République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE DJILLALI LIABES DE SIDI BEL-ABBES

Faculté de Génie Electrique

Département d'Electrotechnique

Thèse présentée par:

BOUCHAIB Ali

Pour l'obtention du diplôme de:

Doctorat en Sciences

Spécialité: Electrotechnique

Option: Commande Electrique

Intitulé de la thèse:

Contribution à la technique géométrique pour la commande d'un quadrotor (Contribution to the geometric technique for control of a quadrotor)

Présentée devant le jury composé de:

SEMMAH Abdelhafid	Professeur.	U.D.L. Sidi Bel-Abbès	Président
TALEB Rachid	Professeur	Université de Chlef	Directeur
MASSOUM Ahmed	Professeur	U.D.L. Sidi Bel-Abbès	Co- Directeur
BENTAALLAH Abderrahim	Professeur	U.D.L. Sidi Bel-Abbès	Examinateur
HELAIMI M'hamed	Professeur	Université de Chlef	Examinateur
BOUDJEMA Zinelaabidine	MCA	Université de Chlef	Examinateur

Soutenue le: 26 / 05 / 2022


بسم الله الرحمن الرحيم

Aforementioned

First of all, I would like to thank ALLAH the Almighty for the will, the health, and the patience he has given me during all these long years.

The work presented in this thesis was carried out at the Intelligent Control and Electrical Power Systems Laboratory (ICEPS) of Djillali Liabes University, Sidi Bel Abbes, Algeria.

I am deeply thankful to Pr. TALEB Rachid, Professor at the UHB de Chlef. This work would not be successful without your supervising. Who guided, advised, and helped me throughout this work, and for offering me such a rich and fascinating subject.

I warmly thank Pr. MAASOUM Ahmed, Professor at the Djillali Liabes University, Sidi Bel Abbes, Algeria, for supervising this thesis. Your scientific and humane qualities will always be a source of motivation.

I want to express my deep gratitude to Prof. Dr. Saad Mekhilef (Professor at the Faculty of Engineering, University of Malaya) for welcoming me to his laboratory and for allowing me to work in excellent conditions. Thank you for your morale and kindness throughout my visit.

I warmly thank Pr. SEMMAH Abdelhafid, Professor at Djillali Liabes University, Sidi Bel Abbes, Algeria, for having done me the honor of chairing the defense jury.

I would also like to express my sincere thanks to Pr. BENTAALLAH Abderrahim, Professor at the Djillali Liabes University, Sidi Bel Abbes, Algeria, to Pr. HELAIMI M'hamed, Professor at the UHB Chlef, Algeria, to Dr. BOUDJMAA Zinelaabidine, MCA at the UHB to Chlef, Algeria, for having accepted the charge of examiners.

My sincere regards also go to all my fellow teachers at the department of electrical engineering, UHB. Chlef. To all those who have helped me by contributing, directly or indirectly to the outcome of this work, may you find in this thesis a trace of my gratitude. I would like to specifically mention Messrs Dr. Abderhmane BENBOUALI and Dr. Sid Ahmed DEROUAZINE, Lecturers at UHB. Chlef.

I do not want to forget mentioning Mouaz Al Kouzbary at University of Malaya for helping me in carrying out this work.

Last but not least I would like to thank my family who always were there for me throughout the ups and downs of my research journey.

Table of Contents

Figure List

Table List

Abbreviations

BGC	Backstepping Geometric Control
CLF	Control Lyapunov Function
HDP	High Dynamic Performance Control
IC	Impedance Control
RMSE	Root Mean Square Error
$SE(3)$	Special Eucledean group
$SO(3)$	Special Orthogonal group
UAV	Unmanned Aerial Vehicle

Nomenclature

a_r	Linear acceleration of the sensor.
$b \in \mathbb{R}$	Thurst coefficient kg. $rad^{-1}.s^{-1}$
$\{\overrightarrow{b_1}, \overrightarrow{b_2}, \overrightarrow{b_3}\}$	Body fixed frame
d	is drag coefficient $kg.m^2/rad^2$
E_K	kinetic energy
E_p	Potential energy
$e_R \in \mathbb{R}^3$	Attitud error
g $\in \mathbb{R}$	Gravitational acceleration, $m.s^{-2}$
G	Groupe Galilean
L	Lagrangian
$m \in \mathbb{R}$	Mass of quadrotor, kg
$M \in \mathbb{R}^3$	Moments vector in the body fixed frame, kg.rad.m
$\mathbb{J}$	Inertia matrix for the full rotational kinetic energy
$r \in \mathbb{R}$	Location of the centre of the mass, m
$T \in \mathbb{R}^3$	Total thrust, kg. $m.s^{-2}$
$R \in SO(3)$	Rotation matrix
Q	The divergence matrix
θ, ϕ, ψ	Respictivly Pitch, Rool and Yaw angles (rad)
Ω	Angular velocity in the body frame rad/s

ξ	Vertual control
Γ_i	The generalized forces
ϑ_i	The generalized coordinates

General Introduction

General Introduction

For the last two decades the field of unmanned aerial vehicles (UAVs) was under rapid growth and many applications were improved by utilizing UAVs (e.g., military inspection, emergency rescue services, mapping, etc.). Because of the intensive research in the filed [1, 2]. Recently, a growing interest in the multi-rotor UAVs for the systems high capabilities (i.e., vertical take-off, and landing, navigation in narrow spaces and maneuvering capacity) [3-5]. However, the advantages of multi-rotor UAVs comes with highly coupled dynamic models which makes designing a decoupled control law that stabilizes the systems a challenging task [6-8]. The progress in the domain of multi-rotor UAVs leads to the adoption of geometric modelling to avoid singularities and ambiguity [9]. However, the nonlinearity and highly coupled systems require an advanced control system. The backstepping control method is considered one of the advanced approaches to regulating nonlinear systems, which can give robust and decoupled control laws [10, 11].

Recently, a new control approach for dynamic systems was developed which is based on nonlinear manifolds that cannot be globally identified with Euclidean spaces. In previous studies [12, 13], a control approach that was designed by characterizing the geometric properties of nonlinear manifolds intrinsically. The geometric control techniques can completely avoid the singularities and ambiguities issues, this approach was applied to fully actuated rigid body utilizing mathematical method of Lie groups to achieve almost global asymptotic stability [14-16]. Furthermore, the flying environment's internal and external perturbations do not affect the durability of the mentioned control [17-20].

In this work, a two-level cascade control system based on integral-backstepping technique was designed to achieve global stability and robustness against mismatching parameters. The low-level is the attitude to adjust the UAVs' attitude angles and maintain a stable flight. The control loop is based on geometric control topology which makes the system able to avoid singularities and commands' ambiguity, then we defined the attitude as a virtual control of the angular dynamic and utilized Lyapunov's theory to ensure system stability.

Moreover, the high-level control system, which is the transitional control loop, we took the linear velocity to be virtual control in this subsystem. First, this control subsystem makes the UAVs reach the desired position. Secondly, it translates the path information to the virtual control inputs of the low-level control system. Finally, the complex trajectories are designed to be the reference input of the system.

More recently, a research team from Washington University suggested new [9, 21-23] control methods based on a completely different kind of attitude representation. This representation takes into account the variety structure of rotation, since each rotation of our robot is, in fact, a displacement from one point to another in the whole group of orthogonal matrices SO (3). The attitude representation based on the SO (3) set avoids singularities and ambiguities. However, these works use very complicated methods, from the point of view of computational capacity, without taking into account the resolution of the problems of the sensitivity of the drone vis-à-vis the various internal and external disturbances, such as: parametric variations or navigation in the presence of gusts of wind.

work contribution is the Development of the "Geometric backstepping" control for the attitude control of the quadrotor taking into account the SO (3) group, while guaranteeing its simplicity, stability, and robustness; Real-time evaluation of attitude control algorithms in the mambo mini-drone by replacing its own impedance control algorithm which has always been considered an inaccessible black box. Insertion of the principle of High Dynamic Performance Control in the controls.

This work is organized as follows:

The first step towards designing control and navigation algorithms is to state a mathematical model to work with. In this chapter, several modelling methodologies are explored such that robust algorithms can be developed. Most of the work is focused on aerial vehicles since they are in general, more challenging control and navigate compared to UGVs due to their faster dynamics and unstable nature, the selected configuration for the UAV in this thesis was a quadrotor.

In the second chapter, a detailed description of geometric mechanics, where we will detail the different notions of the “differential manifold” configuration as well as the mathematical tools necessary for the modelling in this space. In addition, we will develop Euler-Lagrange equations for dynamical systems over an arbitrary Lie group. This synthesis makes it

possible to obtain models of different mechanical systems, in particular, and a detailed description of backstepping control and Lyapunov stability.

The third chapter will be divided into two parts, the first will be reserved for the description and the modelling of the quadrotor in the Lie group. While in the second, we will propose a new control technique, based on the geometric control, nicknamed "the geometric Backstepping control". The study of the stability of this command will also be discussed. The performance of this command will be determined from simulation tests and applied to the AR Drone 2.0. The various robustness tests will also be inserted.

The fourth chapter will be devoted to the most interesting of our contributions; The principles of Impedance control and "High-Performance Dynamic" control will be discussed at the start of this chapter,

The last chapter, we will move on to applying the Geometric Backstepping Control for roll and pitch, experimental tests will be applied with the same scenarios to be able to identify the interest of this new approach [23].

Chapter I
Modelling Approaches of quadrotor

I.1 Introduction

Most of the applied control systems on rigid body systems, are based on simplified dynamic models, which often present singularities when any complex rotation manoeuvre is needed. Fundamentally limiting the UAVs ability to follow important trajectories. In this chapter, different types of dynamics models of the quadrotor are discussed, based on Euler-Lagrange then modelling according to Newton-Euler and modelling according to a quaternion approach,

I.2 Quadrotor modeling and control works

Several works have been published in the field of quadrotors' control, these works differ according to the type of dynamic models and control systems. They can be classified to four methods, to develop the mechanical dynamic model. First, model is based on Newton-Euler which used in [24] with an adaptive control. Moreover, in [25] the same system used with adaptative backstepping control. In [26] a backstepping control system were applied to track nonlinear trajectories where the dynamic model developed based on the first method.

Second, the Euler-Lagrange method to find the system dynamic model [27], the LQR control system was used with manual high-level command. The LQR method depend on the linearized dynamic system to find the feedback gains [28].

Third, the representation of the dynamic model based on quaternion [29] where a nominal controller and a robust compensator were used to stabilize the system. The nominal controller determines the desired poles of the nominal linearized closed-loop control system and the robust compensator is introduced to attenuate the effects of the input equivalent disturbances.

In [30], a quaternion-based dynamic model was established with hysteretic hybrid feedback that robustly globally asymptotically stabilizes the attitude of the quadrotor body. The hybrid control laws are derived using Lyapunov analysis of the nonlinear dynamic model. Another work in [31], where a conventional PD control system with an exact linearization is also derived in order to apply the linear control algorithms.

More recent work on the dynamics of quadrotors UAV start emerging, where the dynamic model can be globally expressed (i.e., the special Euclidean group, SE (3)). Moreover, with the computational capability to apply nonlinear control systems which can expand the ability to track complicated path with different modes under numerous environment

uncertainties. Additionally, several aggressive maneuvers of a quadrotor can be achieved based on a hybrid control architecture. As the hybrid control systems are directly developed on the special Euclidean group, complexities, singularities, and ambiguities associated with minimal attitude representations or quaternions can be completely avoided [7]. One of the first research in this field was represented in [32], where a global dynamic model and developed control systems for three different flight modes were presented. Both were developed in terms of the special Euclidean group that is intrinsic and coordinate-free. In [33], similar control system and dynamic representation were developed and implemented to control a quadrotor that is connected to the ground via a tether. The main contribution of this work is the design of a new geometric controller using rotation matrix attitude representation. In this work geometric controller is integrated with the backstepping controller system, to ensure avoiding ambiguity and singularity with the stability of the system.

I.3 Force and moment in a rotor

According to blade element theory, which is used to model quadrotor and rotor performances, the forces and moment developed on a uniform propeller are determined by the lift and drag forces and a pitching torque. For a given rotor j with angular velocity ω_j, the linear velocity at each point along the propeller is proportional to the radial distance from the rotor shaft $T_j = b\omega_j^2$, $\tau_j = d\omega_j^2$ [34].

I.3.1 Lifting forces:

The lift provides elevation, thanks to this quad-rotor compensates for the weight, takes off, and stays in the air. The calculation of the lift of a rotor is the first step in the process of estimating the forces acting on the structure of the aircraft [3]. The produced lift by a rotor j is given by: $T_j = b\omega_j^2$. Where b the lift coefficient of $kg.m/rad^2$ depends on blade shape and area density [35] ,[36] .

I.3.2 Drag forces:

The drag corresponds to the resistance to advancement through the air The author in [37] proposed the following form of drag $D_i = C_d.\rho.\omega^2$. Where $C_d$represents a positive constant multiplied by the surfaces of the blade, and ρ is air density

The action of air on each rotor creates a counter-rotation torque acting along the pivot axis of each rotor. In our case, the action of air on the blades of a rotor is proportional to the square of the angular speed of these blades. So, the expression of anti-couple is written [3], [38], [39],

$\Gamma_i = d.\Omega_i^2$, where d is drag coefficient $kg.m^2/rad^2$

I.3.3 Different moments of the quad-rotor:

When the four rotors of the quadrotor are powered, it produced a moment which is the cause of the rotational movement (roll, pitch, and yaw). These couples are due to the different forces of lift and drag.

I.3.4 Moments due to lifting forces:

The translation on the X-axis is due to the moment created by the difference between the lift forces of rotors 1,2 and 3,4. This moment is given by the following relation

$$M_1 = lb(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) \quad \text{(I.1)}$$

Where, l is the length of the arm between the rotor and center of gravity of the quadrotor.

The translation on the Y-axis is due to the moment created by the difference between the lift forces of rotors 1,4 and 2,3. This moment is given by the following relation:

$$M_2 = lb(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \quad \text{(I.2)}$$

I.3.5 Moments due to drag forces:

The rotation around the Z-axis is due to a reactive torque caused by drag couples caused by drag torques in each propeller This moment is given by the following relation:

$$M_3 = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \quad \text{(I.3)}$$

I.4 Dynamic Quadrotor model

I.4.1 Rotation matrix

Consider that the fixed frame $O_0x_0y_0z_0$ and current frame $O_1x_1y_1z_1$, Three independent parameters are necessary to completely describe the orientation of the reference frame $O_1x_1y_1z_1$ compared to that of $O_0x_0y_0z_0$. The passage from the reference $O_1x_1y_1z_1$ to the reference $O_0x_0y_0z_0$ will be made by three rotations using two intermediate references $O_ax_ay_az_a$ and $O_bx_by_bz_b$.[40].

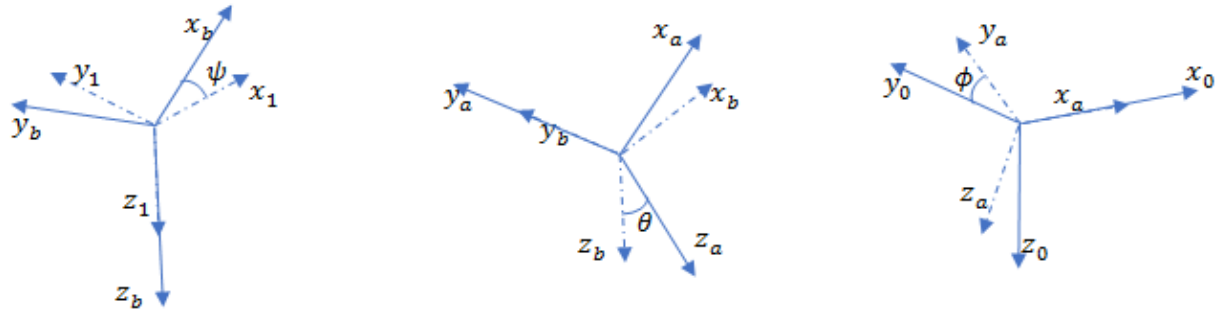


Figure I.1 Rotation around the X-axis (Roll), Y-axis (Pitch) and Z-axis (Yaw)

I.4.2 Passage from the $O_1x_1y_1z_1$ reference mark to the $O_bx_by_bz_b$ reference mark

The rotation is done around the axis $z_1 = z_b$. We pass from the landmark $O_bx_by_bz_b$ to the landmark $O_1x_1y_1z_1$ by rotating the angle ψ called the yaw angle. So, we have the matrix:

$$Rot_{z_0}(\psi) = \begin{bmatrix} c\psi & -s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{(I.4)}$$

I.4.3 Passage from the $O_bx_by_bz_b$ landmark to the $O_ax_ay_az_a$ landmark

The rotation is done around the axis $z_b = z_a$. Then pass from the landmark $O_ax_ay_az_a$ to the landmark $O_bx_by_bz_b$ by rotating the angle θ called the pitch angle. So, we have the matrix:

$$Rot_{y_0}(\theta) = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \quad \text{(I.5)}$$

I.4.4 Passage from the $O_ax_ay_az_a$ landmark to the $O_0x_0y_0z_0$ landmark

The rotation is done around the axis $x_a = x_0$. Then pass from the frame $O_0x_0y_0z_0$ to $O_ax_ay_az_a$ by making a rotation of angle ϕ called roll angle.

$$Rot_{x_0}(\phi) = \begin{bmatrix} 1 & 0 & 1 \\ 0 & s\phi & c\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \quad \text{(I.6)}$$

The passage from the reference $O_1x_1y_1z_1$ to the reference $O_0x_0y_0z_0$ or vice versa is done by three successive rotations so that all the axes of $O_1x_1y_1z_1$ occupy positions different from that of $O_0x_0y_0z_0$. The rotation matrix from $O_1x_1y_1z_1$ to $O_0x_0y_0z_0$ is given by the product of the three successive matrices, we obtain:

$$R = Rot_{z_0}(\psi) * Rot_{x_0}(\theta) * Rot_{y_0}(\theta)$$

$$R = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\phi s\psi \\ s\psi c\theta & s\psi s\theta s\phi + c\phi c\psi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad \text{(I.7)}$$

Where $c\phi = \cos(\phi)$ and $s\phi = \sin(\phi)$ same thing for the angles ψ, θ

I.5 modelling according to Euler-Lagrange

If the solid undergoes three successive rotations according to the aeronautical angles, we can then express any point $r(x_1, y_1, z_1)$ of the solid in the landmark by:

$$r = R(\phi, \theta, \psi)(x_0, y_0, z_0)^T$$

$$\begin{cases} r_1 = (c\psi c\theta)x_0 + (c\psi s\theta s\phi - s\psi c\phi)y_0 + (c\psi s\theta c\phi + s\phi s\psi)z_0 \\ r_2 = (s\psi c\theta)x_0 + (s\psi s\theta s\phi + c\phi c\psi)y_0 + (s\psi s\theta c\phi - c\psi s\phi)z_0 \\ r_3 = (-s\theta)x_0 + (c\theta s\phi)y_0 + (c\theta c\phi)z_0 \end{cases} \quad \text{(I.8)}$$

from (I.8) the speed for any point is given by

$$V^2 = V_1^2 + V_2^2 + V_3^2 \quad \text{(I.9)}$$

Kinetic energy: From equation (I.9) and assuming that the inertia matrix is diagonal, we can extract the expression of kinetic energy:

$$E_K = \frac{1}{2}J_x\left(\dot{\phi} - \dot{\psi}s\theta\right)^2 + \frac{1}{2}J_y\left(\dot{\theta}c\phi + \dot{\psi}s\phi c\theta\right)^2 + \frac{1}{2}J_z\left(\dot{\theta}s\phi - \dot{\psi}c\phi c\theta\right)^2 \quad \text{(I.10)}$$

Potential energy: using the well-known potential energy formula, we find:

$$E_p = \int xdm(x)(-gs\theta) + \int ydm(y)(gs\phi c\theta) + \int zdm(z)(c\phi c\theta) \quad \text{(I.11)}$$

Using the Lagrangian model and the derived formula for the equations of motion:

$$L = E_k - E_p, \ \Gamma_i = \frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial \mathrm{L}}{\partial \dot{\vartheta}_i}\right) - \frac{dL}{\vartheta_i} \quad \text{(I.12)}$$

Where: ϑ_i, Γ_i are generalized coordinates and generalized forces respectively. The three movements are then:

$$\begin{cases} J_x\ddot{\phi} = \dot{\theta}\dot{\psi}(J_y - J_z) \\ J_y\ddot{\theta} = \dot{\phi}\dot{\psi}(J_z - J_x) \\ J_z\ddot{\psi} = \dot{\phi}\dot{\psi}(J_x - J_y) \end{cases} \quad \text{(I.13)}$$

I.5.1 Dynamic derivative model

The dynamic quadrotor model describing the roll, pitch, and yaw rotations then contain three terms which are: the gyroscopic effect resulting from the body rotation, the gyroscopic effect resulting from the rotation of the propeller coupled with the rotation of the body, and finally the moments generated by the actuators:

$$\begin{cases} J_x\ddot{\phi} = \dot{\theta}\dot{\psi}(J_y - J_z) - J\dot{\theta}\Omega_r + M_x \\ J_y\ddot{\theta} = \dot{\phi}\dot{\psi}(J_z - J_x) - J\dot{\phi}\Omega_r + M_y \\ J_z\ddot{\psi} = \dot{\theta}\dot{\phi}(J_x - J_y) + M_x \end{cases} \quad \text{(I.14)}$$

Where:

$$\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4 \quad \text{(I.15)}$$

I.5.2 Dynamic modeling of Lozano

We present in this section the work of Lozano et al. [41, 42] which uses the Euler-Lagrange approach to model the quadrotor. In this work, the goal is to find Lagrangian (L).

Kinetic energy is made up of two terms. The first term is the kinetic energy of translation and the second term is the kinetic energy of rotation. The generalized coordinates of the quadrotor are described by:

$$\vartheta = (x, y, z, \phi, \theta, \psi) \in R^6 \quad \text{(I.16)}$$

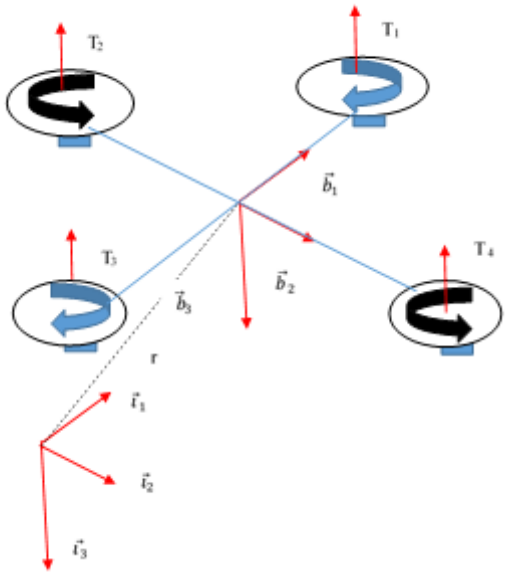


Figure I.2. the quad-rotor in an inertial frame of reference

where $r = (x, y, z)$ represent the position of the helicopter's center of mass with respect to a fixed reference $\vec{\imath}$.

$\Theta = (\phi, \theta, \psi)$ are the three angles of Euler (yaw, pitch, roll) and represent the orientation of the helicopter. The Lagrangian is:

$$L(\vartheta, \dot{\vartheta}) = E_{k_{Trans}} + E_{k_{Rot}} - E_p \tag{I.17}$$

the kinetic energy of translation of the helicopter is given by:

$$E_{k_{Trans}} = \frac{m}{2}\dot{r}^T\dot{r} \tag{I.18}$$

The kinetic energy of rotation is given by:

$$E_{k_{Rot}} = \frac{1}{2}\dot{\Theta}^T\mathbb{J}\dot{\Theta} \tag{I.19}$$

Where:

$$\mathbb{J} = W_\Theta^T I W_\Theta \tag{I .20}$$

And

$$W_\Theta = \begin{bmatrix} -sin\theta & 0 & 1 \\ cos\theta sin\phi & cos\phi & 0 \\ cos\theta cos\phi & -sin\phi & 0 \end{bmatrix} \tag{I.21}$$

Thus, the matrix $\mathbb{J} = \mathbb{J}(\Theta)$ acts as the inertia matrix for the full rotational kinetic energy of the quadrotor, expressed directly in terms of the generalized coordinates Θ. The model of the full rotorcraft dynamics is obtained from Euler–Lagrange equations with external generalized forces [41]

Potential energy is given by:

$$E_p = mgz \tag{I.22}$$

The Lagrangian L is:

$$L(\vartheta, \dot{\vartheta}) = \frac{m}{2}\dot{r}^T\dot{r} + \frac{1}{2}\dot{\Theta}^T\mathbb{J}\dot{\Theta} - mgz \tag{I.23}$$

The dynamic model of the helicopter is obtained from the Euler-Lagrange equations with external generalized force.

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\vartheta}}\right) - \left(\frac{\partial \mathcal{L}}{\partial \vartheta}\right) = \begin{bmatrix} \hat{T} \\ M \end{bmatrix} \tag{I.24}$$

Where, $\hat{T} = RT \in R^3$ is the translational force applied to the aircraft due to the control inputs, $M \in R^3$ is the generalized moments.

We write thus:

$$Te_3 = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} \tag{I.25}$$

Where $e_3 = [0\ 0\ 1]^T$ and

$$T = \sum_{i=1}^{4} T_i \tag{I.26}$$

And

$$M = \begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} (-T_1 + T_2 - T_3 + T_4)c_\tau \\ (T_4 - T_2) \\ (T_1 - T_3)l \end{bmatrix} \tag{I.27}$$

Where l is the distance between the motors and the center of gravity $c_\tau = \frac{d}{b}$ is the drag coefficient.

