2} < \varphi < \frac{\pi}{2} \;;\; -\frac{\pi}{2} < \theta < \frac{\pi}{2} \;;\; -\frac{\pi}{2} < \psi < \frac{\pi}{2}$$

3. Both the position and velocity of the UAV are measurable.
4. The thrust and drag forces are proportional to the squared velocity of the propellers.

The modeling approach is based on Newton's laws and Euler's theorem. It is made up of two dynamics (Arnaud, 2012):

1. Translational dynamic

$$\sum F_{ext} = m\, I_{3x3}\, \dot{V} + \Omega \wedge \mathrm{m\, V} \qquad (1)$$

2. Rotational dynamic

$$\sum M_{ext} = I\, \dot{\Omega} + \Omega \wedge \mathrm{I}\, \Omega \qquad (2)$$

Table 1. Nomenclature of Newton-Euler formalism

Symbol	Meaning
$\sum \mathrm{F_{ext}}$	Sum of external forces
$\sum \mathrm{M_{ext}}$	Sum of external moments
m	mass of the drone
I	Inertial matrix
V	Vector of linear speed
Ω	Vector of angular speed
I_{3x3}	Unit matrix of size 3

All the mechanical actions applied to the UAV are cited in Table 2

Table 2. Mechanical actions and sources

Symbol	Mechanical Action
P	Gravity
F_{thrust}	Propeller thrust
F_{drag}	Air drag
τ_x, τ_y, τ_z	Inertial torques
τ_{aero}	Aerodynamic friction torque
τ_{gyro}	Gyroscopic torque

2.3 Translational dynamics model

The forces applied to the system are:

- Gravity:

Using the universal law of gravitation, we have:

$$P = [0\ 0 - mg]^T \qquad (3)$$

Where P is the vector of the gravity force expressed in R_b frame along z axis, and g is the gravitational constant.

- The thrust force:

To simplify the notation, trigonometric functions *cos(.)* and *sin(.)* are shortened as *c(.)* and *s(.)*

$$F_{thrust} = b \sum_{i=1}^{4} \omega_i^2 \begin{bmatrix} c(\psi)\ s(\theta)\ c(\varphi) + s(\psi)\ s(\varphi) \\ c(\varphi)\ s(\psi)\ s(\theta) - c(\psi)\ s(\varphi) \\ c(\theta)\, c(\varphi) \end{bmatrix} \qquad (4)$$

The thrust force produced by propeller i expressed in R_b, with thrust coefficient b in $N.s^2/m$ and ω_i is angular speed of motor i in rad/s^2.

- The drag force:

$$F_{drag} = -\begin{bmatrix} C_{dx} & 0 & 0 \\ 0 & C_{dy} & 0 \\ 0 & 0 & C_{dz} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = -\begin{bmatrix} C_{dx}\, \dot{x} \\ C_{dy}\, \dot{y} \\ C_{dz}\, \dot{z} \end{bmatrix} \qquad (5)$$

Where $[C_{dx}, C_{dy}, C_{dz}]$ are the translational drag coefficients F_{drag} is expressed in R_b.
Applying equation (1) we have

$$\sum F_{ext} = m\, \ddot{\xi} = F_{thrust} + F_{drag} + P \qquad (6)$$

ξ is the position of mass center in earth coordinate R_g.
The equation of translational motion is defined in the following equation (Mahony et al., 2012), (Pounds et al., 2010):

$$\begin{cases} \ddot{x} = \dfrac{b}{m}\omega_{\Sigma^+}\ (c(\psi)\ s(\theta)\ c(\varphi) + s(\psi)\ s(\varphi)) - \dfrac{C_{dx}}{m}\dot{x} \\ \ddot{y} = \dfrac{b}{m}\omega_{\Sigma^+}\ (c(\varphi)\ s(\psi)\ s(\theta) - c(\psi)\ s(\varphi)) - \dfrac{C_{dy}}{m}\dot{y} \\ \ddot{z} = \dfrac{b}{m}\omega_{\Sigma^+}\ (c(\theta)\, c(\varphi)) - \dfrac{C_{dz}}{m}\dot{z} - g \end{cases} \qquad (7)$$

with: $\omega_{\Sigma^+} = \omega_1{}^2 + \omega_2{}^2 + \omega_3{}^2 + \omega_4{}^2$