So we get:

$$m\ddot{r} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = \hat{T} = RTe_3 \tag{I.28}$$

$$\mathbb{J}\ddot{\Theta} + C\left(\dot{\vartheta}, \vartheta\right)\dot{\vartheta} = M \tag{I.29}$$

$$C\left(\dot{\vartheta}, \vartheta\right) = \dot{\mathbb{J}} - \frac{1}{2}\frac{\partial}{\partial \vartheta}\left(\dot{\vartheta}^T \mathbb{J}\right) \tag{I.30}$$

is the Coriolis matrix and contains the gyroscopic and centrifugal parameters. Finally, we get:

$$\begin{aligned} m\ddot{x} &= T(s\phi s\psi + c\phi c\psi s\theta) \\ m\ddot{y} &= T(c\phi s\theta s\psi + c\psi s\theta) \\ m\ddot{z} &= T(c\phi c\theta - mg) \\ \ddot{\phi} &= \widetilde{M}_\phi \\ \ddot{\theta} &= \widetilde{M}_\theta \\ \ddot{\psi} &= \widetilde{M}_\psi \end{aligned} \tag{I.31}$$

Where x and y are the coordinates in the horizontal plane, z is the vertical position, $\widetilde{M}_\phi$, $\widetilde{M}_\theta$ and $\widetilde{M}_\psi$ are the new angular moments (roll moment, pitch moment, and yaw moment),

$$\begin{bmatrix} \widetilde{M}_{\phi} \\ \widetilde{M}_{\theta} \\ \widetilde{M}_{\psi} \end{bmatrix} = \mathbb{J}^{-1}(M - C(\vartheta, \dot{\vartheta})\dot{\vartheta}) \tag{I.32}$$

It should be noted that each of the control inputs can be used to control one or two degrees of freedom as follows. The collective input T is mainly used to give the altitude the desired value. the the $\widetilde{M}_{\phi}$input is used to control the roll angle and the horizontal movement in the y axis. $\widetilde{M}_{\theta}$ is used to control the pitching movement and the horizontal movement in the x axis, similarly $\widetilde{M}_{\psi}$ is used to control yaw displacement.

I.6 Modeling with Newton-Euler formalism

The general motion of a rigid body in space is a combination of translational and rotational motions. Consider a rigid body moving in inertial space, undergoing both rotations and translations. Let us define now an earth-fixed frame I and a body-fixed frame B, as seen in Fig. I.1. The center of mass and the body-fixed frame are assumed to coincide. Using Euler angles parametrization, the airframe orientation in space is given by a rotation R from B to I, where R ∈ SO (3) is the rotation matrix.

I.6.1 Translation dynamics

According to Newton's first law of dynamics:

$$\frac{d(mV)}{dt} = \sum F_{ex} \tag{I.33}$$

The external forces applied to the quadrotor are:

The weight

$$P = mgz_0 \tag{I.34}$$

Drag force

It is the result of the forces opposing the movement of the quadrotor in the air, in the same direction as the movement of the quadrotor but opposite directions. It somehow represents the viscous friction forces on the object, it is given [43-45]

$$D_i = -C_d|V|V \tag{I.35}$$

Where $C_d = \frac{1}{2}\rho k$

Lifting force

It is perpendicular to the airflow, directed upwards, that is to say, it tends to raise the quadrotor. It represents the total force produced by the four propellers; it is given by:

$$T = (\sum_{i=1}^{4} T_i) z_1 \tag{I.36}$$

We must express this force in the inertial frame of reference, for that we use the rotation matrix R:

$$T = (\sum_{i=1}^{4} T_i) R_{z_0} \tag{I.37}$$

By replacing the expressions of forces, equation (I.33) becomes:

$$\dot{V} = -g z_0 - \frac{C_d}{m} |V| V + \frac{T}{m} R_{z_0} \tag{I.38}$$

By developing equation (I.38), we obtain the system of equations:

$$\begin{cases} \ddot{x} = -\frac{C_d^{\dot{x}}}{m} |\dot{x}| \dot{x} + \frac{T}{m} (s\phi s\psi + s\theta c\phi c\psi) \\ \ddot{y} = -\frac{C_d^{\dot{y}}}{m} |\dot{y}| \dot{y} + \frac{T}{m} (s\theta s\psi c\phi - s\theta c\phi c\psi) \\ \ddot{z} = -\frac{C_d^{\dot{z}}}{m} |\dot{z}| \dot{z} + \frac{T}{m} (c\theta c\phi) + g \end{cases} \tag{I.39}$$

I.6.2 Rotational dynamics

According to Newton's second law of dynamics:

$$\frac{dJ\Omega}{dt} = \sum M \tag{I.40}$$

And as the angular speed is expressed in the reference frame linked to the quadrotor, then:

$$\frac{dJ\Omega}{dt} = \mathrm{J}\dot{\Omega} + \Omega \wedge J\Omega \tag{I.41}$$

I.6.3 Aerodynamic moment

They are produced by the drag and thrust forces created by the rotation of the four propellers.

$$\begin{cases} M_x = bl(\omega_4^2 + \omega_2^2) \\ M_y = bl(\omega_3^2 + \omega_1^2) \\ M_z = d(\omega_1^2 - \omega_2^3 + \omega_3^2 - \omega_4^2) \end{cases} \tag{I.42}$$

I.6.4 The gyroscopic moment

It is created in moving physical systems with rotary parts and tends to resist the movements of the quadrotor. The general expression of this moment is given by:

$$M_{gyro} = \sum_{i=1}^{4} \Omega \wedge J_r \begin{bmatrix} 0 \\ 0 \\ (-1)^i \omega_i \end{bmatrix} \tag{I.43}$$

The inertia matrix of each rotor is assumed to be diagonal:

$$J_r = \begin{bmatrix} J_{rx} & 0 & 0 \\ 0 & J_{ry} & 0 \\ 0 & 0 & J_{rz} \end{bmatrix} \tag{I.44}$$

From equations (I.43) and (I.44) we obtain:

$$M_{gyro} = \sum_{i=1}^{4} \left((-1)^i \omega_i J_{rz}\right) \Omega \wedge z_1 \tag{I.45}$$

By replacing the expression (I.41) in equation (I.45):

$$\mathrm{J}\dot{\Omega} + \Omega \wedge J\Omega = \sum_{i=1}^{4} \left((-1)^i \omega_i J_{rz}\right) \Omega \wedge z_1 \tag{I.46}$$

As the structure of the quadrotor is supposed to be symmetrical, the inertia matrix J is diagonal:

$$J = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \tag{I.47}$$

So, equation (I.46) becomes:

$$\begin{cases} \dot{p} = \left(\frac{J_y - J_z}{J_x}\right) qr + \frac{J_{rz}}{J_{rx}} (\omega_1 - \omega_2 + \omega_3 - \omega_4) q + \frac{M_x}{J_x} \\ \dot{q} = \left(\frac{J_z - J_x}{J_y}\right) pr - \frac{J_{rz}}{J_{ry}} (\omega_1 - \omega_2 + \omega_3 - \omega_4) p + \frac{M_y}{J_y} \\ \dot{r} = \left(\frac{J_x - J_y}{J_z}\right) pq + + \frac{M_z}{J_z} \end{cases} \tag{I.48}$$

I.6.5 Relationship between Euler angles and angular velocities

If a solid rotates at a constant speed, its angular speed Ω is constant, on the other hand, the variations of the angles of Euler will be variable because they depend on the instantaneous angles between the axes of the two landmarks. The sequence of Euler angles is obtained from three successive rotations: roll, pitch, and yaw. Which give:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{I.49}$$

So the relationship between $\Omega = [\dot{\phi} \quad \dot{\theta} \quad \dot{\psi}]^T$ and $\dot{\Phi} = [p \quad q \quad r]^T$

$$\Omega = R_r \, \dot{\Phi} \tag{I.50}$$

$$R_r = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix} \tag{I.51}$$

I.6.6 State model

The final dynamic system of the quadrotor given by (I.38), (I.47) is subdivided into four subsystems (position / linear speed, Euler angles, angular speed, and rotor equations). An architecture of this type, where the state of the first system is controlled by an indirect command, which is actually the next state of the system, is called a cascade structure, this leads to a design of a multi-loop controller. The dynamic model containing the actuator model will be:

$$\begin{cases} \ddot{x} = \frac{1}{m}[(s\phi s\psi + s\theta c\phi c\psi)T] \\ \ddot{y} = \frac{1}{m}[(s\theta s\psi c\phi - s\theta c\phi c\psi)T] \\ \ddot{z} = -\frac{1}{m}[(c\theta c\phi)T] + g \\ \dot{p} = \left(\frac{J_y - J_z}{J_x}\right) qr + J_{rz}\Omega_r q + \frac{1}{J_x} M_1 \\ \dot{q} = \left(\frac{J_z - J_x}{J_y}\right) pr - J_{rz}\Omega_r p + \frac{1}{J_y} M_2 \\ \dot{r} = \left(\frac{J_x - J_y}{J_z}\right) pq + J_{rz} + \frac{1}{J_z} M_3 \end{cases} \tag{I.52}$$

For reasons of simplification and since the angular movements of the quadrotor are of low amplitude and low speed, this amounts to writing $\Omega = [\dot{\phi} \quad \dot{\theta} \quad \dot{\psi}]^T$ and neglecting the drag and lift forces. The simplified dynamic model not containing the actuator model will be:

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}[(s\phi s\psi + s\theta c\phi c\psi)T] \\
\ddot{y} &= \frac{1}{m}[(s\theta s\psi c\phi - s\theta c\phi c\psi)T] \\
\ddot{z} &= -\frac{1}{m}[(c\theta c\phi)T] + g \\
\ddot{\phi} &= \left(\frac{J_y - J_z}{J_x}\right)\dot{\theta}\dot{\psi} + J_{rz}\Omega_r\dot{\theta} + \frac{1}{J_x}M_1 \\
\ddot{\theta} &= \left(\frac{J_z - J_x}{J_y}\right)\dot{\phi}\dot{\psi} - J_{rz}\Omega_r\dot{\phi} + \frac{1}{J_y}M_2 \\
\ddot{\psi} &= \left(\frac{J_x - J_y}{J_z}\right)\dot{\phi}\dot{\theta} + J_{rz} + \frac{1}{J_z}M_3
\end{aligned} \tag{I.53}$$

I.7 modelling according to quaternion:

Quaternions, which belong to the quaternion space $\mathbb{H}$, can be represented in many ways, but the one presented here is as a sum of a scalar component along an imaginary vector [46-48] .

$$\boldsymbol{\alpha} = \alpha_0 + \bar{\alpha} = \alpha_0 + \begin{bmatrix}\alpha_1 \\ \alpha_2 \\ \alpha_3\end{bmatrix} \tag{I.54}$$

Where $\boldsymbol{\alpha} \in \mathbb{H}$ is a given quaternion, $\bar{\alpha}$ is the complex vectorial part of $\boldsymbol{\alpha}$, and α_0 is the scalar part of $\boldsymbol{\alpha}$

The quaternion $\boldsymbol{\alpha}$ can be written

$$\boldsymbol{\alpha} = \alpha_0 + \alpha_1 i + \alpha_2 j + \alpha_3 k \tag{I.55}$$

The quaternion number $\boldsymbol{\alpha}$ consists of four units $1, i, j, and\ k$ with real parameters α_i $(i = 0, 1, 2, 3)$, where, i, j, and k denote three Orthogonal unit spatial vectors. Considering that the quaternion includes three-dimensional vectors, the vectors can be defined as quaternion with no scalar part in all quaternion operations. As viewed in figure I.2, the rigid body rotation is represented in the axis angle framework. The magnitude of the rotation vector $\vec{\vartheta} = [\vartheta_x \quad \vartheta_y \quad \vartheta_z]$ is defined in radians by $\vec{u} = \frac{\vec{\vartheta}}{\|\vartheta\|}$ and the axis angle representation is denoted as V4 using the unitary vector $\vec{\vartheta} = \vartheta\vec{u}$

.

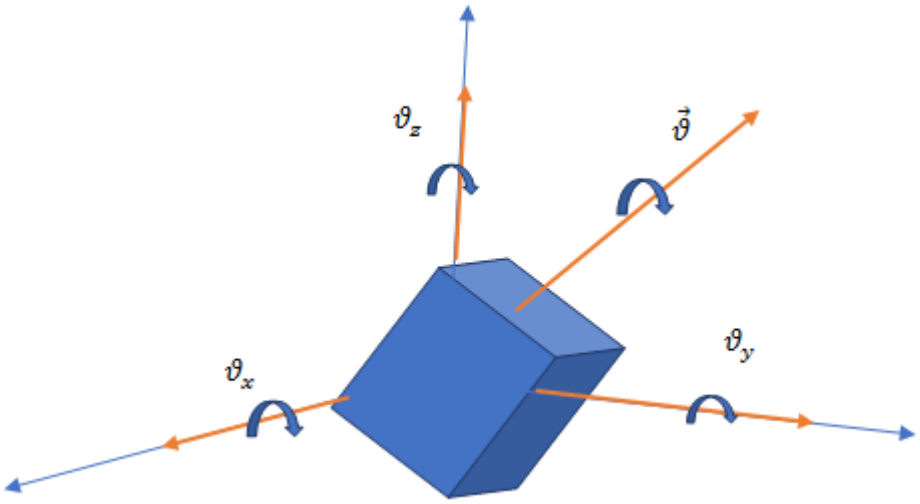


Figure I.3 Axis-angle representation of a rigid body rotation

Using the Euler formula, a simple rotation with a magnitude φ in radians can be defined by $e^{i\omega} = cos\varphi + isin\varphi$. The exponential mapping of the axis representation of a rotation is denoted by the Euler Rodrigues formula; which is called a unit quaternion:

$$\boldsymbol{\alpha} = e^{\frac{1}{2}\vartheta\vec{u}} = \cos(\frac{\vartheta}{2}) + \vec{u}sin(\frac{\vartheta}{2}) \tag{I.56}$$

Were, $\|\boldsymbol{\alpha}\| = 1$.

Deriving the quaternion using the logarithmic mapping $\vec{\vartheta} = 2\ln(\boldsymbol{\alpha})$, the axis angle representation can be obtained revesly [34, 48, 49].

With:

$$\ln(\boldsymbol{\alpha}) = \begin{cases} [0 \quad 0 \quad 0]^T & if \|\vec{\alpha}\| = 0 \\ \frac{\boldsymbol{\alpha}}{\|\boldsymbol{\alpha}\|} & if \|\vec{\alpha}\| \neq 0 \end{cases} \tag{I.57}$$

I.7.1 Quaternion algebra

Product:

In the definition of the quaternion space, the most important operation is the multiplication. from this point, other operations and properties like the conjugate and the norm are derived from this definition. Considering $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{H}$

$$\boldsymbol{\alpha} \otimes \boldsymbol{\beta} := \left(\alpha_0\beta_0 - \vec{\alpha}.\vec{\beta}\right) + \left(\alpha_0\vec{\beta} + \beta_0\vec{\alpha} + \vec{\alpha} \times \vec{\beta}\right) \tag{I.58}$$

Due to the same non-commutativity property of the cross product, quaternion product is not commutative $\boldsymbol{\alpha} \otimes \boldsymbol{\beta} :\neq \boldsymbol{\beta} \otimes \boldsymbol{\alpha}$.

Sum: knowing that the set of all quaternions with operations addition and multiplication defines a non-commutative division ring, the sum of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is defined as [47]:

$$\boldsymbol{\alpha} + \boldsymbol{\beta} \coloneqq \alpha_0 + \beta_0 + \vec{\alpha} + \vec{\beta} \tag{I.59}$$

Conjugate:

The inverse rotation over the same axis is defined as the conjugate of a unit quaternion (I.60) the product of conjugate is expressed by (I.61):

$$\boldsymbol{\alpha}^* \coloneqq \alpha_0 - \vec{\alpha} \tag{I.60}$$

$$(\boldsymbol{\alpha} \otimes \boldsymbol{\beta})^* \coloneqq \boldsymbol{\alpha}^* \otimes \boldsymbol{\beta}^* \tag{I.61}$$

Norm: Equation II.62 defines the norm of a quaternion as follow:

$$\|\boldsymbol{\alpha}\|^2 \coloneqq \boldsymbol{\alpha} \otimes \boldsymbol{\alpha}^* = \alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 \tag{I.62}$$

Reverse:

As the quaternion product forms a closed-loop group, a new quaternion results from the product of two non-null quaternions. Thus, for any non-null quaternion, its inverse can be defined as:

$$\boldsymbol{\alpha}^{-1} \coloneqq \frac{\boldsymbol{\alpha}^*}{\|\boldsymbol{\alpha}\|} \tag{I.63}$$

$$\boldsymbol{\alpha} \otimes \boldsymbol{\alpha}^{-1} = \boldsymbol{\alpha}^{-1} \otimes \boldsymbol{\alpha} = 1 + [0 \quad 0 \quad 0]^T \tag{I.64}$$

Vector Rotation:

Assuming that the vectors $\vec{r} \in R^3$ and $\overrightarrow{r'} \in R^3$ in a fixed frame and a mobile reference frame respectively, the translation of $\vec{r}$ into $\overrightarrow{r'}$ can be achieved using a double quaternion product as:

$$\overrightarrow{r'} = \boldsymbol{\alpha}^{-1} \otimes \vec{r} \otimes \boldsymbol{\alpha} = \boldsymbol{\alpha}^* \otimes \vec{r} \otimes \boldsymbol{\alpha} \tag{I.65}$$

Besides, the rotation of the second reference frame with respect to the first one is represented by the quaternion, where the vectors' magnitude is not affected by this rotation.

Derivative:

Using the previous operation, the derivative of a unit quaternion can be carried out by differentiating equation I.62 as follow:

$$\dot{\vec{r'}} = \dot{\boldsymbol{\alpha}}^{-1} \otimes \vec{r} \otimes \boldsymbol{\alpha} + \boldsymbol{\alpha}^{-1} \otimes \vec{r} \otimes \dot{\boldsymbol{\alpha}} = \dot{\boldsymbol{\alpha}}^{-1} \otimes \boldsymbol{\alpha} \otimes \vec{r}' + \vec{r}' \otimes \boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}} \quad \text{(II.66)}$$

We have $\boldsymbol{\alpha}^{-1} \otimes \boldsymbol{\alpha} = \mathbf{1}$

$$\dot{\vec{r'}} = \vec{r}' \otimes \boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}} \otimes \vec{r} = \mathbf{2}(\boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}}) \times r' \quad \text{(I.67)}$$

Note $\dot{\vec{r'}}$ is the translational velocity of the vector, $\dot{\vec{r'}} = \vec{\Omega} \times \vec{r}'$ where $\vec{\Omega}$ is the rotational velocity of $\vec{r}'$, thus

$$\vec{\Omega} \times \vec{r}' = \mathbf{2}(\boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}}) \times \vec{r}' \quad \text{(I.68)}$$

which can be reduced to

$$\vec{\Omega} = \mathbf{2}(\boldsymbol{\alpha}^{-1} \otimes \dot{\boldsymbol{\alpha}}) \Longrightarrow \dot{\boldsymbol{\alpha}} = \frac{\mathbf{1}}{\mathbf{2}} \boldsymbol{\alpha}^{-1} \otimes \vec{\Omega} \quad \text{(I.69)}$$

I.7.2 Rigid Body Dynamic Modeling

For any rigid body, the translational and rotational state is given by $\boldsymbol{X} \coloneqq \left[\vec{r} \quad \dot{\vec{r}} \quad \boldsymbol{\alpha} \quad \vec{\Omega}\right]^T$. The body position in the inertial frame I is $\vec{r} \in R^3$, the body position velocity is $\dot{\vec{r}}$, the unit queternion is defined by $\boldsymbol{\alpha}$, and $\vec{\Omega}$ is the rotational velocity in the moving reference frame located the body's center of mass. Thus, the dynamic model of any rigid body represented by a unit quaternions is derived using the Newton's equations of motion [34] as follow:

$$\dot{\boldsymbol{X}} = \frac{d}{dt}\begin{bmatrix} \vec{r} \\ \dot{\vec{r}} \\ \boldsymbol{\alpha} \\ \vec{\Omega} \end{bmatrix} = \begin{bmatrix} \dot{\vec{r}} \\ m^{-1} F_{\mathrm{I}} \\ \frac{1}{2}\boldsymbol{\alpha} \otimes \vec{\Omega} \\ J^{-1}\left(\vec{M} - \vec{\Omega} \times J\vec{\Omega}\right) \end{bmatrix} \quad \text{(I.70)}$$

F_{I} is the total force applied to the body in the I frame, J is the inertia matrix and $\vec{M}$ represents the total torque, both with respect to $\mathfrak{B}$. The description of any mechanical system including aerial vehicles can be obtained using equation (I.70) and thereby, the quadrotors dynamics is given in the next subsection using this approach.

I.7.3 Quadrotor quaternion dynamical model

As viewed in figure I.3, the free body diagram of the quadrotor is presented. This diagram can be adopted in this case if and only if the mechanical configuration is considered symmetric, and the blade flapping and the misalignment on the motor axes are approximated small enough. In this case, forces and torques are the only ones that act on the vehicle.

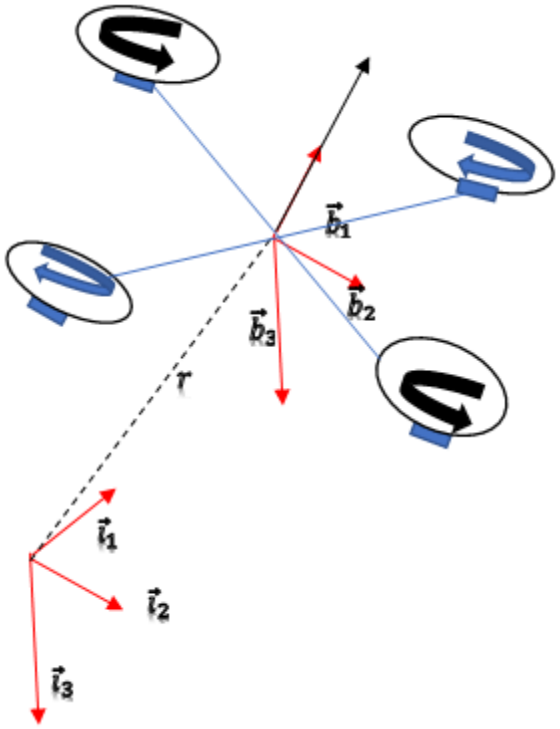


Figure I.4 Quadrotor free body diagram

Figure I.4 show two references are adopted, the fixed inertial coordinates $\mathfrak{I}$ and the moving body frame $\mathfrak{B}$ located on the quadrotor center of mass, in addition to the quaternion rotation from $\mathfrak{I}$ to $\mathfrak{B}$. Equation (I.71), define the aforementioned reference

$$\begin{cases} \mathfrak{I} = [i_1 \quad i_2 \quad i_3]^T \\ \mathfrak{B} = [b_1 \quad b_2 \quad b_3]^T \\ \alpha = e^{\frac{1}{2}\vec{\vartheta}} \end{cases} \tag{I.71}$$

Let $\vec{T}$ and $\vec{M}$ symbolize the total thrust force, and torque, since it act on B, the vehicle dynamics as:

$$\dot{\boldsymbol{X}} = \frac{d}{dt}\begin{bmatrix}\vec{r}\\ \dot{\vec{r}}\\ \boldsymbol{\alpha}\\ \vec{\Omega}\end{bmatrix} = \begin{bmatrix}\dot{\vec{r}}\\ \boldsymbol{\alpha}\frac{\vec{T}}{\boldsymbol{m}} \otimes \boldsymbol{\alpha}^* + \vec{g}\\ \frac{1}{2}\boldsymbol{\alpha} \otimes \vec{\Omega}\\ J^{-1}\left(\vec{M} - \vec{\Omega} \times J\vec{\Omega}\right)\end{bmatrix} \quad \text{(I.72)}$$

The vehicle dynamic is expressed by equation (II.72) using the quaternion rotation defined in equation (I.65), where $\vec{T}$ is the total thrust force and $\vec{M}$ is the torque. In addition, $\vec{T}$ and $\vec{M}$ act on $\mathfrak{B}$.

Knowing that the orientation of $\overrightarrow{T_I}$ depends on the vehicle attitude $\boldsymbol{\alpha}$, the quadrotors rotational and translational are coupled. Therefore, the quadrotor can be easily stabilized although its underactuated nature using a suitable approach and the properties of unit quaternions.

I.8 conclusion

This chapter provides a detailed discussion about different modelling of a quadrotor and discusses I how to find a mathematical model for a quadrotor, Focus has been given to the following modelling Euler-Lagrange, then Newton-Euler, and finally modelling quaternion modelling approach; Another modelling of quadrotor based on a nonlinear space called "variety". Geometric methods for the modelling and the control of these robots evolve on a nonlinear differential variety. Their analysis made it possible to avoid any singularity, ambiguity or complexity, during aggressive wide-angle manoeuvres or even complex manoeuvres.

Chapter II:

Geometric Modeling and backstepping Control

II. 1 Introduction

Another research team from the University of Washington also worked on the quadrotor, their idea consisted of considering the quadrotor as a dynamic system that evolves on a nonlinear space called "variety". They proposed, then, geometric methods for the modelling and the control of these robots which evolve on a nonlinear differential variety. Their analysis made it possible to avoid any singularity, ambiguity or complexity, during aggressive wide-angle manoeuvres or even complex manoeuvres. The following section details this concept by giving the different tools of this analysis. At the end of this chapter, we'll mention backstepping control synthesis, where this latter is a relatively recent control method for nonlinear systems. It allows sequentially and systematically to determine the control law of the system by the choice of a Lyapunov function. Its principle is to know how to constructively establish the control law of the nonlinear system by considering certain state variables as virtual commands and conceive them as intermediate control laws.

II. 2 Principle of Galilean relativity

The principles of relativity address the problem of how events that occur in one place or state of motion are observed from another place. so, how then can we determine the laws of motion, if the events which occur in one place or state of motion seem different from those of another.

Galileo approached this problem through observations of movement made inside a ship by people who could not see the outside. Galileo has shown that people isolated inside a ship, uniformly in motion, are not able to determine, by measurements taken from the inside, if they are moving! . . . Move the ship forward at the speed you desire, as long as the movement is uniform, you will not notice any change, and you will not be able to tell if the ship is moving or stationary [50]. Transformations of a reference location, time, orientation, or state with uniform translation at constant speed are called Galilean transformations.

II. 2.1 Galileo's transformation:

A Galileo transformation corresponds to the formulas of transformations of the spatial and temporal coordinates between two given Galilean frames of reference. Any frame of reference in rectilinear and uniform translational movement with respect to a given frame of reference assumed to be Galilean, is itself Galilean.

The set of Galileo's transformations constitutes a continuous group of transformations which has the ten-parameter Lie group structure, as follows:

- The translations of space-time: $g_1(r,t) = (r + r_0,\ t + \ t_0)$
- Arrangement of rotations and reflections: $g_2(r,t) = (Or,t)$
- For any orthogonal linear transformation $O :\rightarrow \mathbb{R}^{3N} \times \mathbb{R}^{3N}$ where $O^T = O^{-1}$
- Galilean 'boosts': in a uniform rectilinear motion $g_3(r,t) = (r + v_0\ t,\ t)$

II.2.2 Group concept:

The group G is a set of elements that has a binary product (multiplication), $G \times G \rightarrow G$ which have the following properties:

1. associative product, $g,h \in G$, the product gh is associative, when $(gh)k = g(hk)$
2. identity element, $g,e \in G$, $ge = g$.
3. reverse operation $G \rightarrow G$, $g^{-1}g = gg^{-1} = e$

II.2.3 the Lie group

A Lie group is a group that depends on a set of differential parameters [51, 52]. In other words, a Lie group is not only a group but also a differential manifold, for which the operations are differential functions.

II.2.4 Notion of sub-group

A subgroup is a subset of a group whose elements satisfy, in addition to the properties of the group, those of the subgroup. We cite, for example, the subgroups which derive from the Lie group.

II.2.5 Special three-dimensional orthogonal group SO (3):

SO (3) is a subgroup of Lie, and of Galilean in evidence [53], which is characterized by a proper rotation $g_2(O)$ where $g_2(O)r = Or$ where $O^T = O^{-1}$ and $\det(O) = 1$.

II.2.6 three-dimensional Euclidean special group SE (3):

SE (3) is a subgroup of Lie, and of Galilean in evidence [54], which is characterized by spatial translations $g_1(r_0)$composed with appropriate rotations $g_2(O)$ acting on r as:

$E(O,r_0)r = g_1(r_0)g_2(O)r = Or + r_0$, Where $O^T = O^{-1}$ and $\det(O) = 1$

II.3 Relationship between SO (3) and SE (3)

The special Euclidean group SE (3) is linked to the special orthogonal group SO (3) by a semi-direct product, $SE(3) = SO(3).R$

II.4 Lie algebra:

For any Lie group G [55], we denote by $g = T_eG$ the tangent space in the neutral element $e \in G$. This vector space is of capital importance, because it turns out that there is a bilinear map $\mathrm{g} \times \mathrm{g} \rightarrow \mathrm{g}$, called the bracket or commutator, The bracket satisfies bilnearity, alternativity and jacobi identity. In this part, we will expose, only, the mathematical tools necessary for the realization of this chapter

II.4.1 Left and right trivialization

For $\mathrm{g}, \mathrm{h} \in \mathrm{G}$, the left trivialization: $L_h: \mathrm{G} \rightarrow \mathrm{G}$ is defined by $L_h\mathrm{g} = \mathrm{hg}$; in the same way as for the right trivialization where: $R_h: \mathrm{G} \rightarrow \mathrm{G}$ is defined by $R_h\mathrm{g} = \mathrm{hg}$.

II.4.2 Vector field

For $\zeta \in \mathrm{G}$ We define the field $X_\zeta\, \mathrm{G} \rightarrow \mathrm{TG}$ in a way $X_\xi(\mathrm{g}) = T_eL_{\mathrm{g}}.\xi$ and that the corresponding unique integral curve passing through the identity e at $\mathrm{t} = 0$ be noted $\gamma_\zeta(t)$.

II.4.3 Exponential map

The exponential form exp: $g \rightarrow \mathrm{G}$ is defined with $\exp(\zeta) = \gamma_\zeta(1)$ The form exp is a local diffeomorphism from a vicinity of zero in g to a vicinity of e in G.

II.4.4 Adjoint Operator

Adjoint Operator ad_{g}: $g \rightarrow g$ is the differential of the internal product $I_g(h)$ compared to h when $h = e$ following the direction $\varsigma \in \mathrm{g}$ for example $ad_{\mathrm{g}}\varsigma = T_eI_{\mathrm{g}}.\varsigma$ The operator ad

$ad_\zeta : g \rightarrow g$ is obtained by differentiating $ad_{\mathrm{g}}\varsigma$ compared to g in e in the direction ζ $ad_\zeta\varsigma = T_c\left(ad_{\mathrm{g}}.\varsigma\right).\zeta$ This corresponds to the Lie hook $ad_{\mathrm{g}}.\varsigma = [\zeta, \varsigma]$. Remember that the internal product $I_{\mathrm{g}}(h) = \mathrm{g}h\mathrm{g}^{-1}$.

II.4.5 Co-adjoint operator

So be it $<.,.>$ the couple between a tangent vector and a cotangent vector. The coadjoint operator ad_{g}^*: $g^* \rightarrow g^*$ is defined by $< ad_{\mathrm{g}}^*\alpha, \zeta > = \alpha, < ad_{\mathrm{g}}^*\zeta >$ for $\alpha \in g^*$. The operator co-ad isad^* $g \times g^* \rightarrow g^*$ is defined by , $< ad_{\mathrm{g}}^*\alpha, \zeta >$ for $\alpha \in g^*$

II.5 Lagrangian formalism

In order to understand the method used in our work, it is necessary to take stock of the basic formalism allowing the description of a physical system, namely: the formalism Lagrangian.

It requires the introduction of a space of configurations Q of coordinates q^i, with i = {1, ..., n}. The behaviour of the studied system can then be described completely using the Lagrangian $L(q^i, \dot{q}^i)$ function of the positions q^i, and velocity $\dot{q}^i$

If we define the action as Lagrangian function.

$$\iota = L(q^i, \dot{q}^i)dt \tag{II.1}$$

The trajectory chosen by the system set in motion minimizes this functional, that is, it satisfies the equation $\delta\iota = 0$. This condition constitutes the so-called “least action” principle. If we consider a small variation of the position δq^i we can write using integration by parts:

$$\delta\iota = \sum_{i=1}^{n} \int_a^b \left(\frac{\partial L}{\partial q^i}\delta q^i + \frac{\partial L}{\partial \dot{q}^i}\delta \dot{q}^i\right) dt = \sum_{i=1}^{n} \int_a^b \left(\frac{\partial L}{\partial q^i} + \frac{\partial L}{\partial \dot{q}^i}\right) \delta q^i dt = 0 \tag{II.2}$$

Then, we obtain the so-called Lagrange equations of motion:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial q^i} + \frac{\partial L}{\partial \dot{q}^i}\right) = 0 \tag{II.3}$$

II.6 Euler-Lagrange equations

In this section, we will describe the Euler-Lagrange equations for mechanical systems that evolve over an arbitrary Lie group.

Consider a mechanical system evolving on the configuration space G, which in our case is a Lie group. The specificity in this analysis is to express the variation of group elements G in terms of the Lie algebra " g " using the exponential form. To do this, we follow the following steps:

We identify the tangent bundle TG with $G \times g$ by left trivialization. For example, the vector $(\mathrm{g}, \dot{\mathrm{g}}) \in T_{\mathrm{g}}G$ tangent is expressed by

$$\dot{\mathrm{g}} = T_e L_{\mathrm{g}}.\zeta = \mathrm{g}\zeta \tag{II.4}$$

Where $\zeta \in g$, We assume that the Lagrangian of the mechanical system is given $L(\mathrm{g}, \zeta): \mathrm{G} \times \mathrm{g} \rightarrow \mathbb{R}$

The principle of least action results in the variation of the integral action being zero.

$$\delta\zeta = \delta \int_{T_0}^{T_f} L(\mathrm{g}, \zeta) dt \tag{II.5}$$

where g(t) a differential curve in G defined in $t \in [t_0, t_f]$ the variation is a differentiable cartography.

for $c > 0$ in order to $\mathrm{g}^0(t) = \mathrm{g}(t)$ for $t \in [t_0, t_f]$

$$\begin{cases} \mathrm{g}^{\zeta}(t): (-c, c) \times [t_0, t_f] \to G \text{ for } c > 0 \; in \; order \; to \; \mathrm{g}^0(t) = \mathrm{g}(t) \text{ for } t \in [t_0, t_f] \\ \mathrm{g}^{\zeta}(t_0) =: \mathrm{g}(t_0)\mathrm{g}^{\zeta} f =: \mathrm{g}(t_f) for \; \varepsilon \in (-c, c) \end{cases}$$

The variation is expressed in exponential form by:

$$\mathrm{g}^{\zeta}(t) = \mathrm{g}.\exp\varepsilon\varsigma(t) \tag{III.6}$$

With $\varsigma(t) \in g$

The exponential relation (II.5) is a local diffeomorphism between g and G, and satisfies the properties of the variation $\varsigma(t_0) = \varsigma(t_f) = 0$. As long as this is obtained by a group operation, it also guarantees that the variation is on G for all $\varsigma(t)$. The corresponding infinitesimal variation of g is given by:

$$\delta\mathrm{g(t)} = \frac{d}{d\varepsilon}\Big|_{\varepsilon=0} \mathrm{g}^{\varepsilon}(t) = T_e L_{\mathrm{g(t)}} . \frac{d}{d\varepsilon}\Big|_{\varepsilon=0} exp\varepsilon\varsigma(t) = \mathrm{g}(t)\varsigma(t) \tag{II.7}$$

For all the $t \in [t_0, t_f]$ the infinitesimal variation $\delta\mathrm{g(t)}$ is located in the tangent space $\mathrm{g}T_{g(t)}G$ G. Using this expression and (II.4), the infinitesimal variation of $\zeta(t)$ is obtained as follows.

$$\delta\zeta(\mathrm{t}) = \dot{\varsigma}(t) + ad_{\zeta(t)}\varsigma(t) \tag{II.8}$$

Equations (II.6) and (II.7) are infinitesimal variations of $(\mathrm{g(t)}, \zeta(\mathrm{t})): [t_0, t_f] \to G \times g$ respectively.

- The Lagrangian variation can be written in the form:

$$\delta\mathrm{L}(\mathrm{g}, \zeta) = D_{\mathrm{g}} L(\mathrm{g}, \zeta).\delta\mathrm{g} + D_{\zeta} L(\mathrm{g}, \zeta).\delta$$

Where $D_{\mathrm{g}} L \in T * G$ indicate the derivative of the Lagrangian L compared to g, given by

$\frac{d}{d\varepsilon}\Big|_{\varepsilon=0} L(\mathrm{g}^{\varepsilon}, \zeta) = D_{\mathrm{g}}L(\mathrm{g}, \zeta).\delta\mathrm{g}$ and $D_{\mathrm{g}}L(\mathrm{g}, \zeta) \in g^*$ is defined in the same way. Since the expression $\mathrm{T}\left(L_{\mathrm{g}} \circ L_{\mathrm{g}^{-1}}\right)$ is equal to the ID card on TGTG, this can be written as:

$$\delta\mathrm{L}(\mathrm{g}, \zeta) =< D_{\mathrm{g}}L(\mathrm{g}, \zeta), \delta\mathrm{g} > +< D_{\zeta}L(\mathrm{g}, \zeta), \delta\zeta >$$

$$=< D_{\mathrm{g}}L(\mathrm{g}, \zeta), \left(TL_{\mathrm{g}} \circ \mathrm{T}L_{\mathrm{g}^{-1}}\right).\delta\mathrm{g} > +< D_{\zeta}L(\mathrm{g}, \zeta), \delta\zeta > \ ,$$

replacing them with equations (II.7), (II.8), we get:

$$\delta\mathrm{L}(\mathrm{g}, \zeta) =< D_{\mathrm{g}}L(\mathrm{g}, \zeta), T_eL_{\mathrm{g}}.\varsigma > +< D_{\zeta}L(\mathrm{g}, \zeta), \dot{\varsigma} + ad_{\zeta}\varsigma >$$

$$=< T_e^*L_{\mathrm{g}}.D_{\mathrm{g}}L(\mathrm{g}, \zeta) + ad_{\zeta}^*.D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > +< D_{\zeta}L(\mathrm{g}, \zeta), \dot{\varsigma} > \qquad \text{(II.9)}$$

By replacing this expression in the variation of the integration (equation (II.6)) and using the integration by part, we get:

$$\delta\zeta = \int_{t_0}^{t_f} < T_e^*L_{\mathrm{g}}.D_{\mathrm{g}}L(\mathrm{g}, \zeta) + ad_{\zeta}^*.D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > +< D_{\zeta}L(\mathrm{g}, \zeta), \dot{\varsigma} > dt$$

$$= < D_{\zeta}L(\mathrm{g}, \zeta), \varsigma >\Big|_{t_0}^{t_f} + \int_{t_0}^{t_f} < T_e^*L_{\mathrm{g}}.D_{\mathrm{g}}L(\mathrm{g}, \zeta) + ad_{\zeta}^*.D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > -< \frac{d}{dt}D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > \mathrm{dt}$$

(II.10)

As $\varsigma(t) = 0$ in order to $t = t_0$ and $t = t_f$ we find :

$$\delta\zeta = \int_{t_0}^{t_f} < T_e^*L_{\mathrm{g}}.D_{\mathrm{g}}L(\mathrm{g}, \zeta) + ad_{\zeta}^*.D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > -< \frac{d}{dt}D_{\zeta}L(\mathrm{g}, \zeta), \varsigma > \mathrm{dt} \qquad \text{(II.11)}$$

By applying the principle of moindre action $\delta\zeta = 0$ for all $\varsigma(t) \in g$, we obtain the Euler-Lagrange equations in the Lie group G:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{dt}}D_{\zeta}L(\mathrm{g}, \zeta) - ad_{\zeta}^*(\mathrm{g}, \zeta) - T_e^*L_{\mathrm{g}}.D_{\mathrm{g}}\mathrm{L}(\mathrm{g}, \zeta) = 0 & (\mathrm{a}) \\ \dot{\mathrm{g}} = \mathrm{g}.\zeta & (b) \end{cases} \qquad \text{(II.12)}$$

If the Lagrangian does not depend on the Lie group G, the third term of equation (II.12 (a)) would disappear and the equation would become similar to the Euler-Poincaré characteristic [40]; This is why this equation is considered as the general form of the Euler-Poincaré characteristic [56].

II.7 Application to 3-dimensional pendulum (Rigid body) on SO (3)

A 3D pendulum is a rigid body supported by a frictionless pivot acting under a gravitational potential. It is a generalization of a planar pendulum and a spherical pendulum, because it has three degrees of freedom of rotation figure II.1. It has been shown that the 3D pendulum can exhibit chaotic irregular attitude dynamics [57].

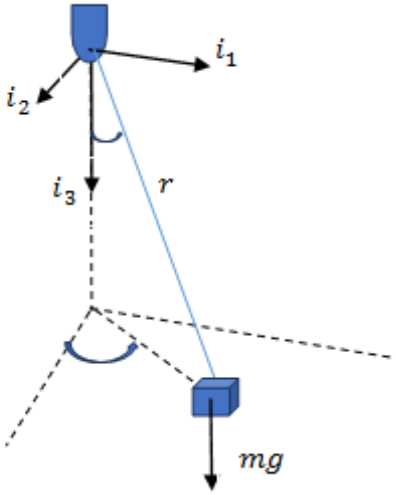


Figure II.1 -Dimensional pendulum mass m.

We assume that the origin of the initial coordinate system is the pivot point, while that of the mobile coordinate system is at the center of gravity of mass m, The attitude of the pendulum is the orientation of the moving landmark (center of mass) with respect to the initial landmark. The variety of configuration is the special orthogonal group

$$SO(3) = \{R \in \mathbb{R}^{3\times3} | R^T R = I, \det(R) = 1\} \tag{II.13}$$

The equation of the attitude kinematics is given by:

$$\dot{R} = R\Omega_X \tag{II.14}$$

Where the angular speed represented in the mobile coordinate system is noted $\Omega \in \mathbb{R}^3$ and operator $\circ_X : \mathbb{R} \to SO(3)$ is isomorphism between $\mathbb{R}^3$ the Lie algebra $SO(3)$ which is the set of 3×3 antisymmetric matrices given by:

$$\Omega_X = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \tag{II.15}$$

For $\Omega = [\Omega_1 \quad \Omega_2 \quad \Omega_3]^T \in \mathbb{R}^3$ The Lie bracket on $so(3)$ corresponds to the cross product on $\mathbb{R}^3$: $[\Omega_X,\ \Omega'_X] = \Omega_X,\ \Omega'_X$ for $\Omega_X,\ \Omega'_X \in \mathbb{R}^3$. Several properties of the hat map are summarized in Appendix A.1. Using the kinematic equation Using, the tangent (fiber) bundle TSO (3) is identified by SO (3) $\times$ so(3) and by SO (3) $\times \mathbb{R}^3$ This defines an internal product in so(3) using the standard internal product over $\mathbb{R}^3$: $\langle \Omega_{X_1}, \Omega_{X_2} \rangle = \frac{1}{2} tr\left[\Omega_{X_1}^T, \Omega_{X_2}\right] = \Omega_1 . \Omega_2$ for $\Omega_1, \Omega_2 \in \mathbb{R}^3$. The interior product of TSO (3) is defined by the left trivialization $\langle X, Y \rangle = \langle T_R\ T_{R^{-1}}.X, T_R\ T_{R^{-1}}.Y \rangle$ for $X, Y \in \mathrm{T}_R\mathrm{SO}$ (3) The $\mathrm{T} * \mathrm{SO}(3)$ cotangent bundle is identified with TSO(3) using this internal product. The adjoint operation ad in SO(3) is given by

$ad_\Omega \Omega' = \Omega_X \Omega', ad^*_\Omega \Omega' = -\Omega_X \Omega'$

The Lagrangian L: SO(3)) $\times$ so(3) $\rightarrow \mathbb{R}$ of the rigid body is the difference between the kinetic energy T: SO(3)) $\times$ so(3) $\rightarrow \mathbb{R}$ and the gravitational potential energy U: SO(3)) $\times$ so(3) $\rightarrow \mathbb{R}$

$$L(R, \Omega) = \frac{1}{2} tr\ [\Omega_X J_d \Omega_X^T] - U(R) \tag{II.16}$$

Where J_d is the non-standard inertia matrix defined by $J_d = \int r . r^T dm$, with $r \in \mathbb{R}^3$ the vector going from the pivot point to the mass represented in the mobile coordinate system, and $U(R) = -mgi_3^T Rr$ is potential energy, J: so(3) $\rightarrow$ so(3)* inertia operator defined by

$J(\Omega_X) = \Omega_X J_d + J_d \Omega_X = (J\Omega)_X$.

The Lagrangian can alternatively be written as:

$$L(R, \Omega) = \frac{1}{2} \langle J_d \Omega_X, \Omega_X \rangle - U(R) \tag{II.17}$$

The derivative of lagrangian

$$D_\Omega L(R, \Omega_X).\delta\Omega_X = \frac{1}{2} tr[\delta\Omega_X J_d \Omega_X + \Omega_X J_d \delta\Omega_X] = \langle (J\Omega)_X, \delta\Omega_X \rangle \tag{II.18}$$

$D_R U(R).\delta R = \frac{d}{d\varepsilon} U(R. exp\varepsilon\varsigma_X)\Big|_{\varepsilon=0} = -tr\left[\varsigma_X R^T \frac{\partial U}{\partial R}\right]$, since

$tr[x_X B] = -tr[x_X B^T]$ for $x \in \mathbb{R}^3$, $B \in \mathbb{R}^{3\times3}$, $\mathrm{tr}[x_X \mathrm{B}] = \frac{1}{2} tr[x_X (B - B^T)] = -\langle B - B^T, x_X \rangle$,

We get :

$$D_R U(R).\delta R = T_e^* L_R . D_R U(R).\varsigma_X = -\langle M_X, \varsigma_X \rangle \tag{II.19}$$

where the moment due to the potential energy $M \in \mathbb{R}^3$ is gravitational moment determined by

$$M_X = \frac{\partial U^T}{\partial R} R - R^T \frac{\partial U^T}{\partial R}; \frac{\partial U}{\partial R} = -mgi_3 r^T \Longrightarrow M = mgR \times R^T i_3 \quad \text{(II.20)}$$

By replacing equations (II.18), (II.19) in (II.20) we get the Euler Lagrange equations of the rigid body in SO (3)

$$\begin{cases} J\dot{\Omega} + \Omega \times J\Omega = \text{mgr} \times R^T i_3 \\ \dot{\text{R}} = \text{R}\Omega_X \end{cases} \quad \text{(III.21)}$$

II.8 Rigid body on SE (3)

The space of the rigid body configuration, this time, is SE (3) = SO (3) R SE (3) = SO (3) R . We assume that there is a fixed initial frame of reference (operator) and a moving frame of reference which has its origin at the center of gravity of body mass m. The position of the body and its attitude which is the orientation of the moving frame (center of mass) with respect to the initial frame are represented by $(r, R) \in SE(3)$. The kinematics equations are given by:

$$\begin{cases} \dot{r} = \text{v} \\ \dot{\text{R}} = \text{R}\Omega_X \end{cases} \quad \text{(II.22)}$$

Where $\text{v} \in \mathbb{R}^3$ translation speed and $\Omega \in \mathbb{R}^3$ is the angular velocity. with the same

manner, where J_d is the non-standard inertia matrix defined by $J_d = \int r.r^T dm$, and

J: $\text{SE}(3) \times \text{se}(3)^* \to \mathbb{R}$ inertia operator defined by $J(\Omega_X) = \Omega_X J_d + J_d \Omega_X = (J\Omega)_X$

The Lagrangian L: $\text{SE}(3) \times \text{se}(3) \to \mathbb{R}$ of the rigid body is the difference between the kinetic energy

T: $\text{SE}(3) \times \text{se}(3) \to \mathbb{R}$ and the gravitational potential energy U: $\text{SE}(3) \to \mathbb{R}$ After calculation, the Lagrangian of the dynamics of the rigid body is given by:

$$L(R, \Omega) = \frac{1}{2} m \|\dot{r}\|^2 + \frac{1}{2} tr[\Omega_X J_d \Omega_X^T] - U(R) \quad \text{(II.23)}$$

Since the configuration space is SE (3), the variations must be carefully chosen in order to respect the geometry of the configuration space. Variation of $r_i, \dot{r}_i$ are :

$$\begin{cases} r_i^\varepsilon = r_i + \varepsilon\delta r_i + O(\varepsilon^2) \\ \dot{r}_i^\varepsilon = \dot{r}_i + \varepsilon\delta\dot{r}_i + O(\varepsilon^2) \end{cases} \quad \text{(II.24)}$$

Where $\delta r_i, \delta \dot{r}_i \in \mathbb{R}^3$ are infinitesimal variations which disappear at the initial instant t_0 and the final time t_f. The infinitesimal variation of a rotation matrix R_i can be expressed in terms of the Lie algebra ηi∈ so (3) $\varsigma_i \in so(3)$ in exponential form:

$$\delta R = \frac{d}{dt}\Big|_{\varepsilon=0} R exp\varepsilon\varsigma = R\varsigma \tag{II.25}$$

The infinitesimal variation of the angular speed is $\delta\Omega_X = -\varsigma\Omega_X + \Omega_X\varsigma + \dot{\varsigma}$; it is in analogy with equation (II.8).

All these variations allow us to write the variation of the integral as follows:

$$\delta\zeta = \int_{t_0}^{t_f} L(\mathrm{g},\zeta)dt = \int_{t_0}^{t_f} -\delta r^T\left\{m\ddot{r} + \frac{\partial U}{\partial r}\right\} + \frac{1}{2}tr[\varsigma\{\left(J\dot{\Omega} + \Omega \times J\Omega\right)_X + 2R^T\frac{\partial U}{\partial R}\}]dt \tag{II.26}$$

By application of the principle of least action and after calculation, the continuous equations of the motion of the rigid body in the group SE (3) are:

$$\begin{cases} \dot{\mathrm{v}} = -\frac{1}{m}\frac{\partial U}{\partial r} \\ J\dot{\Omega} + \Omega \times J\Omega = M \\ \dot{\mathrm{r}} = \mathrm{v} \\ \dot{\mathrm{R}} = R\Omega_X \end{cases} \tag{II.27}$$

II.9 General non-linear theory

considering the standard description for a nonlinear system is

$$\begin{aligned} \dot{x} &= f(x) \\ y &= h(x) \end{aligned} \tag{II.28}$$

The system is said to be in equilibrium or at an equilibrium point When all the state variables are constant. The important part of nonlinear systems is determining the equilibrium point, knowing that one system can contain many different equilibrium points. For this, we will deal with the definition of stability of a certain equilibrium point as follows.

Definition 1 Given *the system (II. 28), assume that* x_e *is an equilibrium point,* $x(0)$ *represents the initial state. Then the equilibrium point is said to be*

- stable, if there for each $\epsilon > 0$ *exists* $\delta(\epsilon) > 0$ *such as*

$$\|x(0) - x_e\| \lessdot \delta \Rightarrow \|x(t) - x_e\| \lessdot \ \epsilon \ for\ all\ t \geq 0 \tag{II .29}$$

- unstable, if not stable
- asymptotically stable, if stable and there exists a $\eta > 0$ such that

$$\|x(0) - x_e\| \lessdot \eta \Rightarrow x(t) \to x_e \ \ ainsi\ que\ t \to \infty \tag{II.30}$$

- globally asymptotically stable if it is asymptotically stable for all initial states

The latter is in most cases a desirable property of the control system. A global asymptotically stable equilibrium point means that all solutions, regardless of the starting point, will converge to the point. This definition is referred to as Lyapunov Stability.