2.4 Rotational dynamics model

The torques applied to the system are:

- Roll torque:

$$\tau_x = \begin{bmatrix} 0 \\ -l \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ F_2 \end{bmatrix} + \begin{bmatrix} 0 \\ l \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ F_4 \end{bmatrix} = \begin{bmatrix} lb(\omega_4{}^2 - \omega_2{}^2) \\ 0 \\ 0 \end{bmatrix} \quad (8)$$

- Pitch torque:

$$\tau_y = \begin{bmatrix} l \\ 0 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ F_1 \end{bmatrix} + \begin{bmatrix} -l \\ 0 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 0 \\ 0 \\ F_3 \end{bmatrix} = \begin{bmatrix} 0 \\ lb(\omega_3{}^2 - \omega_1{}^2) \\ 0 \end{bmatrix} \quad (9)$$

- Yaw torque:

$$\tau_z = \begin{bmatrix} 0 \\ 0 \\ d\ \omega_{\Sigma^-} \end{bmatrix} \quad (10)$$

with: $\omega_{\Sigma^-} = \omega_1{}^2 - \omega_2{}^2 + \omega_3{}^2 - \omega_4{}^2$

F_i**:** force acting on the propeller i in N;
l: distance between the propeller axis and the mass center of the UAV in m.
d: drag coefficient in $N^2 s$.

- Aerodynamic friction torque :

$$\tau_{aero} = -C_a \begin{bmatrix} \dot{\varphi}^2 \\ \dot{\theta}^2 \\ \dot{\psi}^2 \end{bmatrix}$$

$$\tau_{aero} = -\begin{bmatrix} C_{ax} & 0 & 0 \\ 0 & C_{ay} & 0 \\ 0 & 0 & C_{az} \end{bmatrix} \begin{bmatrix} \dot{\varphi}^2 \\ \dot{\theta}^2 \\ \dot{\psi}^2 \end{bmatrix} = -\begin{bmatrix} C_{ax}\,\dot{\varphi}^2 \\ C_{ay}\,\dot{\theta}^2 \\ C_{az}\,\dot{\psi}^2 \end{bmatrix} \quad (11)$$

where $[C_{ax}, C_{ay}, C_{az}]$ are aerodynamic friction coefficients.

- Gyroscopic effect from propeller:

$$\tau_{gyro} = J_r \Omega_r \begin{bmatrix} \dot{\theta} \\ -\dot{\varphi} \\ 0 \end{bmatrix} \quad (12)$$

with: $\Omega_r = \sum_{i=1}^{4} (-1)^{i+1} \omega_i$

Where Jr is the moment of inertia of the propeller.
All the torques are expressed in R_b frame**.**
Based on equation (2) we have

$$\begin{cases} \ddot{\varphi} = \dfrac{lb(\omega_4{}^2 - \omega_2{}^2)}{I_x} - \dfrac{C_{ax}}{I_x}\,\dot{\varphi}^2 - \dfrac{J_r\Omega_r}{I_x}\dot{\theta} - \dfrac{(I_z - I_x)}{I_x}\dot{\theta}\dot{\psi} \\ \ddot{\theta} = \dfrac{lb(\omega_3{}^2 - \omega_1{}^2)}{I_y} - \dfrac{C_{ay}}{I_y}\,\dot{\theta}^2 + \dfrac{J_r\Omega_r}{I_y}\dot{\varphi} - \dfrac{(I_x - I_z)}{I_y}\dot{\varphi}\dot{\psi} \\ \ddot{\psi} = \dfrac{d}{I_z}\,\omega_{\Sigma^-} - \dfrac{C_{az}}{I_z}\,\dot{\psi}^2 - \dfrac{(I_y - I_x)}{I_z}\dot{\varphi}\dot{\theta} \end{cases} \quad (13)$$