II.9.1 Lyapunov theory

To get the type of stability a certain equilibrium point corresponds to, equation (II.1) must be solved to find x(t). Except that it's hard is not possible to do analytically, nonetheless, the direct method of Lyapunov can be used for proving the stability, this method based on the Lyapunov function V(x) for determines the stability properties from the properties of f(x(t)), The Lyapunov function can be interpreted as a generalized measurement of how far from the equilibrium the system is. If this measurement decreases then the system moves towards the equilibrium point [58]. To summarize this in theory, let define some laws.

Definition .2 A function V (x) is said to be

- positive definite if $V(0) = 0\ and\ V(x) > 0, x \neq 0$
- positive semidefinite if $V(0) = 0\ et\ V(x) \geq 0, x \neq 0$
- negative (semi-)definite if $-V(x)$ is positive (semi-)definite
- radially unbounded if $V(x) \to \infty\ as\ x \to \infty$

Theorem .1 Let $x = 0$ be an equilibrium point for (II.94). Let $V(x)$ be a scalar, continuously differentiable function of the state x such that

- $V(x)$ is positive definite
- $V(x)$is radially unbounded
- $\dot{V}(x) = V_x(x)f(x) \leq -W(x)$ Where $W(x)$ is positive semidefinite and $V_x(x)$ represents the row vector $V_x(x)\left(\frac{\partial V}{\partial x_1}, \ldots \frac{\partial V}{\partial x_n}\right)$

Then all solutions satisfy $lim_{x\to\infty}\ W\big(x(t)\big) = 0$. In addition if $W(x)$ is positive definite then the equilibrium $x = 0$ is globally asymptotically stable.

Proof.

When $\dot{V}(x)$ is negative semidefinite the following theorem can be used to prove stability [59].

Theorem 2. Let $x = 0$ be an equilibrium point for (II.94). Let $V(x)$ be a scalar, continuously differentiable function of the state x such that

- $V(x)$ is positive definite
- $V(x)$ is radially unbounded
- $\dot{V}(x)$ is negative semidefinite

Let $E = \{x: \dot{V}(x) = 0\}$and suppose that no other solution than$x(t) \equiv 0$ can stay forever in E. Then $x = 0$ is globally asymptotically stable.

II.9.2 Lyapunov theory and control design

It was illustrated in the previous section that if a Lyapunov function is found so that its time derivative is negative definite within an area, all trajectories starting within that area will converge to the same equilibrium point. In this part, we will work on designing this property in the case of closed-loop systems is suggested as follow.

$$\dot{x} = f\ (x, u) \tag{II .31}$$

Starting with the selection of the positive Lyapunov function V(x) we work to find the law of control u = K (x) which makes the global system asymptotically stable in a closed loop.

$$\dot{V}(x) = V_x f\ \big(x, k(x)\big) = -W(x) \tag{II.32}$$

Based on theorem 1 given closed-loop stability, with choice W (x) positive define). But it is not clear how V (x) and W (x) should be chosen. in order to it most the following definitions should be mentioned.

Definition .3 (Control Lyapunov Function)

If the Lyapunov function V(x) is positive definite and radially unbounded, $V(x)$ is called a control Lyapunov function (CLF) for (II.6) if for all $x \neq 0$,

$$\dot{V}(x) = V_x f\ (x, u) < 0 \tag{II.33}$$

There is another theory developed to make CLF meaningful, known as Artstein's theory where says that the existence of a CLF is equivalent to the existence of a control law which will make the desired state global asymptotically stable [60].

II.10 Integrator backstepping

A short design example is taken into consideration, to illustrate how to find CLF law and control law, so we suggest the system shown below [22].

$$\dot{x} = f(x) + g(x)\xi$$
$$\dot{\xi} = a(x, \xi) + b(x, \xi)u \tag{II.34}$$

Where $x \in \mathbb{R}^n$ and $\xi \in \mathbb{R}$ are state variables and $u \in \mathbb{R}$ a control input.

To make subsystem x a global asymptotically stable, it suffices optimal selection for ξ, where this last is the controller for the subsystem x entry, addition, it determines $\xi_{ref}(x)$ as virtual control law for the x -subsystem, Next, we choose $\xi_{ref}(x)$ so that the time derivative of the Lyapunov function $V_1(x)$ is negative.

$$\dot{V}_1(x) = V_{1x}(x)\Big(f(x) + g(x)\xi_{ref}(x)\Big) < 0\ ,\ x \neq 0 \tag{II .35}$$

System (II.35) is the result of writing the previous system (II .34) in a new state is introduced which represents the residual $\tilde{\xi} = \xi - \xi_{ref}(x)$.

$$\dot{x} = f(x) + g(x)(\tilde{\xi} - \xi_{ref}(x))$$

$$\dot{\xi} = a\left(x, (\tilde{\xi} + \xi_{ref}(x)\right) + b\left(x, \tilde{\xi} + \xi_{ref}(x)\right)u - \frac{\partial \xi_{ref}(x)}{\partial x}(f(x) + g(x)(\tilde{\xi} - \xi_{ref}(x))) \quad \text{(II.36)}$$

Besides, that from $V_1(x)$ and adding a quadratic term to get the second Lyapunov function.

$$V_2(x, \tilde{\xi}) = V_1(x) + \frac{1}{2}\tilde{\xi}^2 \quad \text{(II .37)}$$

Differentiating $V_2(x, \tilde{\xi})$ time yields

$$\dot{V}_2(x, \tilde{\xi}) = V_{1x}(x)\left(f(x) + g(x)\xi_{ref}(x) + g(x)\tilde{\xi}\right) + \tilde{\xi}\left(a\left(x, \tilde{\xi} + \xi_{ref}(x)\right)\right) +$$

$$\tilde{\xi}(b(\left(x, \tilde{\xi} + \xi_{ref}(x)\right)u - \frac{\partial \xi_{ref}(x)}{\partial x}\left(f(x) + g(x)\left(\tilde{\xi} + \xi_{ref}(x)\right)\right)) \quad \text{(III.38)}$$

By eliminating the variables that the functions depend on, the equation (II.103) becomes as follows

$$\dot{V}_2(x, \tilde{\xi}) = V_{1x}(f + g\xi_{ref}) +$$

$$\tilde{\xi}\left(V_{1x}g + a + bu - \frac{\partial \xi_{ref}(x)}{\partial x}\left(f + g(\tilde{\xi} + \xi_{ref})\right)\right) \quad \text{(II.39)}$$

Finally, define the control u in Equation (II.40),in order to ensure the stability of the system, and this is by making $\dot{V}_2$ be negative define.

$$u = \frac{1}{b}(\frac{\partial \xi_{ref}(x)}{\partial x}\left(f + g(\tilde{\xi} + \xi_{ref})\right) - a - V_1 g - k\tilde{\xi}), \quad k > 0 \quad \text{(II.40)}$$

Then $\dot{V}_2$ becomes

$$\dot{V}_2 = V(f + g\xi_{ref}) - k\tilde{\xi}^2 \leq 0 \quad \text{(II.41)}$$

The system can be further expanded by starting over again. when u is not the actual control input but a virtual control law consisting of state variables, we have to repeat these steps until we find the law of control, and this is what distinguishes backstepping control.

II.10.1 system form for applied backstepping

The system to be controlled by backstepping control must have what is called a lower triangular shape. equation 4 The general form shows this system.

$$\begin{aligned}
\dot{x}_1 &= f(x_1) + g(x)x_1 \\
\dot{x}_1 &= f_1(x_1, x_2) + g_1(x, x_2)x_2 \\
&\vdots \\
\dot{x}_i &= f_1(x_1, x_2, \ldots, x_i) + g_1(x_1, x_2, \ldots, x_i)x_{i+1} \\
&\vdots \\
\dot{x}_m &= f_1(x_1, x_2, \ldots, x_m) + g_1(x_1, x_2, \ldots, x_m)\xi_1 \\
\dot{\xi}_1 &= u
\end{aligned} \tag{II.42}$$

Example:

Consider the system as illustrate in figure (II.43) [61]

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1)^T.\theta_1 + \xi_1 \\
\dot{\xi}_1 &= u \\
y &= x_1
\end{aligned} \tag{II.43}$$

Where θ_1, $f_1(x_1)^T$ and y are parametric vector and Nonlinear function vector and output of system respectivly.

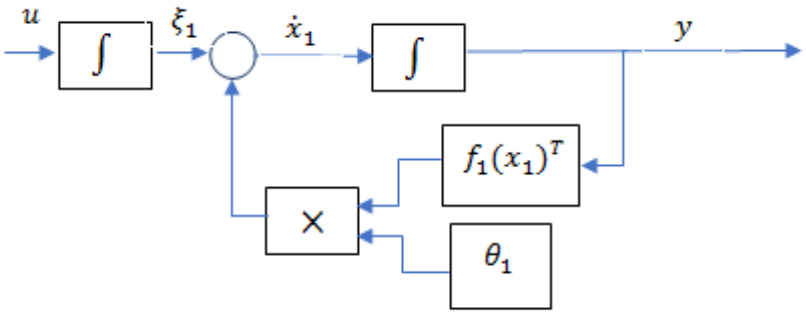


Figure II.2: block diagram of the second order system

The objective of the command is to reach the convergence of the errors towards zero thus achieving the stability and the equilibrium of the system which allows its output y to follow a reference y_{ref}.

Step 1:

We define the 1 st tracking error as:

$$e_1 = x_1 - y_{ref} \tag{II.44}$$

And the following stabilizing function chooses so as to eliminate the nonlinearity of the subsystem:

$$\begin{aligned}
\alpha_1(x) &= -k_1 e_1 - f_1(x_1)^T.\theta_1 \\
&= -k_1(x_1 - y_{ref}) - f_1(x_1)^T.\theta_1
\end{aligned} \tag{II .45}$$

k_1 is positive constant, this solution is designed to stabilize the first subsystem and since this is not the case, we define the second error by:

$$e_2 = \xi_2 - \alpha_1(x) - \dot{y}_{ref} \tag{II.46}$$

e_2 is the variable which expresses the reality that ξ_2 is not the exact command

We choose the first Lyapunov function:

$$V_1 = \frac{1}{2}e_1^2 \tag{II.47}$$

Its derivative is:

$$\dot{V}_1 = -k_1 e_1^2 + e_2 e_1 \tag{II.48}$$

The term $e_2 e_1$ will be eliminated in the next step where the second subsystem will be approached.

Step 2

The derivative of e_2 is expressed by:

$$\begin{aligned} \dot{e}_2 &= \dot{\xi}_1 - \dot{\alpha}_1(x) - \ddot{y}_{ref} \\ &= \dot{\xi}_1 - \frac{\partial \alpha_1}{\partial x_1}\dot{x}_1 - \frac{\partial \alpha_1}{\partial y_{ref}}\dot{y}_{ref} - \ddot{y}_{ref} \\ &= u - \frac{\partial \alpha_1}{\partial x_1}(f_1(x_1)^T.\theta_1 + \xi_1) - \frac{\partial \alpha_1}{\partial y_{ref}}\dot{y}_{ref} - \ddot{y}_{ref} \end{aligned} \tag{II.49}$$

Consider the second Lyapunov function:

$$V_2 = \frac{1}{2}e_1^2 + \frac{1}{2}e_1^2 \tag{II.50}$$

The derivative of V_2 is:

$$\begin{aligned} \dot{V}_2 &= e_1\dot{e}_1 + e_2\dot{e}_2 \\ &= -k_1 e_1^2 + e_2\left(u + e_1 - \frac{\partial \alpha_1}{\partial x_1}(f_1(x_1)^T.\theta_1 + \xi_1) - \frac{\partial \alpha_1}{\partial y_{ref}}\dot{y}_{ref} - \ddot{y}_{ref}\right) \end{aligned} \tag{II .51}$$

The command u is chosen such that $\dot{V}_2 < 0$ (Lyapunov stability condition).

We obtain :

$$\begin{aligned} u &= -k_2 e_2 - e_1 + \frac{\partial \alpha_1}{\partial x_1}(f_1(x_1)^T.\theta_1 + \xi_1) + \frac{\partial \alpha_1}{\partial y_{ref}}\dot{y}_{ref} + \ddot{y}_{ref} \\ &= -k_2 e_2 - e_1 + \frac{\partial \alpha_1}{\partial x_1}\dot{x}_1 + \frac{\partial \alpha_1}{\partial y_{ref}}\dot{y}_{ref} + \ddot{y}_{ref} \end{aligned} \tag{II .52}$$

k_2 is positive constant, In these conditions: $\dot{V}_2 = -k_1 e_1^2 + -k_2 e_2^2 < 0$. This allows for the equilibrium and the system is stable.

Equation (II.49) takes the form:

$$\dot{e}_2 = -e_1 - k_2 e_2 \tag{II.53}$$

The closed-loop system (II.43) can be formulated using the new coordinates at e:

$$e = K\dot{e} \tag{II.54}$$

Where :

$K = \begin{bmatrix} k_1 & 1 \\ 1 & k_2 \end{bmatrix}$, and $e = [e_1 \quad e_2]^T$

The solution is given by:

$$e = e(0)\exp(Kt) \tag{II.55}$$

With a good choice of the gains k_1 and k_2 (pole placement method), the solution given by (II.52) indeed gives a convergence of the errors towards zero, so the output of the system y follows the reference y_{ref}.

II.10.2 General design procedure

In this part, we will try to generalize the application of the Backstepping approach for systems of order n:

$$\begin{gathered} \dot{x}_1 = f(x_1)^T.\theta_1 + x_2 \\ \dot{x}_2 = f(x_1, x_2)^T.\theta_2 + x_3 \\ \vdots \\ \dot{x}_n = f(x_1, x_2, \dots, x_1, x_n)^T.\theta_n + g(x_1, x_2, \dots, x_1, x_n)\xi \\ \dot{\xi} = u \end{gathered} \tag{II.53}$$

With $g(x_1, x_2, \dots, x_1, x_n) \neq 0$

In general, the design, by the backstepping principle, of the control law u is carried out in n steps. In the i^{Th} step, an i^{Th} order subsystem is stabilized with respect to a Lyapunov function V_i by the design of a stabilizing function i. The control law is then established in the final step [62].

By convention, we define: $e_0 = 0, \alpha_0 = 0, e_1 = x_1 - y_{ref}$

$e_i = x_i - \alpha_{i-1} - y_{ref}^{(i-1)}$

$\alpha_i = -e_i - k_i e_i - B_i$

Where :

$B_1 = w_1$; $B_i = w_i - \sum_{j=1}^{i-1} \frac{\partial y \alpha_{i-1}}{\partial x_j} x_{j+1} - \sum_{j=1}^{i-1} \frac{\partial y \alpha_{i-1}}{\partial y_{ref}^{(j-1)}} y_{ref}^{(j)}$, $\quad 2 < i < n$

$w_1 = f(x_1)^T.\theta_1$, $w_i = f(x_i)^T.\theta_i - \sum_{j=1}^{i-1} \frac{\partial y \alpha_{i-1}}{\partial x_j} (f(x_j)^T.\theta_j)$, $\quad 2 \leq i \leq n$

$u = \frac{1}{g}(\alpha_n - y_{ref}^{(n)})$

$$\dot{e} = Ae$$

$$\text{With : } A = \begin{bmatrix} -k_1 & 1 & 0 & \cdots & \dots & 0 \\ 1 & -k_2 & 1 & 0 & \dots & 0 \\ 0 & 1 & -k_2 & 1 & \cdots & 0 \\ 0 & 0 & 1 & -k_4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -k_n \end{bmatrix}, e = [e_1 \quad e_2 \quad e_3 \quad e_4 \quad \cdots \quad e_1]^T$$

$$e = e(0)\exp(At)$$

II.1 conclusion

In this chapter, we detailed geometric methods for the modeling quadrotor by giving the different tools of this analysis. and we mentioned backstepping control synthesis, where this latter is a relatively recent control method for nonlinear systems. It allows sequentially and systematically to determine the control law of the system by the choice of a Lyapunov function. Its principle is to know how to constructively establish the control law of the nonlinear system by considering certain state variables as virtual commands and conceive them as intermediate control laws.

Chapter III
Geometric Backstepping Control of quadrotor

III.1 Introduction

In this chapter we try to applied geometric backstepping control method developed directly on the quadrotor to avoid singularities and ambiguities inherent to other attitude representations. in order to command the position and altitude for quadrotor, we used geometric model of the quadcopter in SO (3).

III.2 Quadrotor Dynamic Model

As illustrated in Figure III.1, a quadrotor vehicle model is a system of four identical motors and propellers that are located at the vertices of a square and generated thrust and torque normal to the plane of this square. Then, an inertial reference frame $\{\vec{\imath_1}, \vec{\imath_2}, \vec{\imath_3}\}$ and a body-fixed frame $\{\vec{b_1}, \vec{b_2}, \vec{b_3}\}$ were chosen. Also, the body-fixed frame origin is located at the center of the mass of this vehicle.

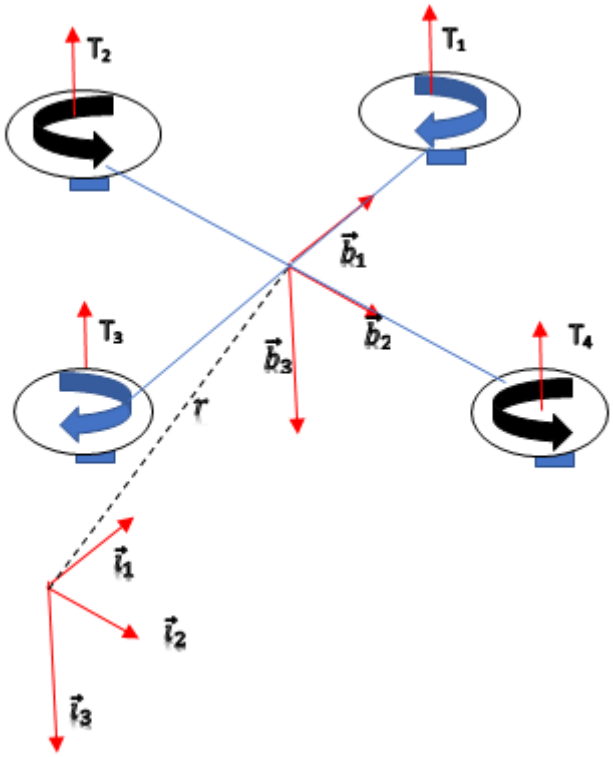


Figure III.1. Quadrotor dynamique

The model is represented by the rotation matrix $R \in SO(3)$ [63, 64]. Consequently, the configuration manifold is the special Euclidean group SE(3), which is the semi-direct product of $\mathbb{R}^3$ and the special orthogonal group SO(3). The rotation matrix R is given by the product of the three successive matrices: $R = Rot_z(\psi) * Rot_y(\theta) * Rot_x(\theta)$, where $Rot_z(\psi)$, $Rot_y(\theta)$ and $Rot_x(\theta)$ are the partial rotation matrices around $\vec{b_3}$, $\vec{b_2}$ and $\vec{b_1}$. where $SO(3) =$

$\{R \in R^{3\times3}: R^T = R^{-1} \wedge \det(R) = 1\}$ [65, 66]. the representation of the rotation matrix is depicted in (III.1) as follow:

$$R = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ c\theta s\psi & s\psi s\theta s\phi - c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad \text{(III.1)}$$

We note that $s\phi$ and $c\phi$ denote $\sin(\phi)$ and $\cos(\phi)$ respectively and similarly for θ and ψ. The dynamical model of the quadrotor can be described as :

$$\begin{cases} m\ddot{r} = mgi_3 - TRi_3 + F_{ext} & (III.a) \\ \dot{R} = R\Omega_\times & (III.b) \\ J\dot{\Omega} + \Omega \times J\Omega = M + M_{ext} & (III.c) \end{cases}$$

Where the m is a mass and J inertia matrix, r is the quadrotor position in the inertial frame, Ω is the angular velocity in the body frame, F_{ext} and M_{ext} are external force and moment, respectively.

$\circ_\times : \mathbb{R}^3 \to SO(3)$ is defined by the condition that $a_\times b = a \times b$ for all $a, b \in \mathbb{R}^3$

The Euler equation determines the angular acceleration:

$$\begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} \dot{\Omega}_1 \\ \dot{\Omega}_2 \\ \dot{\Omega}_3 \end{bmatrix} = \begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} - \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} \times \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} \quad \text{(III.3)}$$

Each rotor of the quadrotor has an angular velocity ω_j to define the trust and the drag forces $T_j = k_T\omega_j^2$ and $\tau_j = k_m\omega_j^2$ respectively, where the k_T and k_m are the thrust and drag coefficients. The equation for mapping of the roll, pitch, and yaw direction can be written as:

$$\begin{bmatrix} T \\ M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -l & 0 & l \\ l & 0 & -l & 0 \\ -c_\tau & c_\tau & -c_\tau & c_\tau \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} \quad \text{(III.4)}$$

If we take into account the speeds of the attached propellers to the drivers (motors), the last equation becomes:

$$\begin{bmatrix} T \\ M_1 \\ M_2 \\ M_3 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -bl & 0 & bl \\ bl & 0 & -bl & 0 \\ -bc_\tau & bc_\tau & -bc_\tau & bc_\tau \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad \text{(III.5)}$$

III.3 Control Design

After developing the dynamic model of the vehicle in the SE(3), we present our strategy of backstepping control. Thus, the moment vector and the thrust magnitude are the control inputs of our system[67].

The structure of the geometric control law is shown in fig.2.

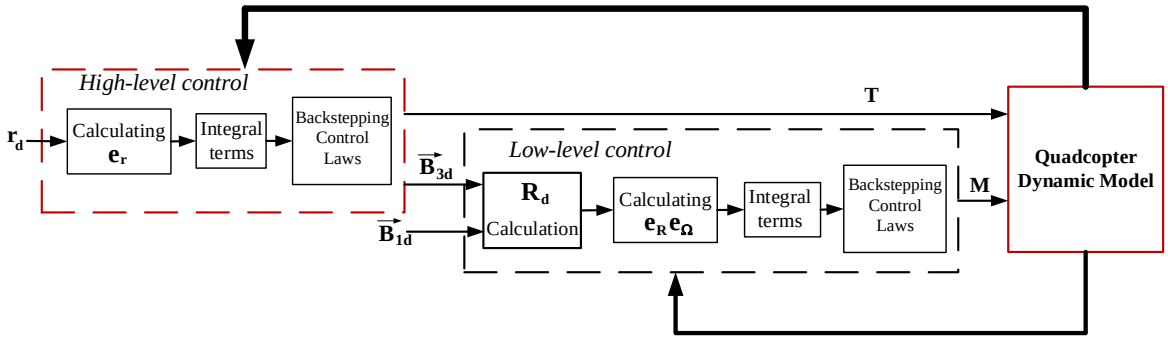


Figure III.2. Bloc Schematic of the control

Where $T \in \mathbb{R}$ associated with the desired position of the quadrotor r_d, the moment control is associated with the desired direction of the first body-fixed frame $\vec{b}_{1d}$ and the desired direction of the third body-fixed frame $\vec{b}_{3d}$ [68, 69]. Thus, the desired position is r_d, wherein this case, we can obtain the corresponding desired attitude R_d:

$$R_d = [\vec{R}_{1d}, \vec{R}_{2d}, \vec{R}_{3d}] \in SO(3) \tag{III.6}$$

Where $\vec{R}_{1d} = \vec{b}_{2d} \times \vec{b}_{3d}$, $\vec{R}_{2d} = \vec{b}_{2d} = \frac{\vec{b}_{3d} \times \vec{b}_{1d}}{\|\vec{b}_{3d} \times \vec{b}_{1d}\|}$ and $R_{3d} \overrightarrow{=} \vec{b}_{3d}$

From the attitude kinematics represented by (III.a) and (III.6), we obtain angular velocity command $\Omega_d \in \mathbb{R}^3$.

III.3.1 Control of position:

In this part, we use backstepping control to deduce the thrust magnitude $T \in \mathbb{R}$ and the desired direction of the third body-fixed frame $\vec{b}_{3d}$. Let's define the position error as $e_1 = r - r_d$ and its derivative $\dot{e}_1 = \dot{r} - \dot{r}_d$. Then, designing the ξ to be as the virtual input and $e_2 = \dot{r} - \xi$ to be the virtual control error, we select the Lyapunov function V_1 and its derivative as follow:

$$V_1 = \frac{1}{2} e_1^T I e_1 \tag{III.7}$$

$$\dot{V}_1 = e_1^T I \dot{e}_1 = e_1^T I(\dot{r} - \dot{r}_d) = e_1^T I(e_2 + \xi - \dot{r}_d) \tag{III.8}$$

To make e_1 progressive stable, the Eq. (III.8) needs to content $\dot{V}_1 < 0$. Then, we select $\xi = \dot{r} - k_1 e_1$, where k_1 is a positive constant.

$$\dot{V}_1 = e_1^T I(e_2 + \dot{r}_d - k_1 e_1 - \dot{r}_d) = e_1^T I e_2 - k_1 e_1^T I e_1 \tag{III.9}$$

Where $e_2 = \dot{r} - \xi = \dot{r} - \dot{r}_d + k_1 e_1$. Because of e_2 is not generally a zero, so the second step of backstepping is given by V_2 and its derivative:

$$V_2 = \frac{1}{2} e_1^T I e_1 + \frac{1}{2} e_2^T I e_2 \tag{III.10}$$

$$\begin{aligned} \dot{V}_2 &= e_2^T I \dot{e}_2 + e_1^T I \dot{e}_1 = e_2^T I(\ddot{r} - \ddot{r}_d + k_1 \dot{e}_1) + \dot{V}_1 \\ &= e_2^T I(\ddot{r} - \ddot{r}_d + k_1 \dot{e}_1) - k_1 e_1^T I e_1 + e_1^T I e_2 \\ &= e_2^T \left(g i_3 - \frac{1}{m} T R i_3 - \ddot{r}_d + k_1 \dot{e}_1 \right) - k_1 e_1^T I e_1 + e_1^T I e_2 \end{aligned} \tag{III.11}$$

Then, the control law T is designed as follow:

$$T = (-mg i_3 + m\ddot{r}_d + (k_1^2 - 1)e_1 + (k_1 + k_2)e_2) R i_3 \tag{III.12}$$

Where k_1 and k_2 are positive constants. The Lyapunov function's derivative is negative [65, 68, 70, 71], so in this time, we say that global stability is according and it can control the position r to be progressive stability by the control T. After the simulation is done, it becomes clear that we need to add integration, the control T became:

$$T = \begin{pmatrix} mg i_3 - m\ddot{r}_d + k_i \int e_1 + (k_1^2 - 1)e_1 \\ +(k_1 + k_2)e_2 \end{pmatrix} R i_3 \tag{III.13}$$

III.3.2 Control of attitude

The backstepping geometric controller uses an error function while using the backstepping control approach to improve its robustness. W*e* notice that it is not possible to compare directly between R and R_d due to the tangent vector of $\dot{R}$ and $\dot{R_d}$ which are in different tangent spaces [71-74].

Eq. (III.14) defines the attitude error, and Eq. (III.15) defines the Angular Velocity error:

$$e_R = \frac{1}{2}(R_d^T R - R^T R_d)^v \tag{III.14}$$

Where $\circ^v$ is vee map ($\circ^v : SO(3) \rightarrow \mathbb{R}^3$ is the inverse of $\circ_\times$ which was defined before.

$$e_\Omega = \Omega - R^T R_d \Omega_d \tag{III.15}$$

$$\Omega_{xd} = R^T \dot{R}_d \ \rightarrow \Omega_d$$

However, before obtaining the attitude controller dynamics, we need to know $\dot{e}_R$ and $\dot{e}_\Omega$, which are defined by (III .16) -(III.17) .

$$\dot{e}_R = \frac{1}{2}(tr(R_d^T R)^T I])e_\Omega = De_\Omega \tag{III.16}$$

$$\dot{e}_\Omega = \dot{\Omega} + (\Omega_x R_d^T R - R_d^T R \dot{\Omega}_d) \tag{III.17}$$

The matrix *D* defined in eq (III.17) is invertible when the rotation angle between *R* and *Rd* is less than 180° [71, 73].

To control the position of our quadrotor and the attitude implicitly, we use the following positive Lyapunov function:

$$V_3 = \frac{1}{2} e_R^T I e_R + \frac{1}{2} e_\Omega^T J e_\Omega \tag{III.18}$$

The derivative of the Lyapunov function is given as:

$$\dot{V}_3 = e_\Omega^T J \left(\dot{\Omega} + \left(\Omega_x R_d^T R - R_d^T R \dot{\Omega}_d\right)\right) + e_R^T I \dot{e}_R \tag{III.19}$$

$$\dot{V}_3 = e_\Omega^T \left((M - \Omega \times J\Omega) + J\left(\Omega_x R_d^T R - R_d^T R \dot{\Omega}_d\right)\right) + e_R^T I \dot{e}_R \tag{III.20}$$

The control law becomes:

$$M = -k_3 e_R - k_5 e_\Omega + \Omega \times J\Omega - \mathrm{J}\left(\Omega_\times R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d\right) \tag{III.21}$$

Concerning the attitude error, we add an integrator controller to the control law M:

$$M = -k_3 e_R - k_4 \int e_R dt - k_5 e_\Omega$$

$$+\Omega \times J\Omega - \mathrm{J}\left(\Omega_\times R^T R_d \Omega_d - R^T R_d \dot{\Omega}_d\right) \tag{III.22}$$

Where k_3, k_4 and k_5 are positive constants, which can be calculated approximately as demonstrated in the stability analysis section as follow:

Among the properties of these errors, we quote:

$$\frac{\|\dot{e}_R\|}{2} \leq \rho(R, R_d) \leq \frac{1}{\alpha - 1} \|e_R^2\| \tag{III.23}$$

The time derive of $\rho(R, R_d)$ is given by

$$\dot{\rho}(R, R_d) = e_R e_\Omega \tag{III.24}$$

As presented in appendix B, we have:

$$\|\dot{e}_R\| < \|e_\Omega\| \tag{III.25}$$

According to the relation of Eq. (III.25), we can write:

$$z_1^T M_{11} z_1 + k_4 \|\int e_R dt\| . \left\| e_{\mathrm{R}} + \frac{1}{2}\int e_R dt \right\| \leq V \leq z_1^T M_{12} z_1 + k_4 \|\int e_R dt\| . \left\| e_{\mathrm{R}} + \frac{1}{2}\int e_R dt \right\| \tag{III.26}$$

Where $z_1^T = [\|e_{\mathrm{R}}\| \quad \|e_\Omega\|]$, $M_{11} = \frac{1}{2}\begin{bmatrix} k_3 + k_4 & \lambda_M(J) \\ \lambda_M(J) & \lambda_m(J) \end{bmatrix}$,

$M_{12} = \frac{1}{2}\begin{bmatrix} \frac{2(k_3+k_4)}{2-\alpha} & \lambda_M(J) \\ \lambda_M(J) & \lambda_M(J) \end{bmatrix}$, $\lambda_m(J)$ and $\lambda_M(J)$ represent the minimum and maximum eigenvalues of the matrix J, respectively.

Even to be the derive of Lyapunov function is a negative define should check that relation:

$$\dot{V} < -z_1^T L_1 z_1 \tag{III.27}$$

Where:

$$L_1 = \begin{bmatrix} (k_3 - k_4 & 0 \\ 0 & k_3 - \lambda_m(J) \end{bmatrix} \tag{III.28}$$

The matrices M_{11}, $M_{11}$and L_1 must be defined positive with the following conditions:

$$\begin{cases} k_4 < k_3 \\ k_3 < \min\{\frac{\lambda_M^2(J)}{\lambda_m(J)}, \frac{\lambda_M(J)}{2}(\alpha - 2)\} \\ k_5 > Max\{\frac{\lambda_M^2(J)}{\lambda_m(J)} - k_3, \frac{\lambda_M(J)}{2}(\alpha - 2) - k_3, \lambda_M(J)\} \end{cases} \tag{III.29}$$

The formula of $\vec{b}_{3d}$ became [10]

$$\vec{b}_{3d} = -\frac{\begin{matrix}(mgi_3 - m\ddot{r}_d + k_i \int e_1 + (k_1^2 - 1)e_1 + \\ (k_1 + k_2)e_2)\end{matrix}}{\left\| \begin{matrix} mgi_3 - m\ddot{r}_d - k_i \int e_1 - (k_1^2 - 1)e_1 - \\ (k_1 + k_2)e_2 \end{matrix} \right\|} \tag{III.30}$$

where we assume that

$$mgi_3 - m\ddot{r}_d - k_i \int e_1 - (k_1^2 - 1)e_1 - (k_1 + k_2)e_2 \neq 0 \qquad \text{(III.31)}$$

The controllers are based on a Backstepping geometric method. This control is a cascade where we used control backstepping for the position, and attitude tracking with the need for the tow desired values r_d and b_{1d}.

III.4 Simulation Results

In this section, we define the quadrotor's parameter values, as shown in Table 1. Then, we specify the desired input.

Table III.1 The parameters of quadrotor UAV

Point Size	Parameters	Values
Mass	m	4.34 kg
thrust coefficient	b	$3.3.10^{-5}$ N/rad/s
drag coefficient	c_τ	0.0226 m
longer	l	0.314 m
Inertia x	j_{xx}	0.082 kgm^2
Inertia y	j_{yy}	0.082 kgm^2
Inertia z	j_{zz}	0.1377 kgm^2
Maximum speed	*Max.rpm*	1300 rpm

The initial hovering conditions are: $r(0) = \Omega(0) = [0,0,0]^T$, $R(0) = I_{3\times3}$

The desired trajectory $r_d = [r_{1d},\ r_{2d}, r_{3d}]^T$ varies as follows:

First, the quadrotor takes off vertically to 15 m. Then, the desired direction of the first body-fixed frame $\vec{b}_{1d}$ takes 5 m at t = 2s, while at time equal 4s the quadrotor's coordinate becomes [5, 10, -15]. At t=8s, the desired direction of the second body-fixed frame $\vec{b}_{2d}$ decreases to 3m and the $\vec{b}_{1d}$ decreases to 8 m at t=10s.

In the second case, we design robust control by changing the inertia by 100% in the initial $J = 1.5 * [j_{xx}, j_{yy}, j_{zz}]$ and $m = 2 * m$.In the third section, we apply an external force that we consider, such as wind gust $F_{ex} = [20,\ 30, 0]N$. The first force (20) at time $t \geq 2s$ and the second force (30) at $t \geq 6s$.

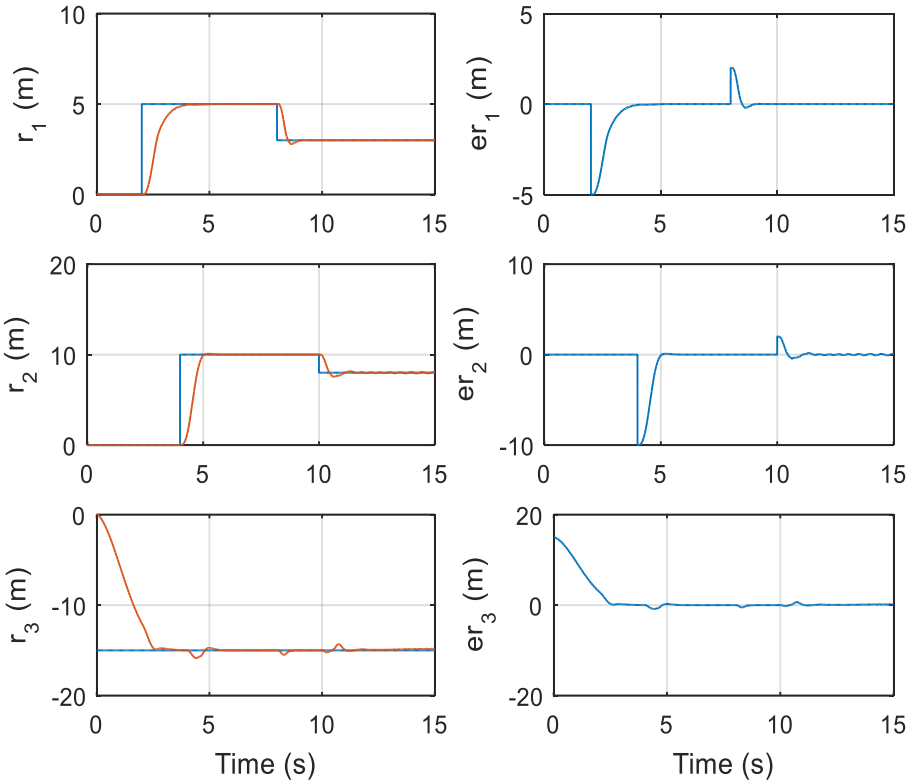
In figure III. 3.a, we show the simulation results of position and its error for quadrotor we observe actual value follow reference value in good time r_1 and r_2 about 1 second but r_3 about 2.5 seconds.

In figure III. 3.b, we show the control input simulation results where thurst force and moment of the roll, pitch, and yaw.

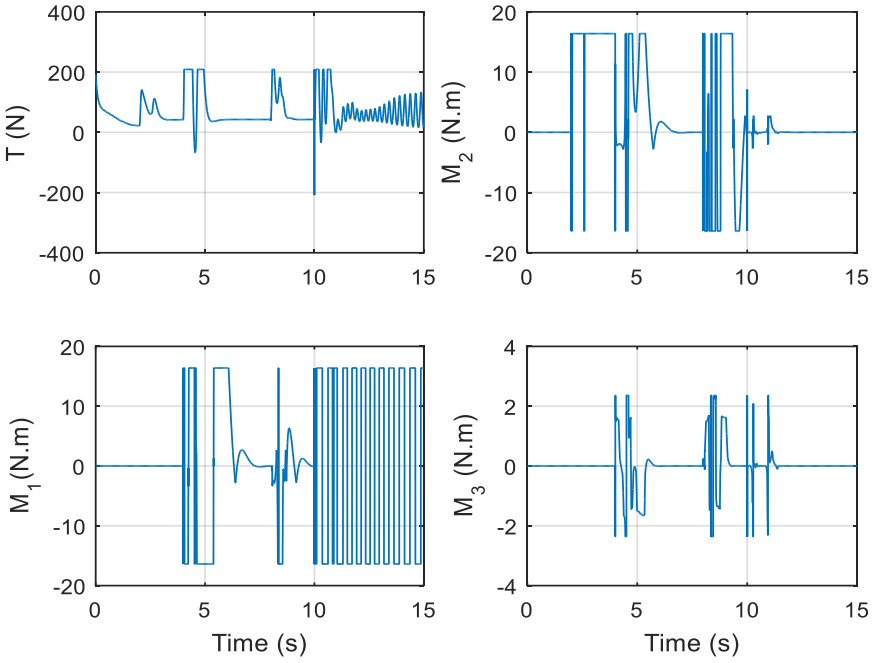
Figure III. 4. shows the phase plan of Euler angles where we notice that they are stable.

In figure III . 5. and figure III. 6. we show the simulation results of the study of robustness when changing the parameters and applying to apply an external force F_{ex}.

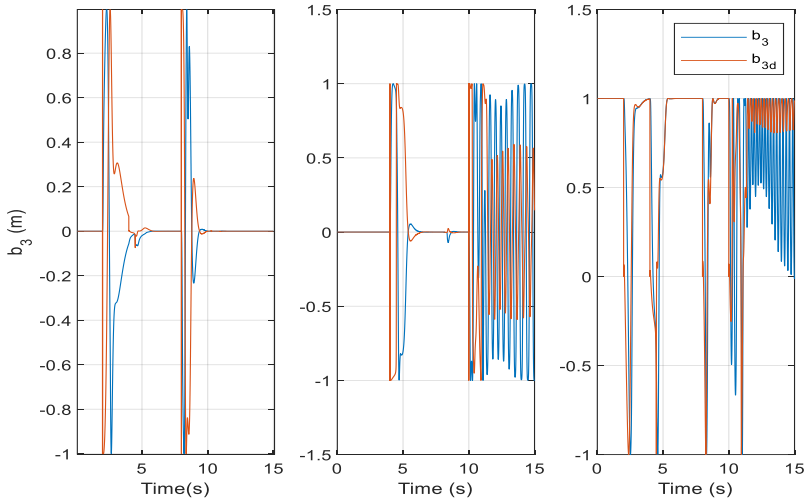
Finally, in figure III. 7.a and figure III. 7.b trajectory without an external force, then trajectory with an external force.



a. Position and its error



b- Thrust and different moments



c- The desired direction of the third body-fixed frame $\vec{b}_{3d}$

Figure III.3 Flipping using backstepping control Without changing the parameters

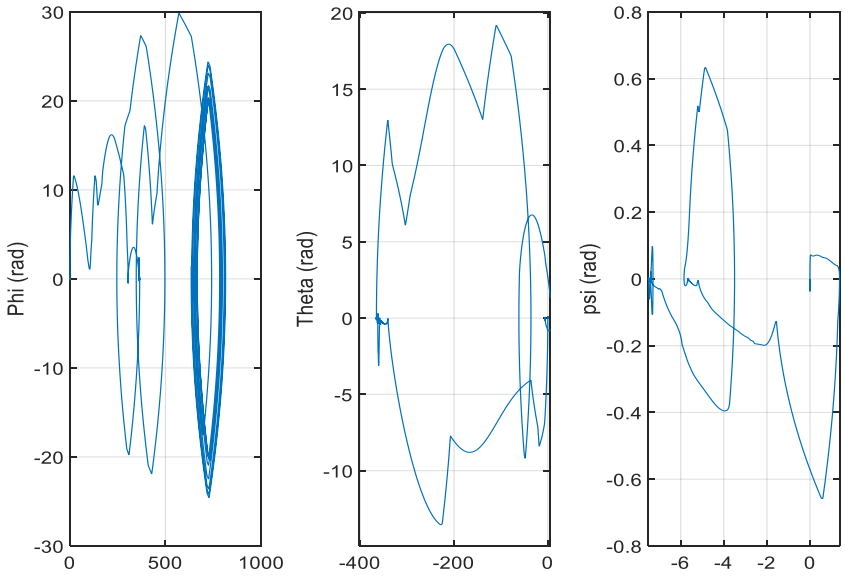
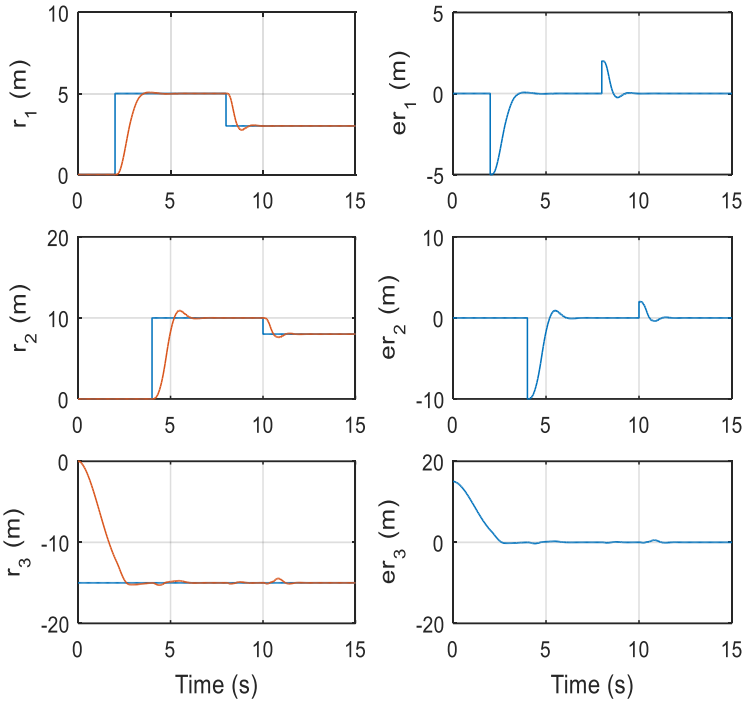
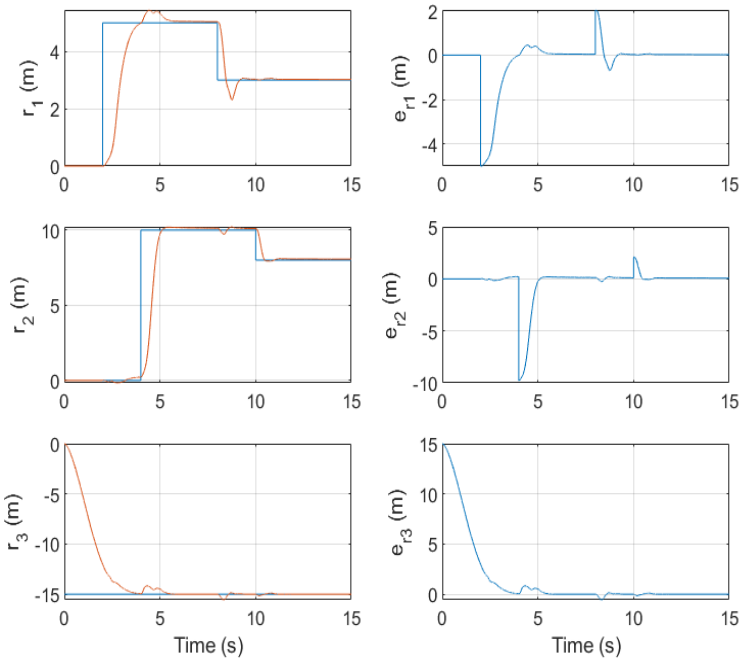


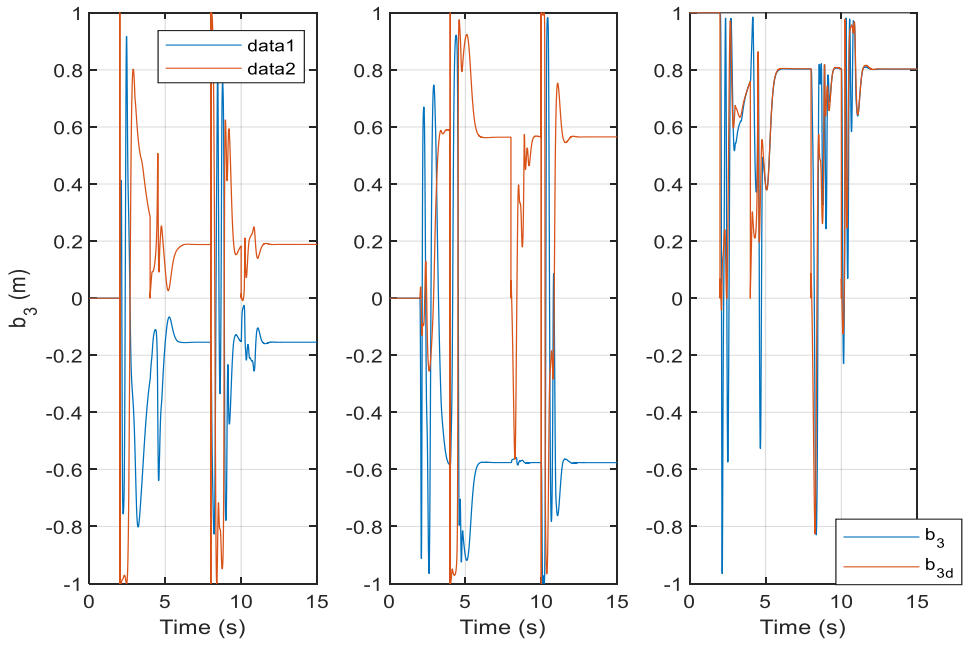
Figure III. 4 Phase plan of Euler anglers



FigureIII.5 Flipping using backstepping control with changing the parameters

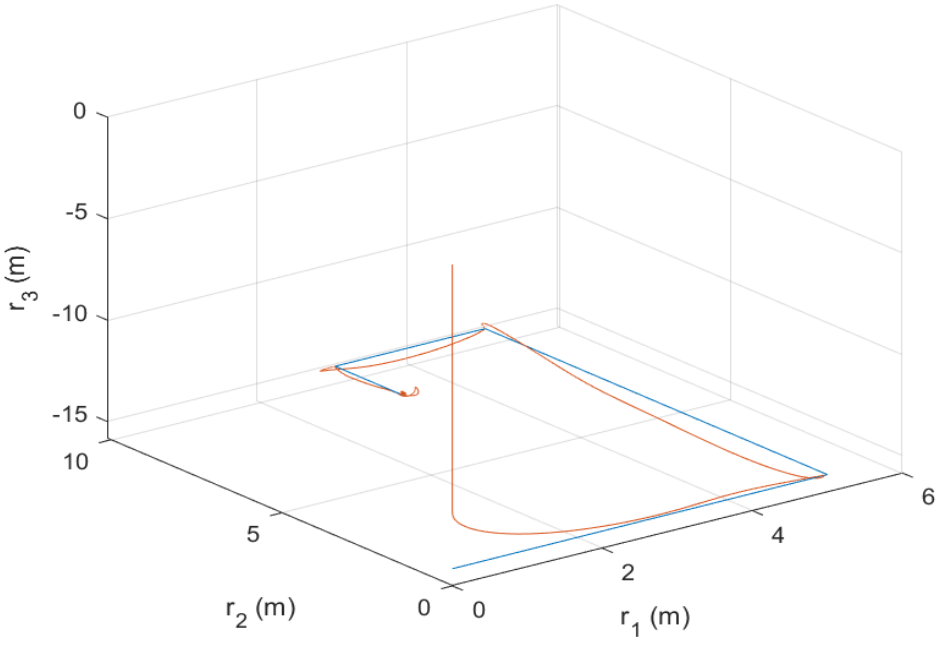


a. Position and its error

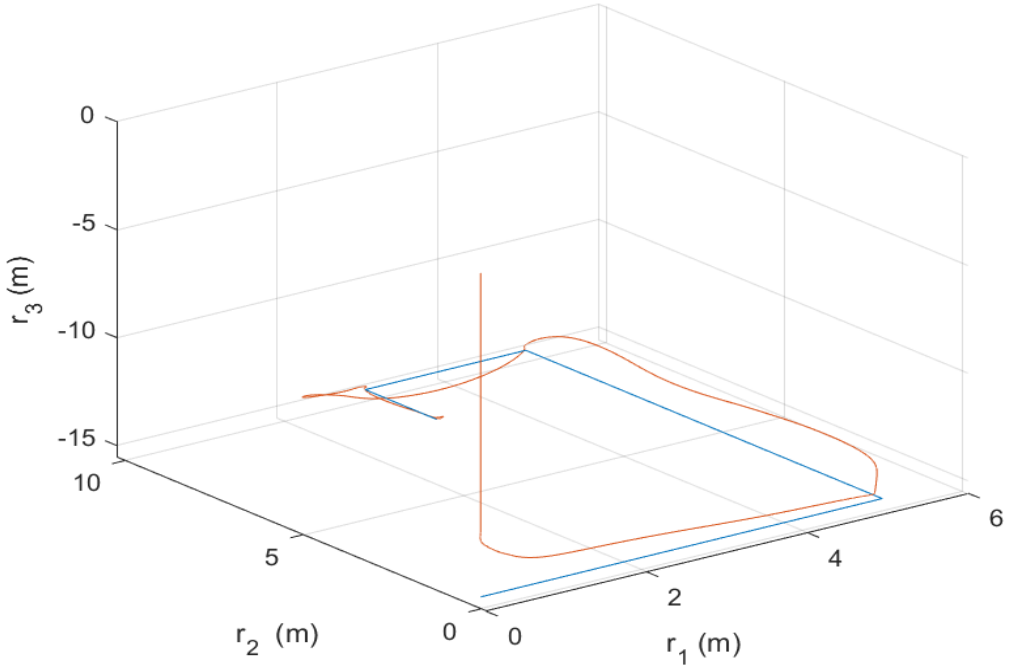


b. The desired direction of the third body-fixed frame $\vec{b}_{3d}$

Figure III. 6. Flipping using backstepping control with we apply an external force F_{ex}.



a- Trajectory without an external force



b- Trajectory with an external force

Figure III.7 Global trajectory of the quadrotor in 3 Dimensions

Simulation results of Figure III.3.a on the right side show the position of the components in three directions, which are almost the same as the desired values, and on the left side show its error, which is defined as $e_1 = r - r_d$. The components in the three directions converge to the desired states or stable states. We observed the position of x, where it converges to the desired trajectory at t=1.5s, y converges at t=1s, while z converges in almost 2.5s. Figure III.3.b clearly shows the input control *T* and *M*, where the thrust was between$\mp 200\ N$. After 12s, *M2* and *M3* become zero. Figure III.3.c shows the third body-fixed frame $\vec{b}_3$ without changing the parameters. Figure III.4 shows the phase plan of Euler angles, which presents the clear stability of the system. Figure III.5 shows the trajectory in three dimensions.