Combining (7) and (13), the final dynamic model governing the system is as shown in equation (14):

$$\dot{X} = f(X, U) \quad (14)$$

$$\dot{x}_1 = x_2 \quad 14.a$$

$$\ddot{x}_1 = \frac{u_1}{m}u_x - \frac{C_{dx}}{m}x_2 \quad 14.b$$

$$\dot{y}_1 = y_2 \quad 14.c$$

$$\ddot{y}_1 = \frac{u_1}{m}u_y - \frac{C_{dy}}{m}y_2 \quad 14.d$$

$$\dot{z}_1 = z_2 \quad 14.e$$

$$\ddot{z}_1 = \frac{u_1}{m}c(\theta)\,c(\varphi) - \frac{C_{dz}}{m}z_2 - g \quad 14.f$$

$$\dot{\varphi}_1 = \varphi_2 \quad 14.g$$

$$\ddot{\varphi}_1 = \frac{u_2}{I_x} - \frac{C_{ax}}{I_x}\,\varphi_2{}^2 - \frac{J_r\Omega_r}{I_x}\theta_2 - \frac{(I_z - I_x)}{I_x}\theta_2\psi_2 \quad 14.h$$

$$\dot{\theta}_1 = \theta_2 \quad 14.i$$

$$\ddot{\theta}_1 = \frac{u_3}{I_y} - \frac{C_{ay}}{I_y}\,\theta_2{}^2 + \frac{J_r\Omega_r}{I_y}\varphi_2 - \frac{(I_x - I_z)}{I_y}\varphi_2\psi_2 \quad 14.j$$

$$\dot{\psi}_1 = \psi_2 \quad 14.k$$

$$\ddot{\psi}_1 = \frac{u_4}{I_z} - \frac{C_{az}}{I_z}\,\psi_2{}^2 - \frac{(I_y - I_x)}{I_z}\varphi_2\theta_2 \quad 14.l$$

where the state vector of the system:

$$X = (x_1\ \ x_2\ \ y_1\ \ y_2\ \ z_1\ \ z_2\ \ \varphi_1\ \ \varphi_2\ \ \theta_1\ \ \theta_2\ \ \psi_1\ \ \psi_2\,)^T = (x\ \ \dot{x}\ \ y\ \ \dot{y}\ \ z\ \ \dot{z}\ \ \varphi\ \ \dot{\varphi}\ \ \theta\ \ \dot{\theta}\ \ \psi\ \ \dot{\psi})^T \quad (15)$$

the command vector U defined by:

$$U = (u_1\ \ \ u_2\ \ \ u_3\ \ \ u_4)^T \quad (16)$$

And I_x, I_y, and I_z are moments of inertia about x, y, and z axes, respectively, in $kg.m^2$.
$u_x; u_y;\ u_1;\ u_2;\ u_3;\ u_4$ are defined as:

$$\begin{cases} u_x = c(\psi)\ s(\theta)\ c(\varphi) + s(\psi)\ s(\varphi) \\ u_y = c(\varphi)\ s(\psi)\ s(\theta) - c(\psi)\ s(\varphi) \\ u_1 = b\ \omega_{\Sigma^+} \\ u_2 = lb(\omega_4{}^2 - \omega_2{}^2) \\ u_3 = lb(\omega_3{}^2 - \omega_1{}^2) \\ u_4 = d\ \omega_{\Sigma^-} \end{cases} \quad (17)$$

3. QUADCOPTER CONTROL

The quadcopter is an under-actuated system (Hou et al.,2010), which means, the six degrees of freedom in space (*x, y, z,* φ, θ, ψ) are controlled only with four motors (motor1, motor2, motor3, motor4) as shown in the first figure.
The main objective is to design a robust controller to track all types of input commands in a noisy environment. The controller is conceived as a trade-off of robustness and performance to track the desired trajectory without going beyond the constructor's limitations of the motors. The control system is used to provide signals $(u_1; u_2; u_3; u_4)$ to the motor drivers in order to control their speed according to the desired movement. Thus thrust, pitch, yaw, and roll of the quadcopter can be controlled.
The control scheme of the quadcopter can be represented as in Figure 2, it consists of two loops: the attitude control loop (*the inner loop*), produces the control commands $(u_2; u_3; u_4)$ for the quadcopter to move. Moreover, the position control loop (*the outer loop*) produces the references $(\varphi_d\,, \theta_d\,, \psi_d\,)$ for the

inner loop through a nonlinear decoupling equations bloc. It also produces the control command (u_1) for the UAV to take off.

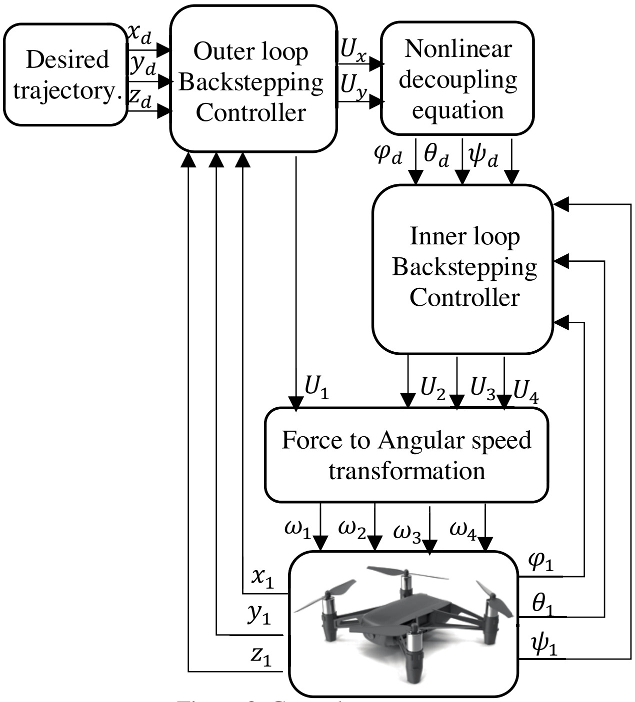

Figure 2. Control structure

According to (14) The instantaneous mathematical model describing the system can be expressed by the following nonlinear differential equations:
Let the tracking errors defined as:

$$\begin{cases} e_{z1} = z_1 - z_1^*\,;\; e_{z2} = z_2 - \dot{z}_1^* - C_{z1}\, e_{z1} \\ e_{x1} = x_1 - x_1^*\,;\; e_{x2} = x_2 - \dot{x}_1^* - C_{x1}\, e_{x1} \\ e_{y1} = y_1 - y_1^*\,;\; e_{y2} = y_2 - \dot{y}_1^* - C_{y1}\, e_{y1} \\ e_{\varphi 1} = \varphi_1 - \varphi_1^*\,;\; e_{\varphi 2} = \varphi_2 - \dot{\varphi}_1^* - C_{\varphi 1}\, e_{\varphi 1} \\ e_{\theta 1} = \theta_1 - \theta_1^*\,;\; e_{\theta 2} = \theta_2 - \dot{\theta}_1^* - C_{\theta 1}\, e_{\theta 1} \\ e_{\psi 1} = \psi_1 - \psi_1^*\,;\; e_{\psi 2} = \psi_2 - \dot{\psi}_1^* - C_{\psi 1}\, e_{\psi 1} \end{cases} \quad (18)$$

where

- e_{z1}, e_{x1}, e_{y1} are respectively altitude, x-motion and y motion tracking errors
- z_1^*, x_1^*, y_1^* are desired trajectory specified by a reference model.
- $e_{\varphi 1}$, $e_{\theta 1}$, $e_{\psi 1}$ are respectively roll, pitch and yaw tracking errors.
- $\varphi_1^*, \theta_1^*, \psi_1^*$ are desired orientation specified by a reference model.