Figure III.6 shows the results of the position of the third body-fixed frame $\vec{b}_3$ after changing the mass and the inertia, and then an external force F_{ex} was applied. We observed the suggested control robustness. Figure III.7 shows the trajectory of the quadrotor in 3D for two cases before and after changing the mass and the inertia by applying an external force F_{ex}.

In order to achieve the results obtained, we wanted to get the AR 2.0 quadrotor, but because of difficulty of getting we have chosen the Parrot Mambo mini-drone.

III .5 Hardware and Software Components

The experimental environment of the Parrot Mambo mini-drone is an indoor environment that has a size of 4x4 in meters. The Parrot Mambo mini-drone, which is shown in Fig IV.1, is equipped with 6-DOF (3-axis accelerometer and 3-axis gyroscope) Inertial Measurement Unit (IMU), an Ultrasound sensor, a Pressure sensor and a downward facing camera that has a 120x160 pixel resolution with a 60FPS measurement speed. The handled mini-drone has a Matlab/Simulink support package that gives the opportunity to access the internal sensor data and deploy control algorithms and sensor fusion methods in real-time. The generated MATLAB/Simulink codes are deployed wirelessly to Parrot Mambo mini-drone over Bluetooth. The sampling time of flight algorithms is fixed as $T = 0.005s$. The angular velocity w, linear velocity v, the position r and the attitude η of the mini-drone are estimated via a Kalman filter based sensor fusion algorithm. In order to overcome possible drifts in ζ, the optical flow algorithm has been employed, that uses the downward facing camera, to correct

the linear velocity vectors of the mini-drone. Therefore, in order to end up with an efficient optical flow detection performance, non-parallel lines called markers are randomly placed on the ground of the environment. The mini-drone has built-in impedance controller structure (abbreviated as Parrot PD but in fact, is impedance controller) for both the altitude and position control loops. The Parrot PD structures suffer from steady-state error, as expected, and result with poor control performances[75].

Figure III.8 The means used to achieve the experiment

1- Parrot mini drone (Mambo)
2- Safety goggles
3- Micro-USB cable
4- Bluetooth low energy dongle (CSR V4.0)
5- Additional drone battries
6- Drone battries charger

Any system has a set of sensors, which play the main role in acquiring the feedback signals for the control loops, which makes the set of inertial sensors that make up the IMU one of the

most important units in the Quadcopter system, which contributes to the stability and balance of the system.

The IMU sensor used is a 6-axis sensor, which combines a 3-axis Accelerometer with 3-Axis Gyroscope illustrated in figure IV.2, as well as a Digital Motion Processer, with an I2C pass, it accepts input values coming directly from the 3-axis compass sensor to achieve a full 9 Motion Fusion output Axes

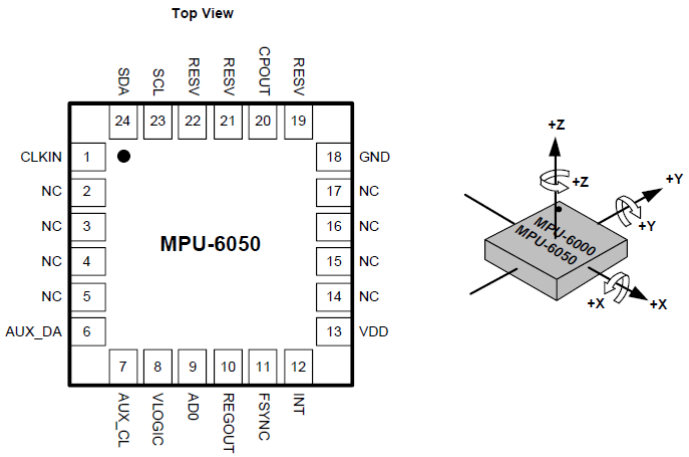


Figure III.9 The IMU sensor

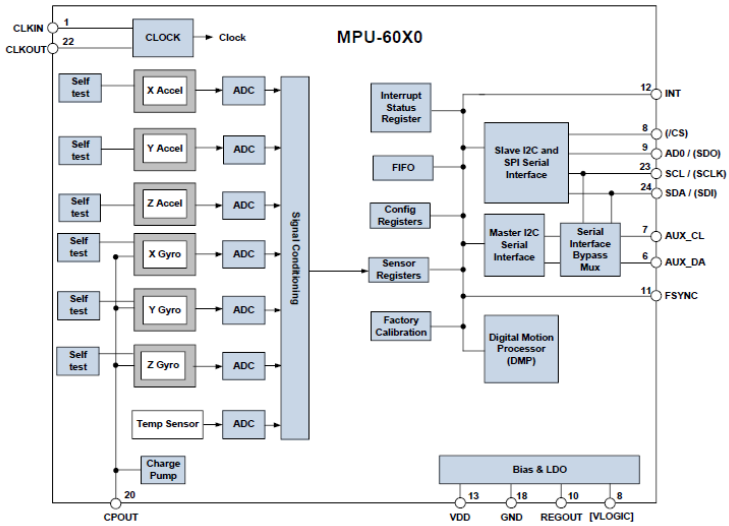


Figure III.10 MPU6050 . Sensor Block Diagram

III.5.1 Gyroscope

The sensor has three MEMS units whose function is to sense rotational movements on the x, y and z axes, where the maximum field of rotation around one of the axes (Scale Range Full) can be set according to the following areas $\pm 250, \pm 500, \pm 1000$ or ± 2000 [degree per sec(dps)] . The sensor also contains a LPF low-pass filter

III.5.2 working principle

When the sensor is rotated, as a result of the reaction, the MEMS unit will vibrate, and as a result of this vibration, a voltage signal is generated, which is processed to match the angle of rotation of the sensor, and then the resulting signal is enter into the ADC converter with an accuracy of 16bit. Figure IV.4 shows the microstructure of the gyroscope sensor built with MEMS technology.

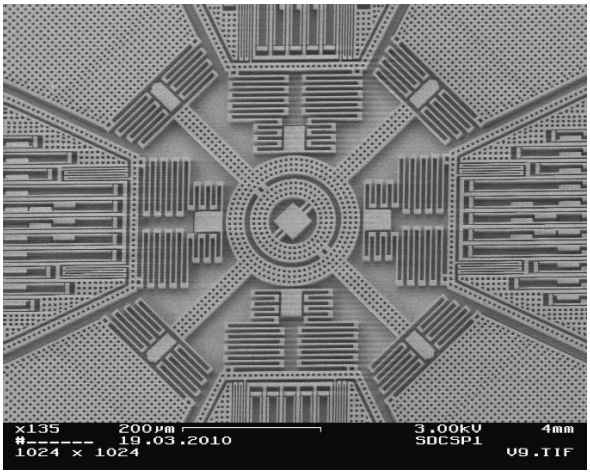


Figure III.11 The internal structure of the gyroscope

III.5.3 Accelerometer

This sensor senses acceleration through three MEMS units. Any acceleration of the sensor on either the x,y or z axes will result in a displacement of the Proof Mass (the mechanical part of the acceleration sensor MEMS), whereby the resulting displacement is measured using a capacitive sensor.

When the sensor is placed on a polished surface, the sensor output will be "0" on the x, y axes, while its value on the z-axis is +1g (where g is the acceleration due to gravity, with a weight factor of one). The accelerometer's weight factor is completely independent of the power supply, and the sensor allows the ability to adjust the full scale range according to the following areas $\pm 2g, \pm 4g, \pm 8g$ *or* $\pm 16g$. Figure IV.5 shows an illustration of the accelerometer sensor on one of the axis.

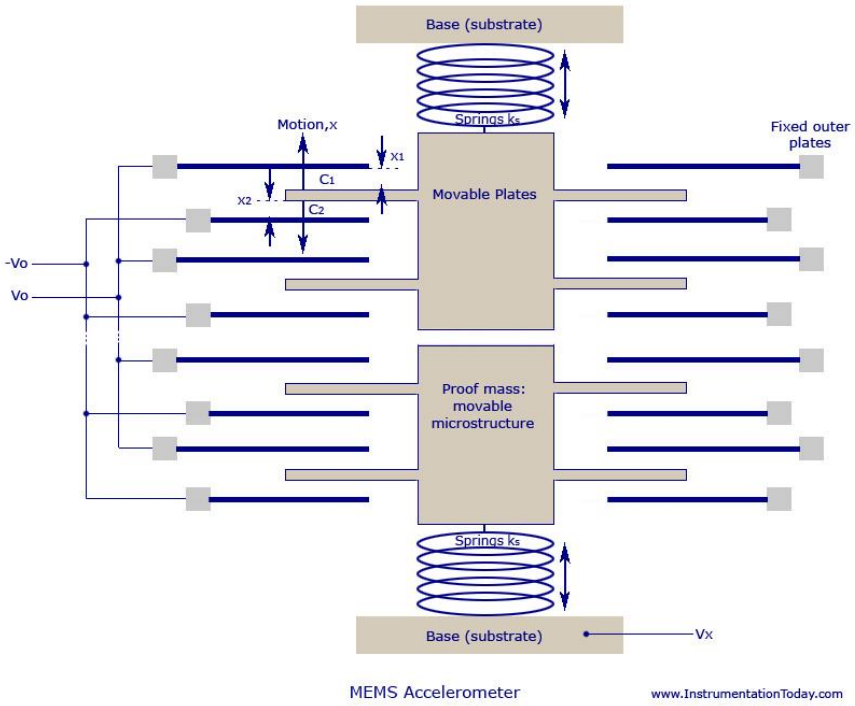


Figure III.12 the MEMS Accelerometer

In addition to information from an external unit (as in the Flight Controller where information is taken from a magnetometer), this information is necessary because the accelerometer is sensitive to vibrations, mechanical noise and linear acceleration.

The gyroscope is not sensitive to linear mechanical motion, but it has a zero-drift problem, which means that the sensor does not return to zero at the output after the motion is gone, and the information provided by the magnetometer is unreliable because its values are affected by the surrounding magnetic materials.

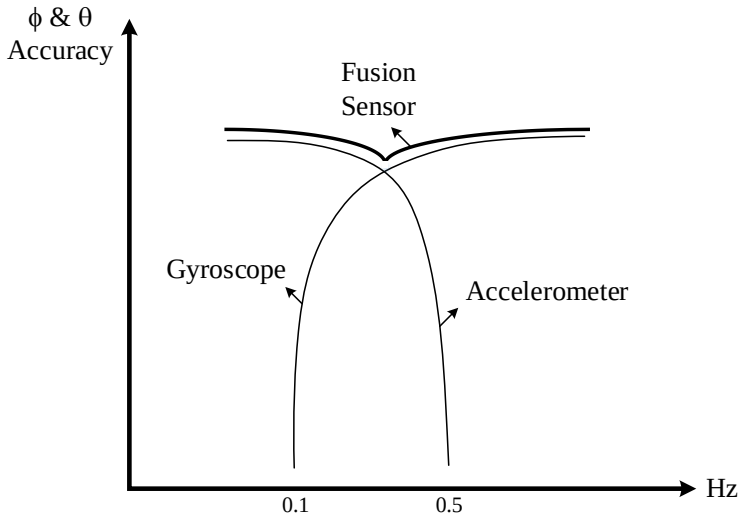


Figure III.13 the motion processing algorithm function

III.5.4 Calculating the two angles θ, ϕ using the accelerometer

$$G_p = \begin{bmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{bmatrix} = R(g - a_r) \tag{III.32}$$

G_p : The vector representing the acceleration sensor's output.

R : Orientation matrices on the axes that describe the orientation of the Quadcopter with respect to the fixed reference system.

G : The vector of acceleration due to gravity.

a_r : Linear acceleration of the sensor.

To solve equation (IV.32) we consider that the linear acceleration of the sensor is equal to zero (0), the appearance of any linear acceleration will lead to the generation of errors in the measured value of the steering angle (one of the reasons for using the gyroscope with the Accelerometer).

The z-axis mounted on the fuselage is along the z-axis of the reference coordinates, and as mentioned earlier, the unit of measure for the sensor's output values is (g), the vector g become $g = [0 \quad 0 \quad 1]^T$. Equation (IV.1) can be written as:

$$G_p = \begin{bmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{bmatrix} = R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{III.33}$$

The multiplication operation can be performed to obtain the matrix R in six different arrangements, each of which indicates a different orientation of the sensor, according to the order of the rotations around the axes.

$$R_{xyz} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -S_\theta \\ C_\theta S_\phi \\ C_\theta C_\phi \end{bmatrix} \tag{III.34}$$

$$R_{yxz} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -S_\theta C_\phi \\ S_\phi \\ C_\theta C_\phi \end{bmatrix} \tag{III.35}$$

$$R_{xzy}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}-C_{\psi}S_{\theta}\\C_{\theta}S_{\phi}+C_{\phi}S_{\psi}S_{\theta}\\C_{\phi}C_{\theta}-S_{\theta}S_{\phi}S_{\psi}\end{bmatrix} \quad \text{(III.36)}$$

$$R_{yzx}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}C_{\theta}S_{\phi}S_{\psi}-S_{\theta}C_{\phi}\\C_{\psi}S_{\phi}\\C_{\theta}C_{\phi}+S_{\theta}S_{\phi}S_{\psi}\end{bmatrix} \quad \text{(III.37)}$$

$$R_{zxy}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}C_{\theta}S_{\phi}S_{\psi}-S_{\theta}C_{\psi}\\C_{\psi}C_{\theta}S_{\phi}+S_{\theta}S_{\psi}\\C_{\theta}C_{\phi}\end{bmatrix} \quad \text{(III.38)}$$

$$R_{zyx}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}S_{\phi}S_{\psi}-C_{\phi}C_{\psi}S_{\theta}\\C_{\psi}S_{\phi}-C_{\phi}S_{\psi}S_{\theta}\\C_{\theta}C_{\phi}\end{bmatrix} \quad \text{(III.39)}$$

Equation (III.34) and (III.35), the two angles can be estimated ϕ,θ. From equation (III.36) to equation (III.39) it will result in equations with a number of unknowns greater than the number of equations, and this will lead to the inability to measure the angle ψ ، which expresses the rotation around the z-axis using the accelerometer readings. From equation (III.34) and since the measurements (sensor output measurements) are subject to the process of normalization in order to become g = 1, the equation will be written as follows:

$$\frac{G_p}{|G_p|}=\frac{1}{\sqrt{G_{px}^2+G_{py}^2+G_{pz}^2}}\begin{bmatrix}G_{px}\\G_{py}\\G_{pz}\end{bmatrix}=\begin{bmatrix}-S_{\theta}\\C_{\theta}S_{\phi}\\C_{\theta}C_{\phi}\end{bmatrix} \quad \text{(III.40)}$$

From equation (III.40) it can be found:

$$\tan\phi_{xyz}=\left(\frac{G_{py}}{G_{pz}}\right) \quad \text{(III.41)}$$

$$\tan\theta_{xyz}=\frac{-G_{px}}{G_{px}\sin\phi+G_{py}\cos\phi}=\frac{-G_{px}}{\sqrt{G_{pz}^2+G_{py}^2}} \quad \text{(III.42)}$$

Calculate the values of angles based on equation (III.35):

$$\frac{G_p}{|G_p|} = \frac{1}{\sqrt{G_{px}^2 + G_{py}^2 + G_{pz}^2}} \begin{bmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{bmatrix} = \begin{bmatrix} -S_\theta C_\phi \\ S_\phi \\ C_\theta C_\phi \end{bmatrix} \tag{III.43}$$

$$\tan \phi_{yxz} = \frac{G_{py}}{\sqrt{G_{px}^2 + G_{pz}^2}} \tag{III.44}$$

$$\tan \theta_{yxz} = \left(\frac{-G_{px}}{G_{pz}} \right) \tag{III.45}$$

III.5.5 Getting rid of the Duplicate Solutions problem in trigonometric functions

The equations (III.33) and (III.34) can be written as follows:

$$\begin{bmatrix} -\sin\theta \\ \cos\theta \sin\phi \\ \cos\theta \cos\phi \end{bmatrix} = \begin{bmatrix} -\sin(\pi - \theta) \\ \cos(\pi - \theta) \sin(\pi + \phi) \\ \cos(\pi - \theta) \cos(\pi - \phi) \end{bmatrix} \tag{III.46}$$

$$\begin{bmatrix} -\sin\theta \cos\phi \\ \sin\phi \\ \cos\theta \cos\phi \end{bmatrix} = \begin{bmatrix} -\sin(\pi + \theta) \cos(\pi - \phi) \\ \sin(\pi - \phi) \\ \cos(\pi + \theta) \cos(\pi - \phi) \end{bmatrix} \tag{III.47}$$

This problem can be solved by specifying the range of change of one of the two angles θ, ϕ within the domain $[-90^\circ, +90^\circ]$ and leaving a range in front of the other angle open $[-180^\circ, +180^\circ]$، However, the problem of the emergence of solutions belonging to different quarters will arise, The correct angle can be determined by studying the signs G_{px}, G_{py}, G_{pz} .In order to obtain the values of angles using the gyroscope, a single integration operation must be performed on the reading at each axis to reach the required angle value.

III.5.6 Deducing Euler angles in a magnetometer

As in the Accelerometer, the magnetometer gives a measurement related to the radius of the Earth's magnetic field, so we need transmission matrices, to deduce the value of the angle ψ based on the sensor reading values.

$$B_r = B \begin{bmatrix} C_\delta \\ 0 \\ S_\delta \end{bmatrix} \tag{III.48}$$

Where:

B_r : The measured value

B : Earth's magnetic field strength

δ : The angle of inclination of the Earth's magnetic field beam considering the south pole represented by -90° and the north pole by $+90^{\circ}$.

$$B_p = R_x(\phi)R_y(\theta)R_z(\psi)B_r + V \tag{III.49}$$

V : The ray expressing the effect of magnetic objects (Hard Iron)

$$\begin{bmatrix} C_\psi BC_\delta \\ -S_\psi BC_\delta \\ BS_\delta \end{bmatrix} = \begin{bmatrix} C_\theta & 0 & -S_\theta \\ 0 & 1 & 0 \\ S_\theta & 0 & C_\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_\phi & S_\phi \\ 0 & -S_\phi & C_\phi \end{bmatrix} \begin{bmatrix} B_{px} - V_x \\ B_{py} - V_y \\ B_{pz} - V_z \end{bmatrix} \tag{III.50}$$

$$\begin{bmatrix} B_{fx} \\ B_{fy} \\ B_{fz} \end{bmatrix} = \begin{bmatrix} (B_{px} - V_x)C_\theta + (B_{py} - V_y)S_\phi S_\theta - (B_{pz} - V_z)C_\phi S_\theta \\ (B_{py} - V_y)C_\phi - (B_{pz} - V_z)S_\phi \\ -(B_{px} - V_x)S_\theta + (B_{py} - V_y)S_\phi C_\theta + (B_{pz} - V_z)C_\phi C_\theta \end{bmatrix} \tag{III.51}$$

$$B_{fx} = C_\psi BC_\delta \tag{III.52}$$

$$B_{fy} = -S_\psi BC_\delta \tag{III.53}$$

$$\tan\psi = \frac{-B_{fy}}{B_{fx}} = \frac{(B_{py} - V_y)C_\phi - (B_{pz} - V_z)S_\phi}{(B_{px} - V_x)C_\theta + (B_{py} - V_y)S_\phi S_\theta - (B_{pz} - V_z)C_\phi S_\theta} \tag{III.54}$$

The method of conclusion of Euler angles and processes to be applied to sensors are reviewed to reach the values of angles, However, as it was pointed out, that each sensor has several problems that lead to an increase in the uncertainty of the measured value, Therefore, a motion-processing algorithm must be added to eventually lead to reliable information based on sensor readings. There are several algorithms followed.

III.5.7 Extended Kalman Filter

The EKF algorithm consists of two main stages, the first is the Prediction stage, and the second is stage is the Correction stage. Also, the EKF algorithm handles the singularity condition that can appear when one of the x or y axes moves to be aligned with the gravitational ray.

1- The Prediction stage

The vector of the state variables estimated by the algorithm is shown in equation (III.55).

$$\hat{x} = \begin{bmatrix} \hat{\theta} & \hat{\dot{\theta}}_b \end{bmatrix}^{\mathrm{T}} \tag{III.55}$$

Where:

$\hat{\theta}$: Estimated angle

$\hat{\dot{\theta}}_b$ **:** The displacement value generated during the measurement process

$$\hat{x}_{k+1|k} = F\hat{x}_{k|k} + Bu_k \tag{III.56}$$

$\hat{x}_{k+1|k}$: The radius of state variables estimated in the current discrete value

F : system matrix

$\hat{x}_{k|k}$: vector Corrected state variables in the previous discrete value

B : input matrix

u_k : input vector

$$F = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \tag{III.57}$$

It is noticeable that the element $F_{1\times 2} = -\Delta t$ This is in order to eliminate the displacement caused by the gyroscope sensor.

$$u_k = \dot{\theta}_k \tag{III.58}$$

The input vector is shown in the relation (III.59) which is the angular velocity measured using a gyroscope.

$$B = \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \tag{III.59}$$

The element represents the time value to perform the integration necessary to the conclusion of the value of the angle.

$$Z_k = Hx_k + R \tag{III.60}$$

Where $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$ and $x_k = \begin{bmatrix} \theta & \dot{\theta}_b \end{bmatrix}^{\mathrm{T}}$

Z_k : Output vector measured based on the accelerometer

x_k : Real state variables vector

H : Output vector

R : The noise accompanying the measurement

$$P_{k+1|k} = FP_{k|k}F^{\mathrm{T}} + Q \tag{III.61}$$

$P_{k|k}$: The divergence matrix between the estimated value and the corrected measured value in the previous discrete value

$P_{k+1|k}$: The divergence matrix between the estimated value and the measured value that is computed in the current discrete value

Q : The divergence matrix

$$Q = \begin{bmatrix} Q_{\theta} & 0 \\ 0 & Q_{\dot{\theta}_b} \end{bmatrix} \Delta t \tag{III.62}$$

Q_{θ} : The divergence between the real value and the measured value

$Q_{\dot{\theta}_b}$: The divergence in the displacement value of the gyroscope

It is noticeable that the gain values are multiplied by the time value, because the noise value increases with the increase in the time interval from the last update of the values.

The angle value based on the acceleration sensor reading is given by (IV.62).

The big advance for these common sensors is that they are being combined on a single chip which makes them extremely easy to use. You can now get one chip that has three accelerometers, three gyroscopes, and a magnetometer [76, 77] for about $5. This multi-sensor chip trend will continue and dramatically lower the cost of each individual sensor.

Software Project

1- At least Matlab R2017 a

2- Simulink support package for parrot Minidrone (v.17.2.1 or hight)

3- Aerospace Blockset

4- Simulink 3D animation

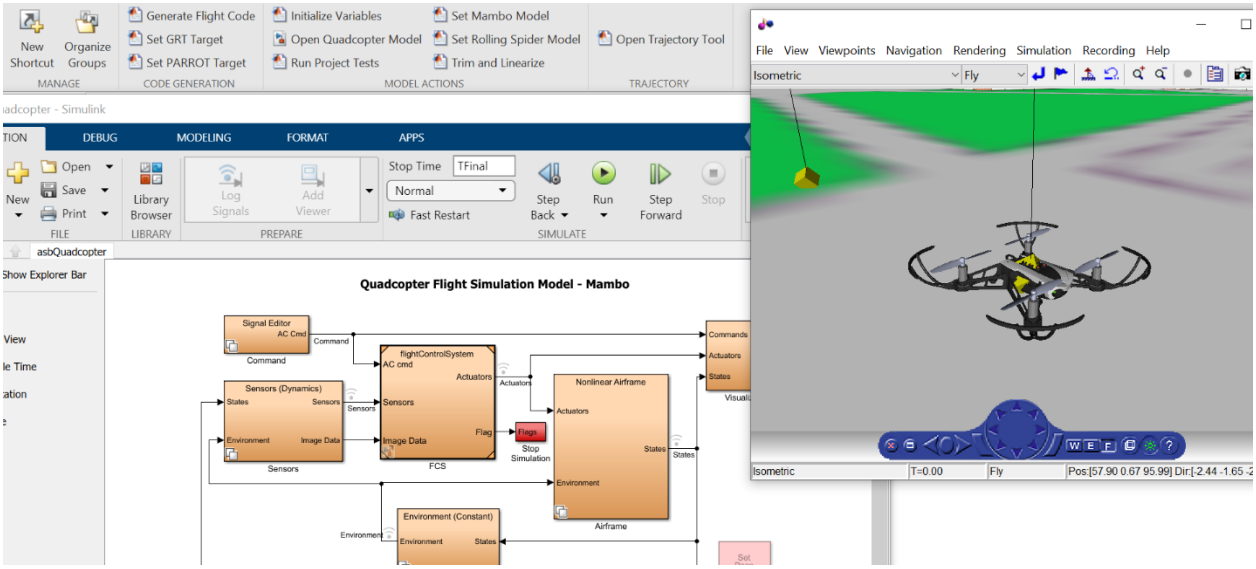


Figure III.14 asbQuadcopterStart project used to achieve the experiment

III.5 Conclusion

In this work, a dynamic global model for a quadrotor UAV was presented. Furthermore, a new tracking controller's development was based on a backstepping geometric method developed directly on the SE (3) to avoid singularities and ambiguities inherent to other attitude representations. The main conclusions are summarized as follows, (a) The state variables can converge fast to the desired value. (b) The errors of position and attitude angles can quickly converge to 0 or stay in a small percentage. (c) The new control strategy which suits complex flight missions can make the quadrotor track different trajectories. Thus, simulations have validated the different prescribed trajectories tracked by actual trajectories very well with this work proposed method.