3.1 Altitude control

Considering the subsystem (14.e) and (14.f):

Step 1

The tracking error in altitude is defined as:

$$e_{z1} = z_1 - z_1^* \quad (19)$$

Based on equation (20), derivative of the tracking error will therefore be:

$$\dot{e}_{z1} = z_2 - \dot{z}_1^* \quad (20)$$

Considering the following Lyapunov function:

$$V_1 = \frac{1}{2} e_{z1}^2 \quad (21)$$

Its time derivative is given by:

$$\dot{V}_1 = e_{z1}\, \dot{e}_{z1} = -C_{z1}\, e_{z1}^2 < 0 \quad (22)$$

where C_{z1} is a positive parameter.

$$\dot{e}_{z1} = z_2 - \dot{z}_1^* = -C_{z1}\, e_{z1}$$

$$\Rightarrow z_2^* = -C_{z1}\, e_{z1} + \dot{z}_1^* \quad (23)$$

As z_2^* is not the actual control law, we define the second tracking error:

$$e_{z2} = z_2 - z_2^* \quad (24)$$

Equation (24) becomes:

$$\dot{e}_{z1} = -C_{z1}\, e_{z1} + e_{z2} \quad (25)$$

Rewriting Lyapunov's function time derivative $\dot{V}_1$as:

$$\dot{V}_1 = -C_{z1}\, e_{z1}^2 + e_{z1} e_{z2} \quad (26)$$

Step 2:

Time-derivation of the second tracking error results becomes:

$$\dot{e}_{z2} = \frac{u_1}{m} c(\theta)\, c(\varphi) - \frac{C_{dz}}{m} z_2 - g - z_2^* \quad (27)$$

Let us consider the following choice of the augmented Lyapunov function:

$$V_2 = V_1 + \frac{1}{2} e_{z2}^2 \quad (28)$$

Using the previous equations, we get to:

$$\dot{V}_2 = -C_{z1}\, e_{z1}^2 + e_{z2}(e_{z1} + \dot{e}_{z2}) \quad (29)$$

Our objective is to make $\boldsymbol{\dot{V}_2}$ negative by the next choice:

$$e_{z1} + \dot{e}_{z2} = -C_{z2}\, e_{z2} \quad (30)$$

Where C_{z2} is a positive regulator parameter.
The combination of equations (14.f) and (30) lead to the control laws given by:

$$u_1 = \frac{m}{c(\theta)\, c(\varphi)} \left(-C_{z2}\, e_{z2} - e_{z1} + \dot{z}_2^* + \frac{C_{dz}}{m} z_2 + g \right) \quad (31)$$

3.2 x and y Motion Control

Now, considering the subsystems (14.a), (14.b) and (14.c), (14.d)
u_x and u_y are the orientations of u_1 which is responsible for the x and y motion respectively, can be extracted in a similar way as:

$$u_x = \frac{m}{u_1} \left(-C_{x2}\, e_{x2} - e_{x1} + \dot{x}_2^* + \frac{C_{dx}}{m} x_2 \right) \quad (32)$$

$$u_y = \frac{m}{u_1} \left(-C_{y2}\, e_{y2} - e_{y1} + \dot{y}_2^* + \frac{C_{dy}}{m} y_2 \right) \quad (33)$$

3.3 Roll control

Based on subsystem (14.g) and (14.h) and similarly the control input u_2 responsible for generating the Roll control φ can be calculated as:

$$\begin{pmatrix}\dot{\varphi}_1\\ \dot{\varphi}_2\end{pmatrix} = \begin{pmatrix}\varphi_2\\ \frac{u_2}{I_x} - \frac{C_{ax}}{I_x}\varphi_2{}^2 - \frac{J_r\Omega_r}{I_x}\theta_2 - \frac{(I_z-I_x)}{I_x}\theta_2\psi_2\end{pmatrix} \quad (34)$$

$$\mathrm{u}_2 = \mathrm{I}_x\left(-C_{\varphi 2}\,e_{\varphi 2} - e_{\varphi 1} + \dot{\varphi}_2{}^* + \frac{\mathrm{C_{ax}}}{\mathrm{I_x}}\varphi_2{}^2 + \frac{\mathrm{J_r\Omega_r}}{\mathrm{I_x}}\theta_2 + \frac{(\mathrm{I_z}-\mathrm{I_x})}{\mathrm{I_x}}\theta_2\psi_2\right) \quad (35)$$