Chapter IV:
Navigation techniques for control attitude of quadrotor

IV.1 Introduction

Most practical application for mini quadcopter requires robust and accurate control system. In mambo mini quadcopter the impedance control system is used to reduce the risk while a human interaction is essential. However, the impedance control system has some limitations (i.e., tracking high frequency command and load distance). In this chapter, High Dynamic Performance control system was designed and implemented to show the limitation of the impedance control.

IV.2 Experiments Design:

The control systems were tested for two degrees of freedom only (roll and pitch) the bench test experiment figure VI.1 is designed to restrict the system transitional movements and allow the quadcopter to rotate around on axis (x or y) at a time. During the experiments, the thrust was selected to be minimum for take of and the rotation around other axes were set to zero. All tracking experiments were executed for 50 second and done for two rotations (roll and pitch). The control systems ability to track high frequencies commands a sinusoidal reference was selected with 0.4 rad of amplitude and increasing frequency starting with 0.1 rad/s to reach the highest frequency (3 rad/s) possible to achieve by quadcopter's motors.

The accurate tracking reference were set to test the control systems with large angles to establish system ability under nonlinear conditions. Three mechanical loads (60, 75 and 95 g) were used to assess the control system performance when the load is suddenly released. During unloading the reference was set to zero.

The wind disturbance was imitated by using an industrial fan with three speed effecting the body of the drone at three different wind speeds (2.2, 3.2 and 4.2 m/s). In Contrary to the impulse unloading experiments the reference were additionally set to be sinusoidal.

Figure IV.1 Environmental test bench

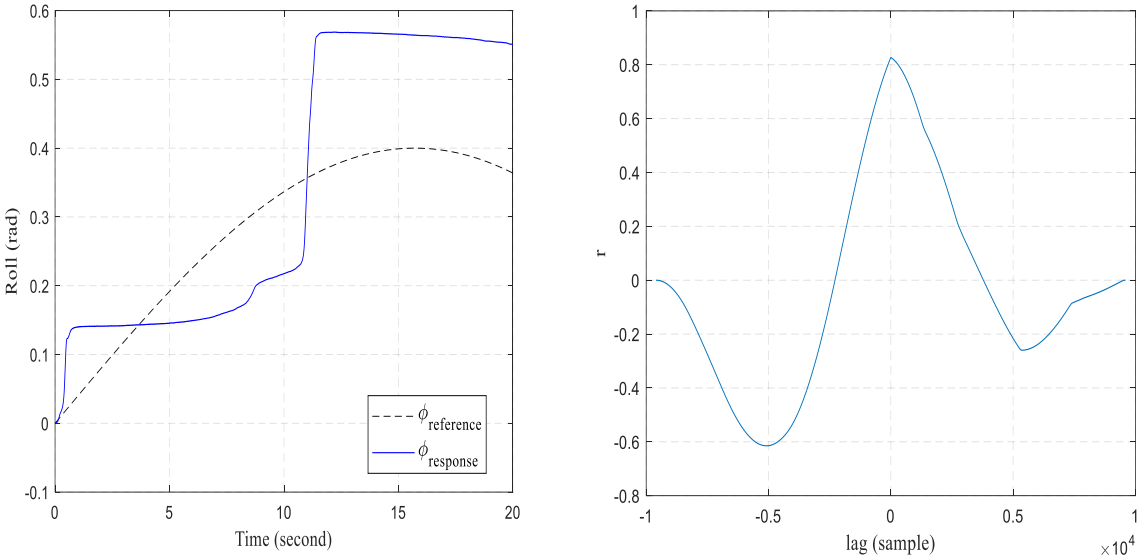
IV.3 Impedance control

One primitive approach to implementing impedance control[78-80], proposed by Hogan (1985), has been used with considerable success. Termed “simple” impedance control, it consists of driving an intrinsically low friction mechanism with force- or torque-controlled actuators, and using motion feedback to increase output impedance[81, 82]. This approach makes no attempt to compensate for any physical impedance (mass, friction) in the mechanism, so the actual output impedance consists of that due to the controller plus that due to the mechanism.

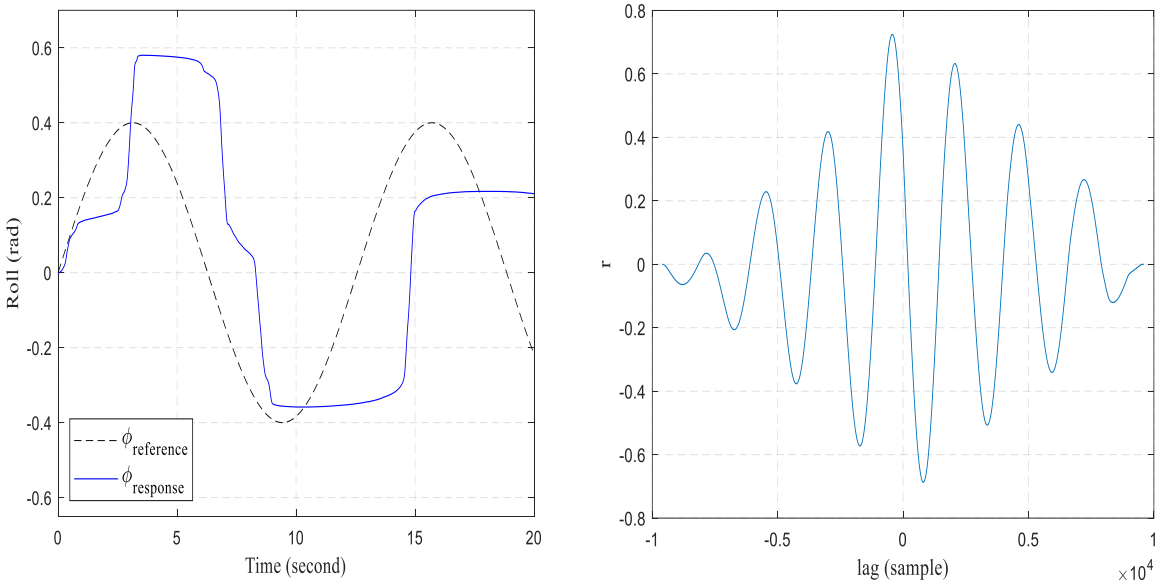
A distinction between impedance control and the more conventional approaches to manipulator control is that the controller attempts to implement a dynamic relation between manipulator variables[83-86], such as end· point position and force rather than just control these variables alone. This change in perspective results in a simplification of several control problems.

IV.4 Result experiment for Roll and pitch

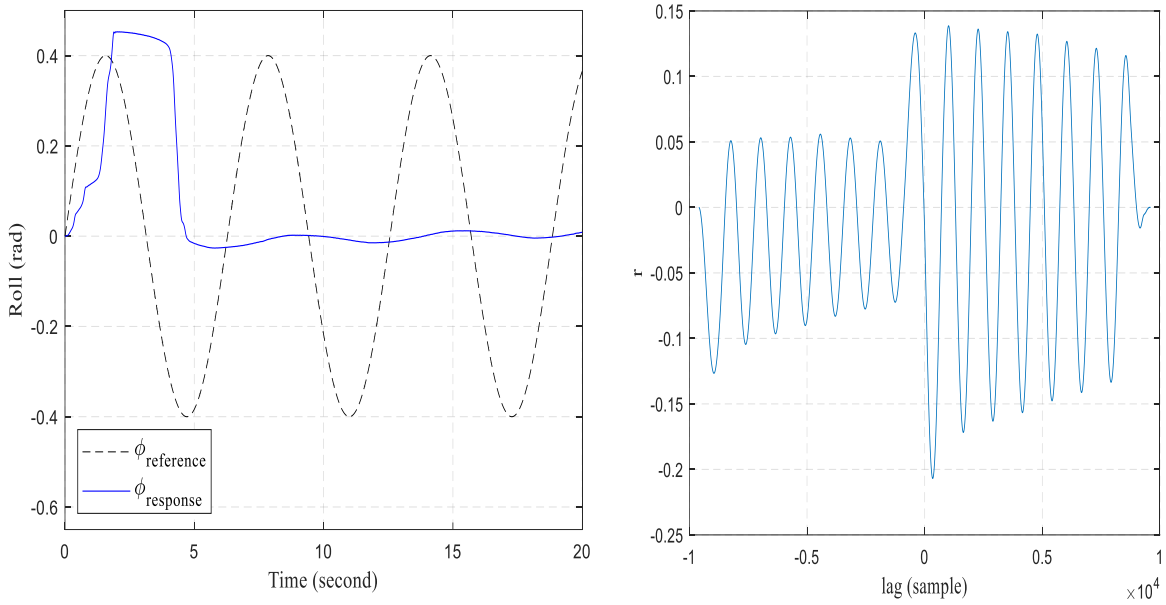
IV.4.1 Result experiment for Roll



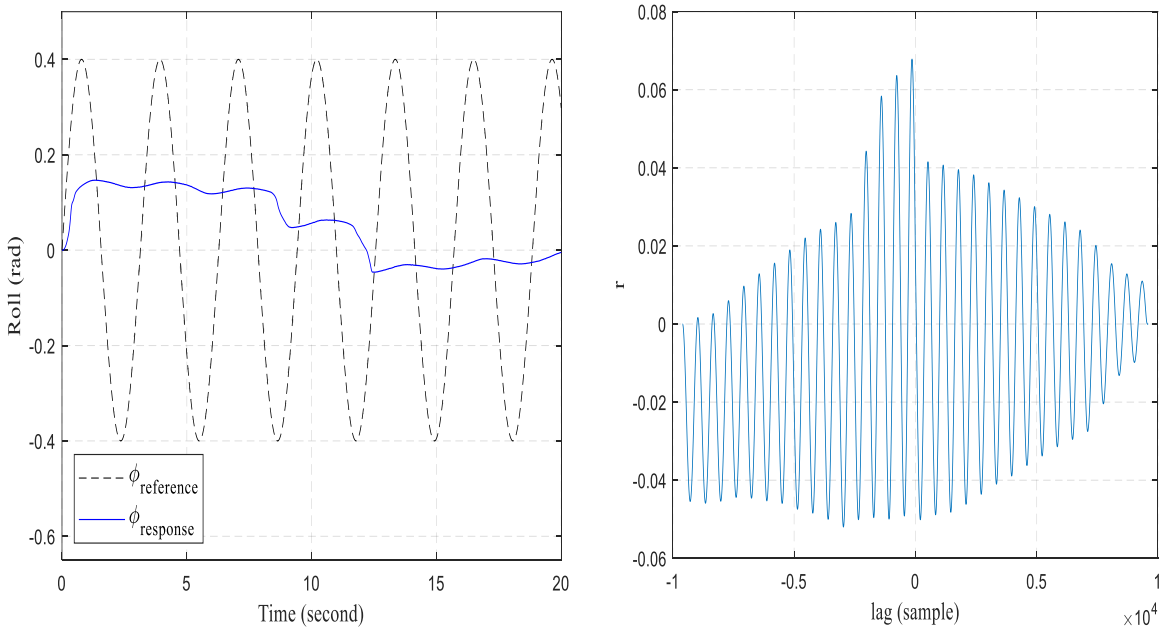
a sinusoidal reference 0.1 rad/s



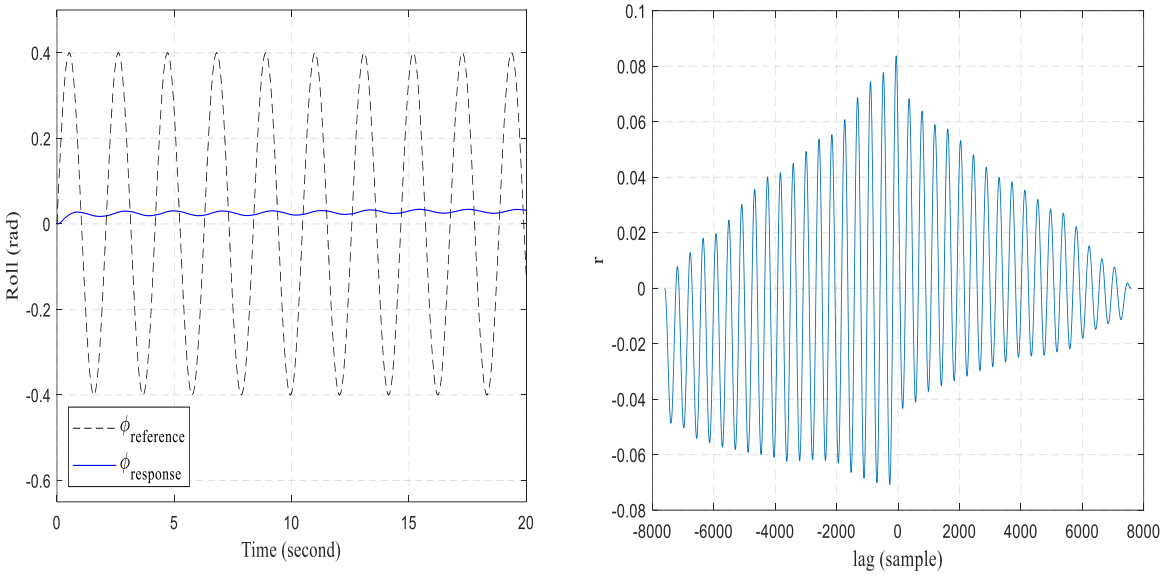
b sinusoidal reference 0.5 rad/s



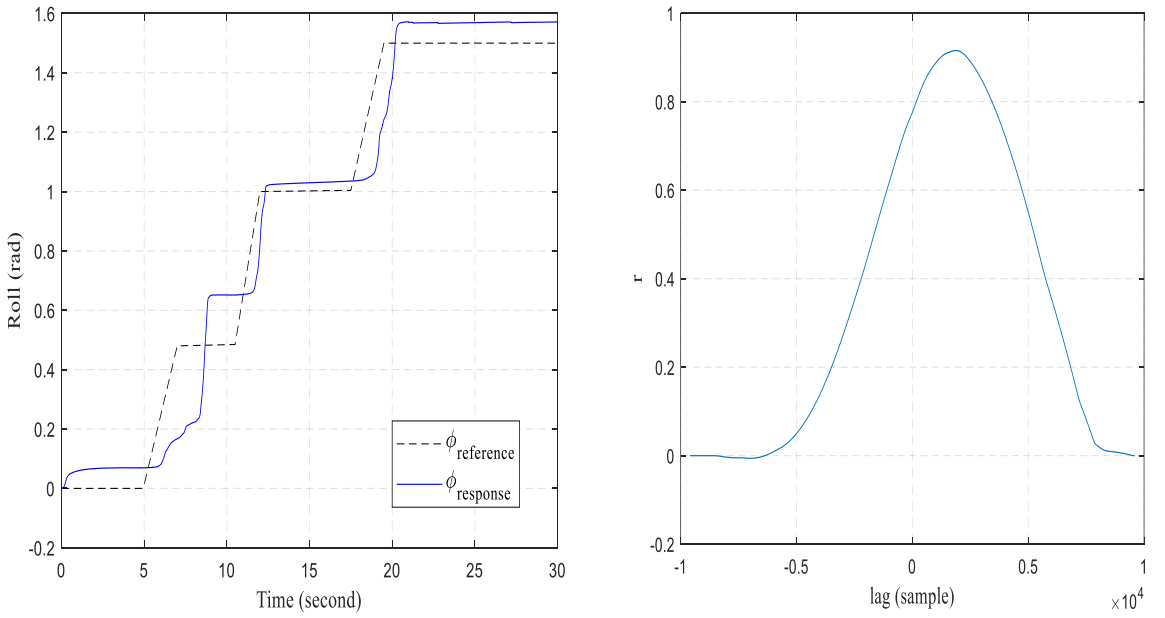
c sinusoidal reference 1 rad/s



d sinusoidal reference 2 rad/s



e sinusoidal reference 3 rad/s

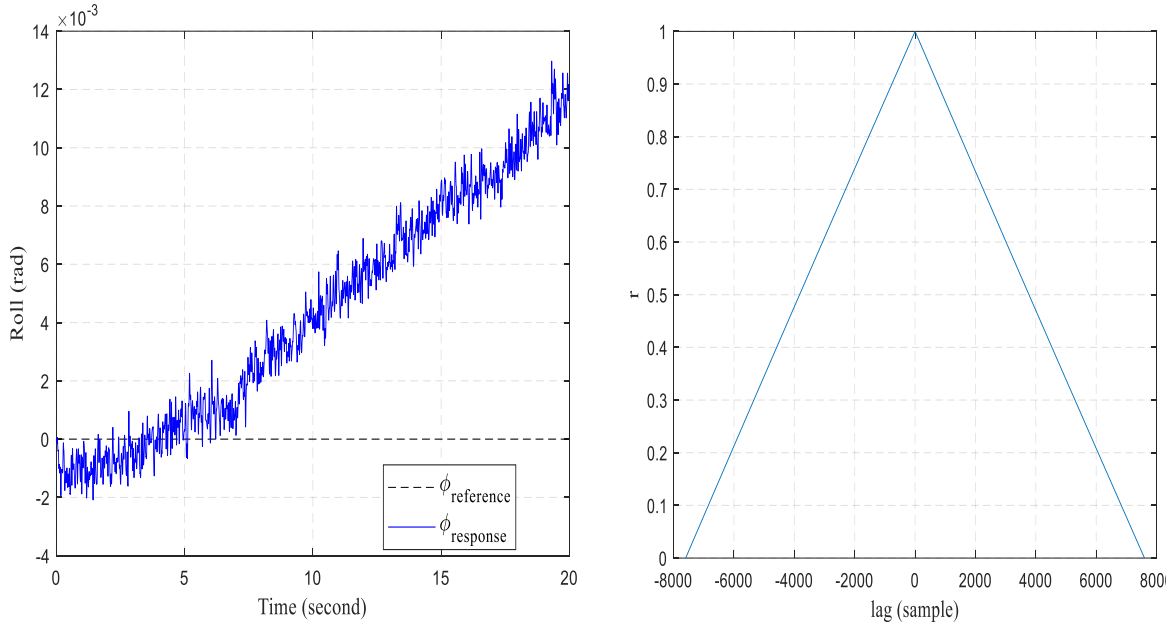


f ramp reference.

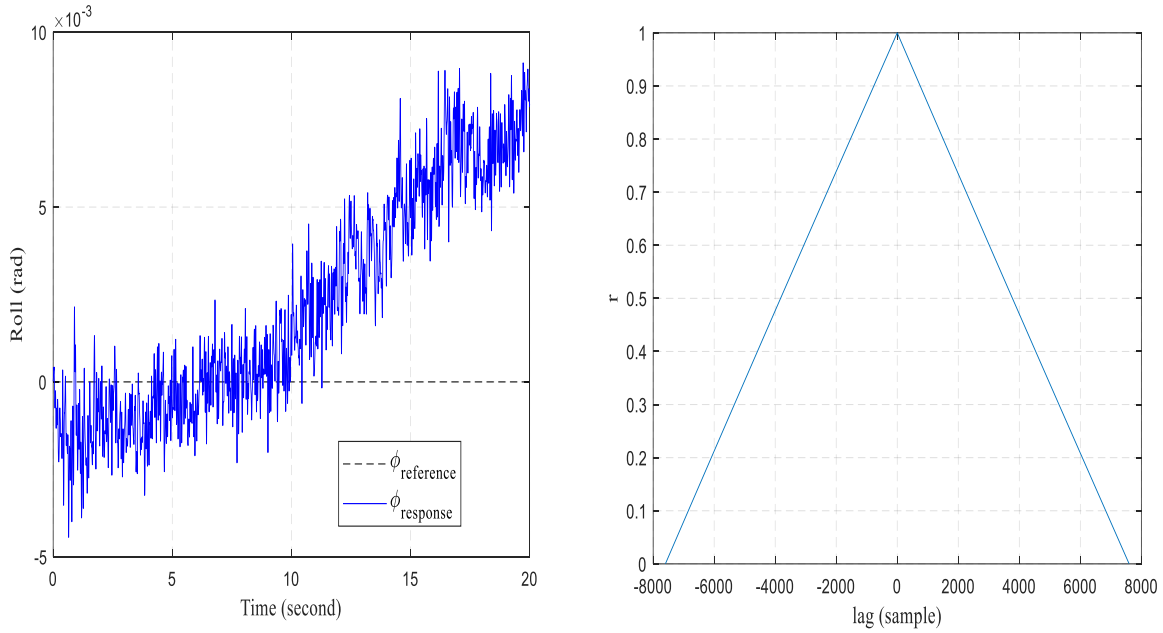
Figure IV.2 track different frequencies command a sinusoidal reference and ramp reference.

The experimental results of Impedance control of roll and pitch demonstrate the inefficiency of this control, in terms of robustness and tracking, where figure IV.2 shows tracking frequencies command a sinusoidal reference in frequency equal 1 rad/s and of roll and pitch and ramp reference. Besides, To show the control ability to track reference with big tilt

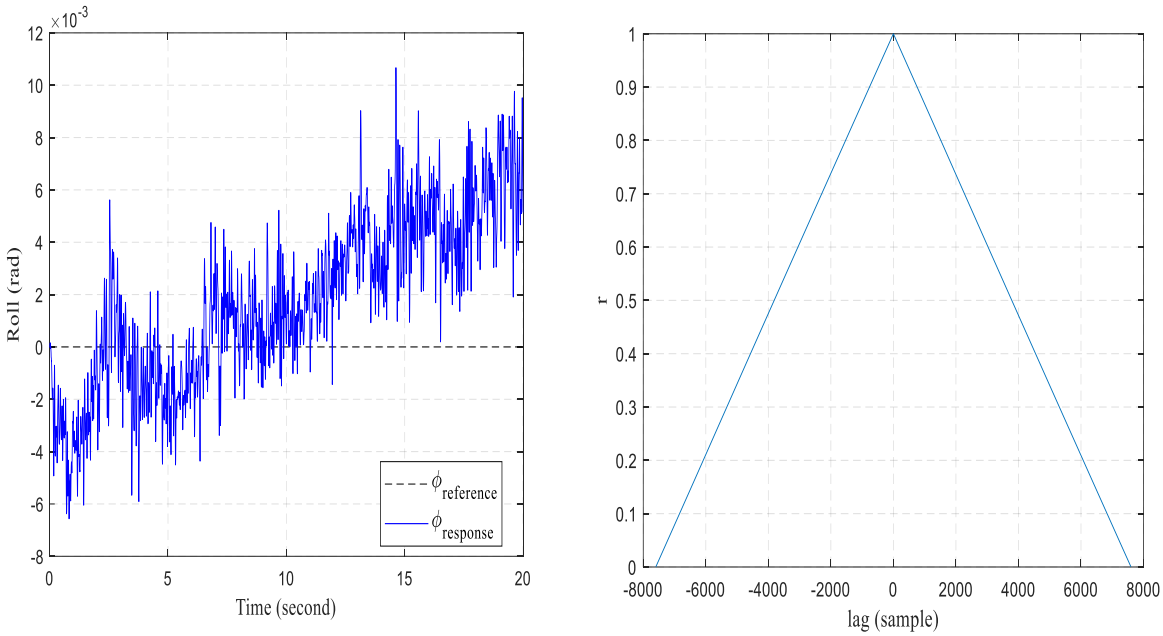
values (more than 10 degrees), a ramp command was generated to prove system stability around 90 degrees command. The cross-correlation shows the system is not able to track the command with lag equal 0.9155/1866. Similar, result is found when the control system was tested pitch (see figures IV.5)



a The wind disturbance 2.2m/s



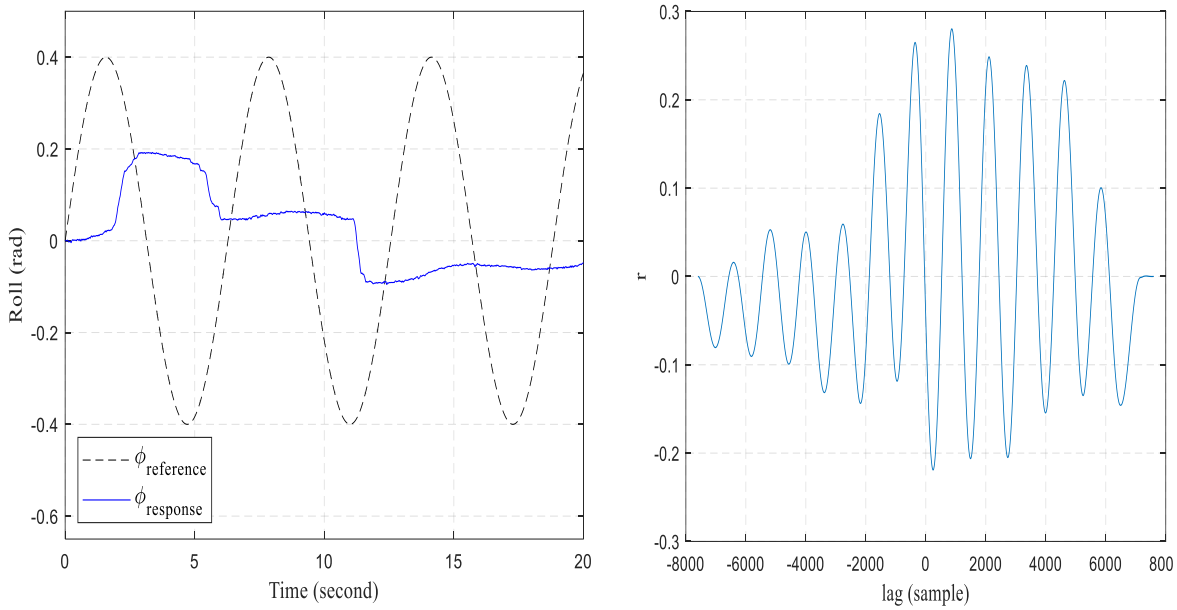
b The wind disturbance 3.2m/s



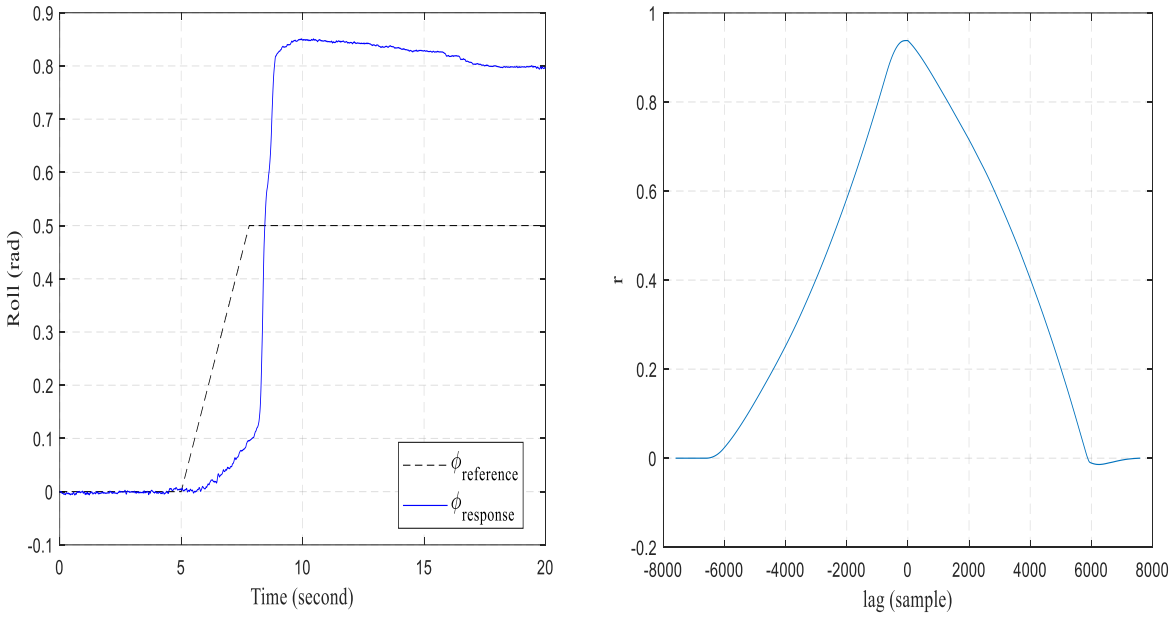
c The wind disturbance 4.2m/s

Figure IV.3 The wind disturbance.