3.4 Pitch control:

Considering subsystem (14.i) and (14.j) and similarly to the previous work the control input u_3 responsible for generating the pitch rotation θ can be calculated as:

$$\mathrm{u}_3 = \mathrm{I_y}\left(-\mathrm{C}_{\theta 2}\,\mathrm{e}_{\theta 2} - \mathrm{e}_{\theta 1} + \dot{\theta}_2{}^* + \frac{\mathrm{C_{ay}}}{\mathrm{I_y}}\theta_2{}^2 - \frac{\mathrm{J_r\Omega_r}}{\mathrm{I_y}}\varphi_2 + \frac{(\mathrm{I_x}-\mathrm{I_z})}{\mathrm{I_y}}\varphi_2\psi_2\right) \quad (36)$$

3.5 Yaw or heading control:

Base on subsystem (14.k) and (14.l) the control input u_4 responsible for generating the yaw rotation ψ can be calculated as:

$$u_4 = I_z\left(-C_{\psi 2}\,e_{\psi 2} - e_{\psi 1} + \dot{\psi}_2{}^* + \frac{C_{az}}{I_z}\psi_2{}^2 + \frac{(I_y - I_x)}{I_z}\varphi_2\,\theta_2\right) \quad (37)$$

4. NONLINEAR DECOUPLING EQUATIONS

The desired φ_{1d} and θ_{1d} angles are determined according to the following equations (38) and (39):

$$U_x = c(\psi_1)\,.s(\theta_{1d}).c(\varphi_{1d}) + s(\psi_1)\,s(\varphi_{1d}) \quad (38)$$

$$U_y = s(\psi_1)\,.s(\theta_{1d}).c(\varphi_{1d}) - c(\psi_1)\,s(\varphi_{1d}) \quad (39)$$

After some arrangement, we get:

$$\varphi_{1d} = sin^{-1}(U_x.s(\psi_1) - U_y.c(\psi_1)) \quad (40)$$

$$\theta_{1d} = sin^{-1}\left(U_x.\frac{c(\psi_1)}{c(\varphi_{1d})} + U_y.\frac{s(\psi_1)}{c(\varphi_{1d})}\right). \quad (41)$$

Desired yaw angle ψ_{1d} will be obtained such that quadrotor's heading and direction of motion in $x - y$ plane is on the same line.

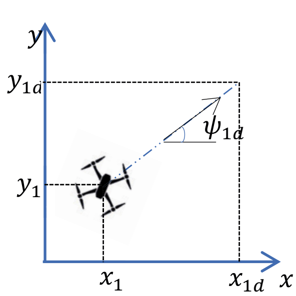

Figure3: Top view of the quadcopter, motion in x-y plane

Using Figure 3 and basic trigonometry, desired yaw angle ψ_{1d} can be obtained as follows

$$\psi_{1d} = tan^{-1}\frac{y_{1d}-y_1}{x_{1d}-x_1} \quad (42)$$

In case where the trajectory (x, y) is not in the form of a line, the desired yaw angle ψ_{1d} will be calculated at each step time (depending on the simulator sampling time), so that the shape of the trajectory between t and $t + T_e$ is a line, until reaching the target. Sampling time (T_e) must imperatively be very small.

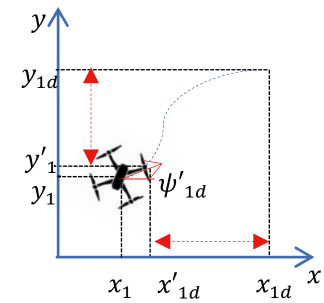

Figure 4: Top view of the quadcopter, motion in x-y plane

5. SIMULATION RESULTS

This section illustrates simulation results of the regulator using MATLAB/Simulink environment. In order to test the performances acquired to control attitude, altitude.

Figure 5 represents the trajectory in the case of a helical path. Figure 6 shows the desired and real positions (x, y, z) for this same setpoint.

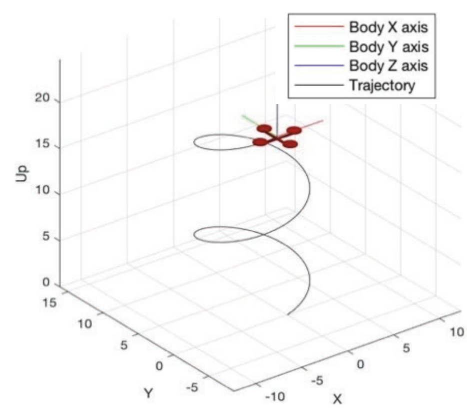

figure 5: 3D real trajectory of the UAV for an helicoidal path

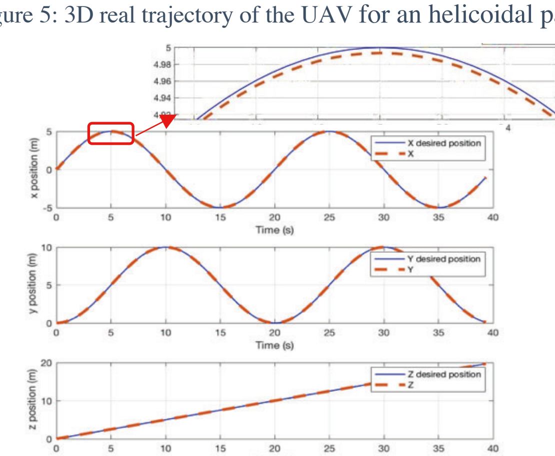

Figure 6: real and desired positions for an helicoidal path

Figure 7 highlights the trajectory in the case of a step reference.
Figure 8 represents the desired positions (x, y, z) and the corresponding position measures for this same path.
The plot of time response x and y shows a response time equal to 5s as for z the response time is fixed at 0.4s; no overshoot detected.

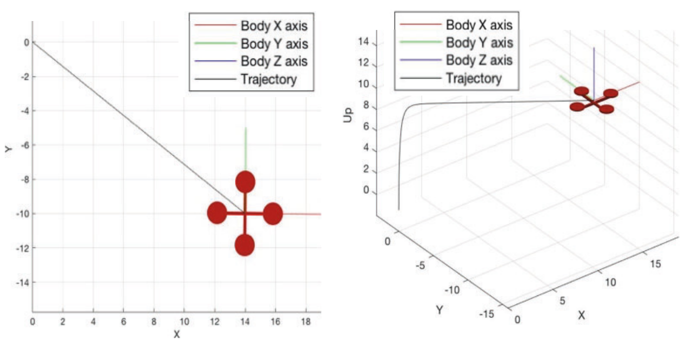

figure 7: 3D and 2D real trajectory of the UAV for step setpoint

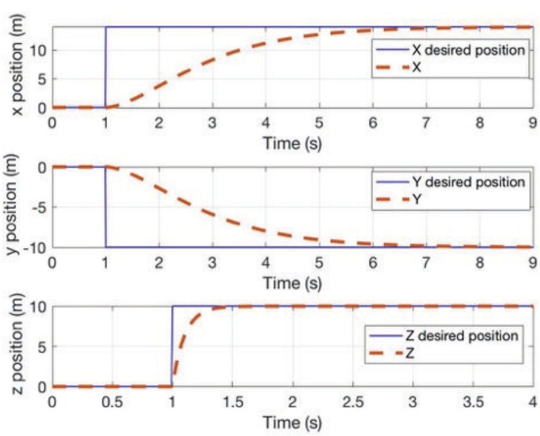

Figure 8: x, y and z desired and real trajectory for step setpoints

The following figure 9 shows the desired and measured positions for linear path.

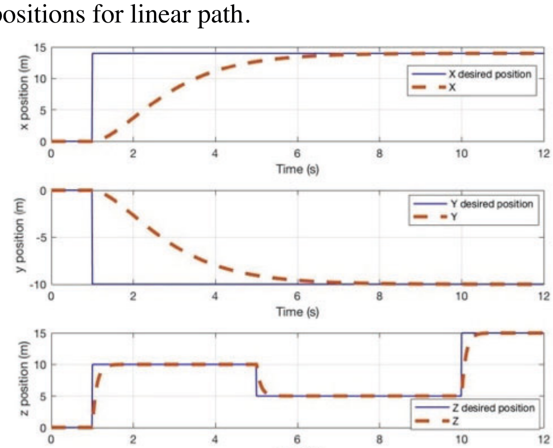

Figure 9: x, y and z desired and real for a specific linear scenario

6. CONCLUSION

In this paper, the dynamic modeling of a quadcopter drone is obtained using the Euler-Lagrange formalism. Based on this physical model, a cascade control strategy with the backstepping approach was developed with the main objective of guiding the quadcopter in both altitude and attitude to assure trajectory tracking tasks. Simulations in a closed-loop system were used to validate the controller's effectiveness, and characteristics such as stability, response time, and stationary error were observed. It is worth mentioning that the control system presented here was considered effective since it performed the proposed trajectory tracking and desired orientation tasks. As a suggestion for future work, a robustness analysis should be performed in the face of parametric uncertainties and aerodynamic disturbances.

7. REFERENCES

Arnaud K (2012) Modélisation, observation et commande d'un drone miniature à birotor coaxial. *Automatique Robotique*. Université Henri Poincaré - Nancy I Français. fftel-00709015f.

Bashi O.I.D, Hasan W.Z.W, Azis N, Shafie S, Wagatsuma H (2016) Unmanned aerial vehicle quadcopter A review. *J. Comput. Theor. Nanosci.*, *38*,529–554.

Faizan A, Warsi D, Hazry S, Faiz Ahmed, M Kamran Joyo, M Hassan Tanveer (2013). Roll Angle Stabilization of Fixed-wing UAVs in occurrence of noises by using PID with EKF controller. *Journal of JÖKULL* Vol 63, No. 9, 0449-0576.

Hamel T, Mahoney R, Lozano R, Ostrowski J (2002). Dynamic modelling and configuration stabilization for an X4-flyer. *15th IFAC world congress*, Barcelona.

Hussein Hamadi, (2020). Fault-tolerant control of a multirotor unmanned aerial vehicle under hardware and software failures. *Automatic Control Engineering*. Université de Technologie de Compiègne.

Hou H, Zhuang J, Xia H, Wang G, and Yu D (2010). A simple controller of minisize quad-rotor vehicle. *International Conference on Mechatronics and Automation (ICMA)*, CHINA

Mahony R, Kumar V, Corke P (2012). "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *Robotics Automation Magazine, IEEE*, vol. 19, no. 3, pp. 20–32.

Pounds P, Mahony R, Corke P (2010). "Modelling and control of a large quadrotor robot," *Control Engineering Practice*, vol. 18, no. 7, pp. 691 – 699.

Sabir A, Alia Z (2018). Modeling of a Quadcopter Trajectory Tracking System Using PID Modeling of a Quadcopter Trajectory Tracking System Using PID. *The 12th International Conference Interdisciplinarity in Engineering*, Romania.

Vibhu K, Laxmidher B, Nishchal V (2015). Design of Sliding mode and Backstepping Controllers for a Quadcopter. *In 39th National Systems Conference (NSC)*, Greater Noida, India.

View publication stats