Figure IV.3 and Figure IV.6 clarified the wind disturbance with tracking different reference commands is illustrated in figure VI.1. Also, the experimental results of Impedance control of roll and pitch demonstrate the inefficiency of this control with a maximum tracking error 2.06 degree at 4.2 m/s wind speed. Furthermore, the system shows bad tracking for sinusoidal and ramp command with maximum wind disturbance (4.2 m/s) as is shown in figure IV.4 and figure IV.7.



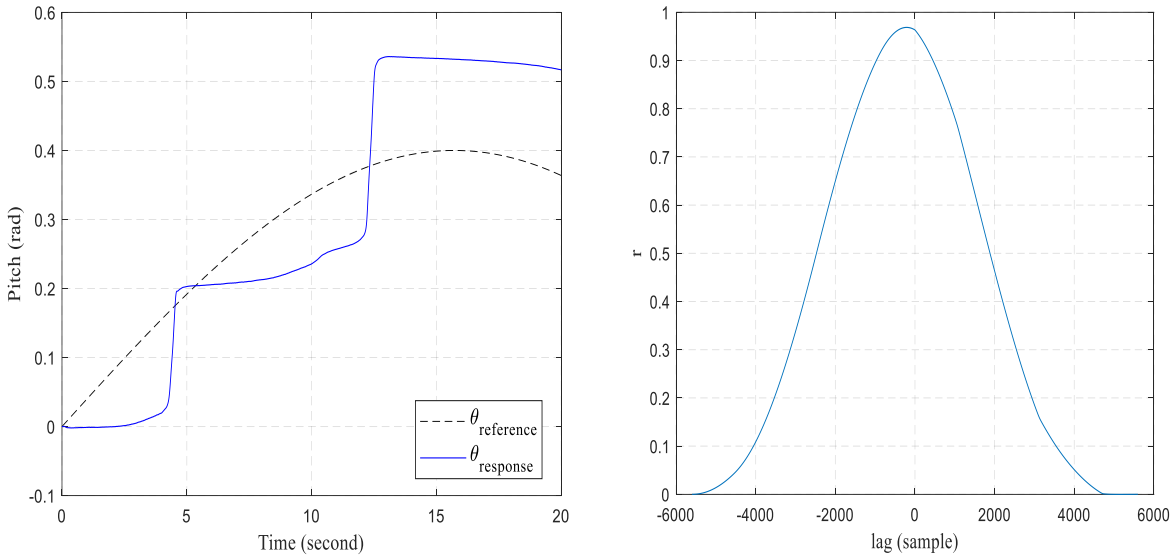
a sinusoidal reference 3 rad/s and wind disturbance 4.2m/s



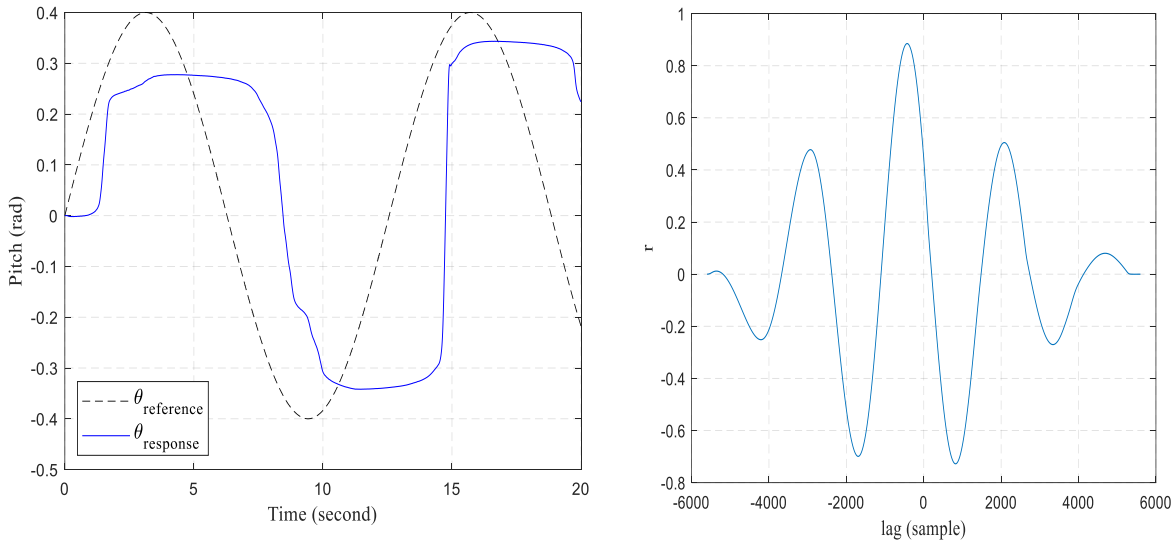
b ramp reference and wind disturbance 4.2m/s

Figure IV.4 The wind disturbance with tracking different reference command

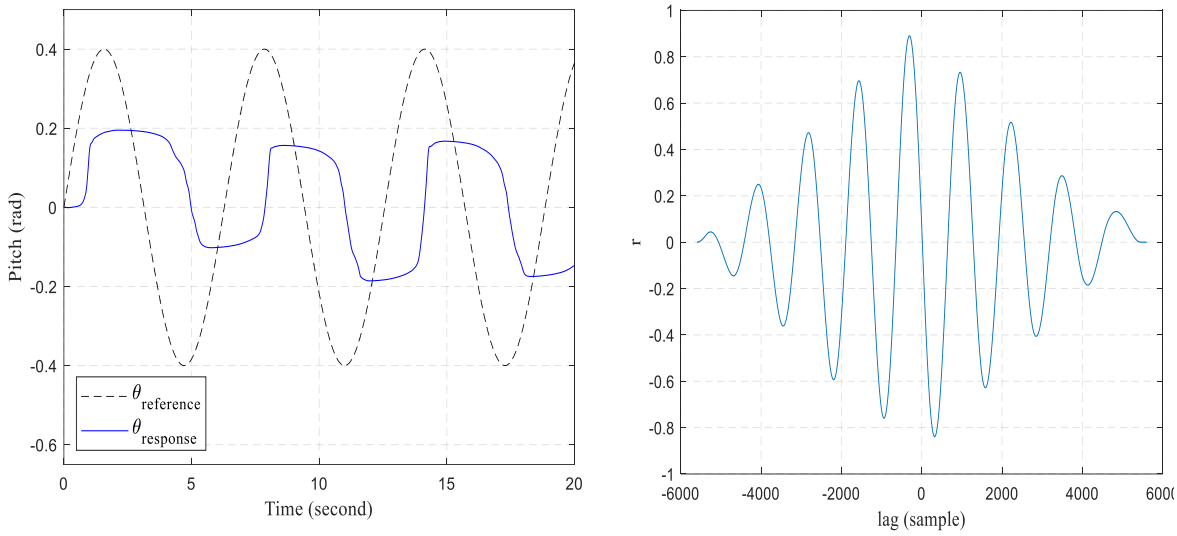
IV.4.2 Result experiment for Pitch



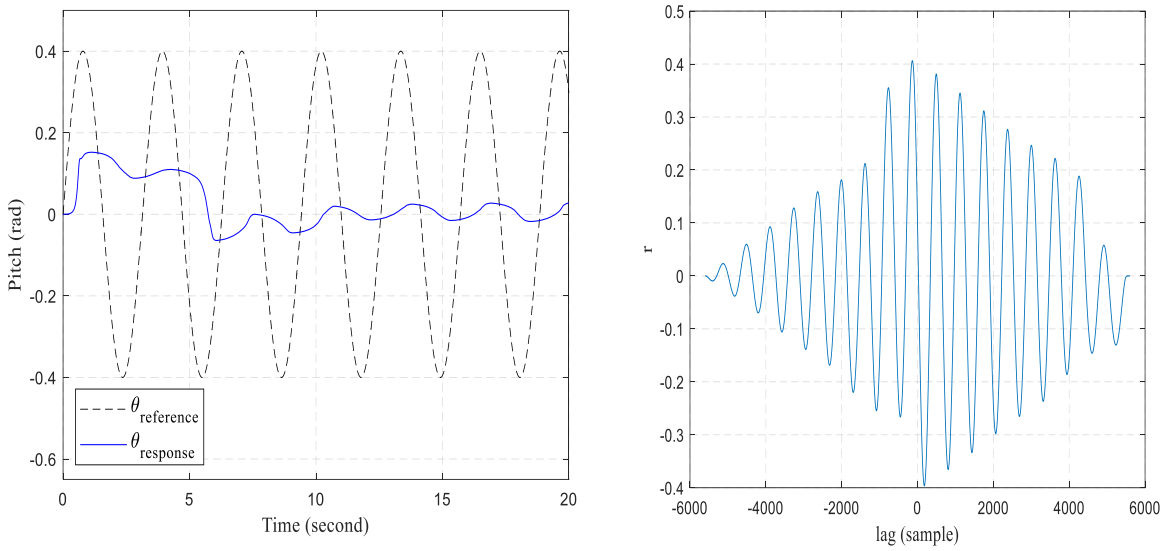
a sinusoidal reference 0.1 rad/s



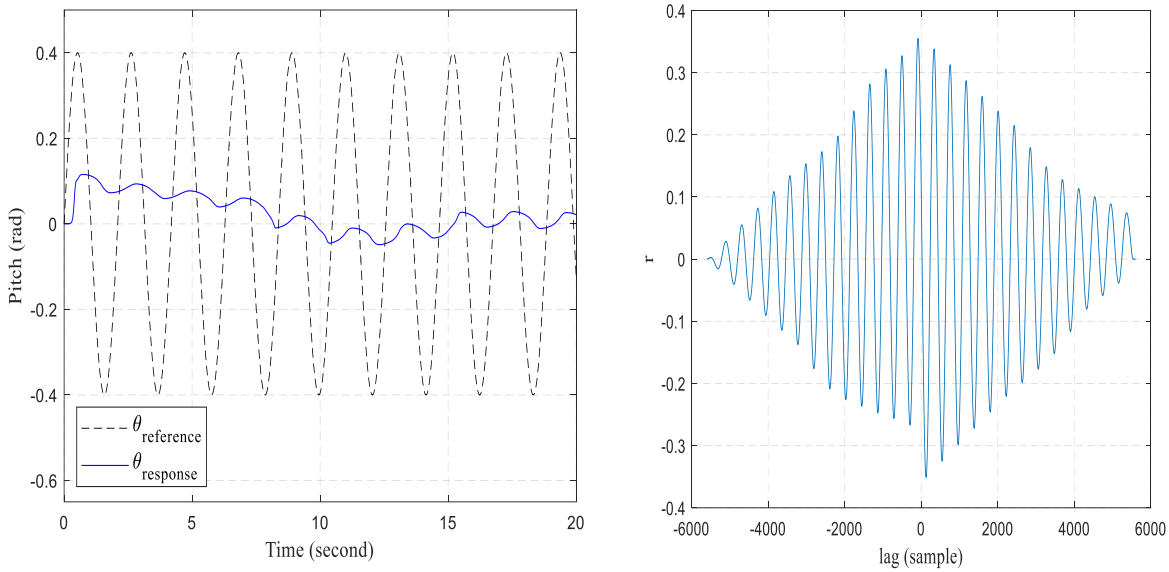
b sinusoidal reference 0.5 rad/s



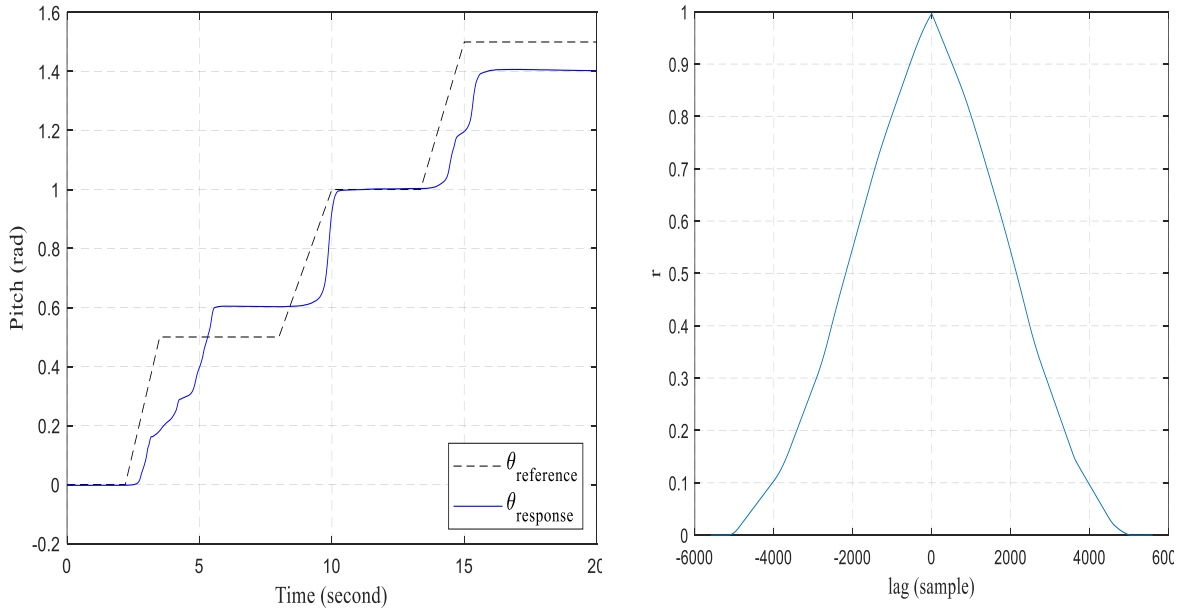
c sinusoidal reference 1 rad/s



d sinusoidal reference 2 rad/s

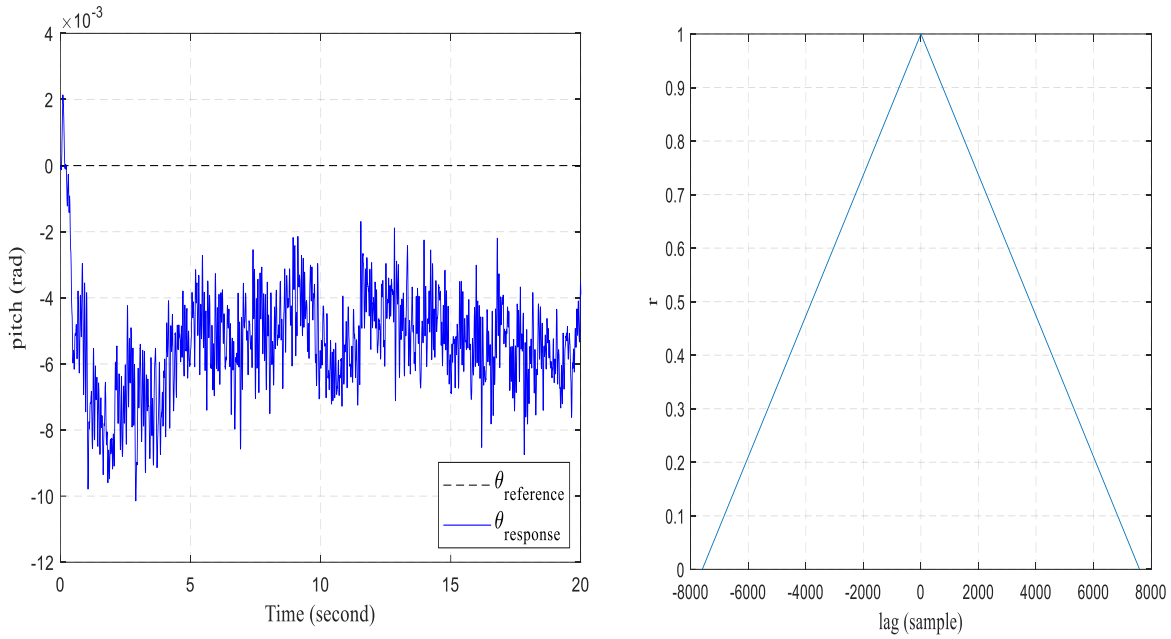


e sinusoidal reference 3 rad/s

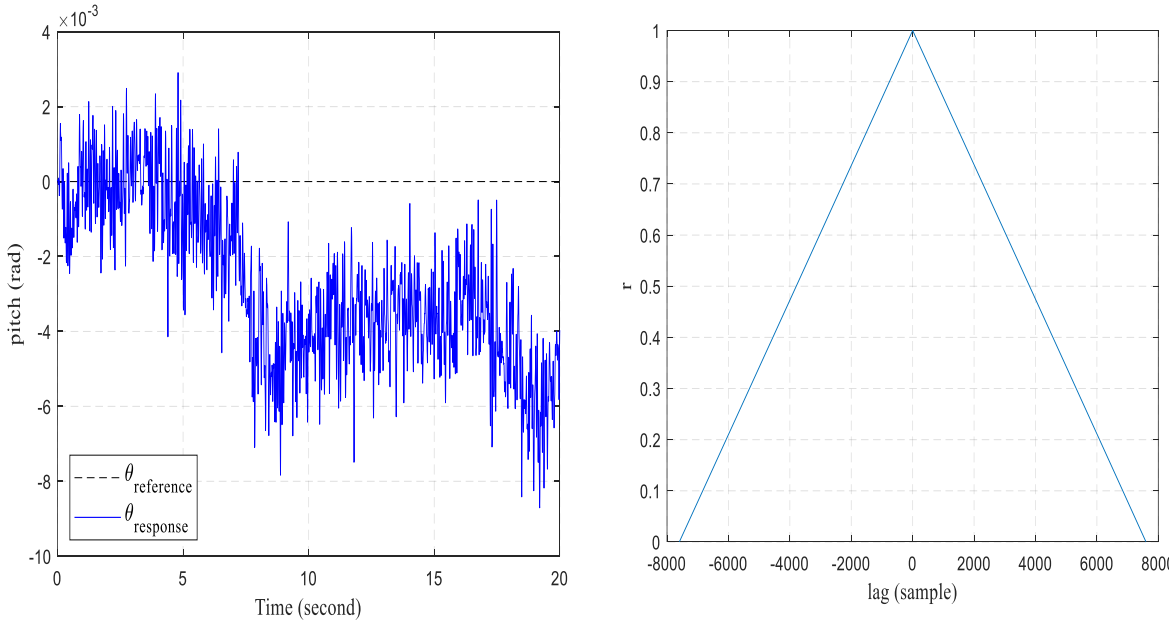


f ramp reference.

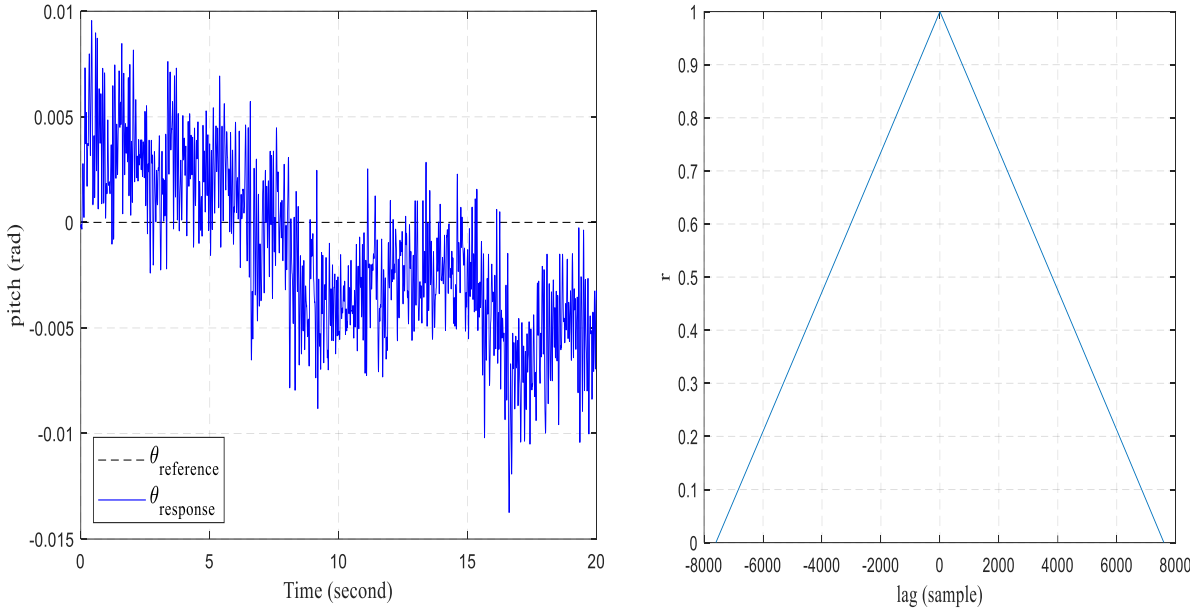
Figure IV.5 track different frequencies command a sinusoidal reference and ramp reference.



a The wind disturbance 2.2m/s

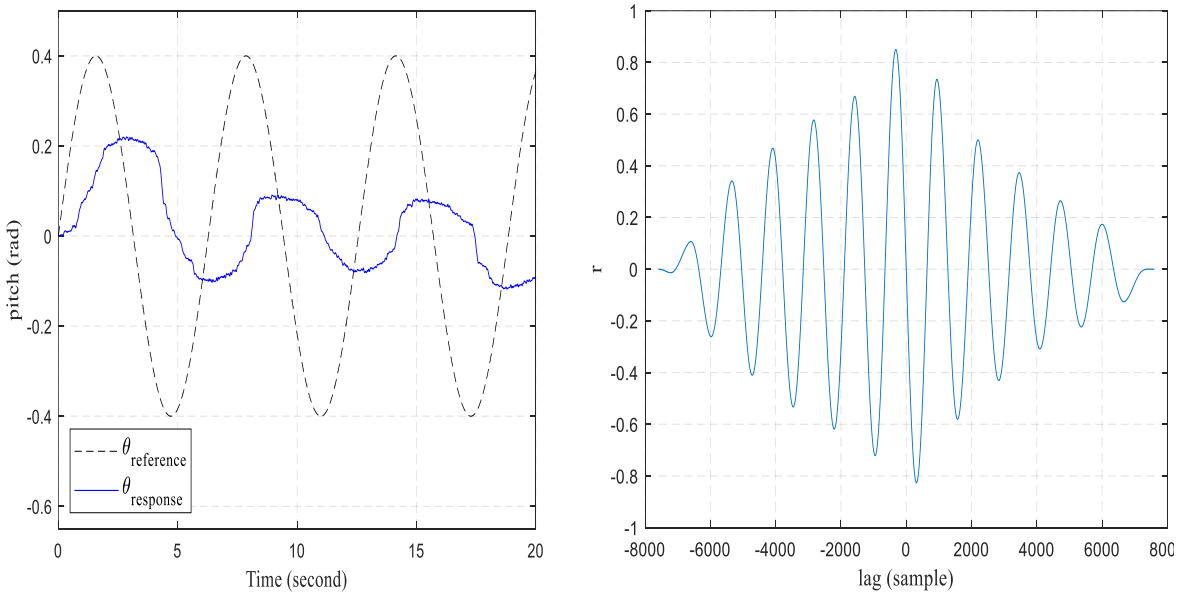


b The wind disturbance 3.2m/s

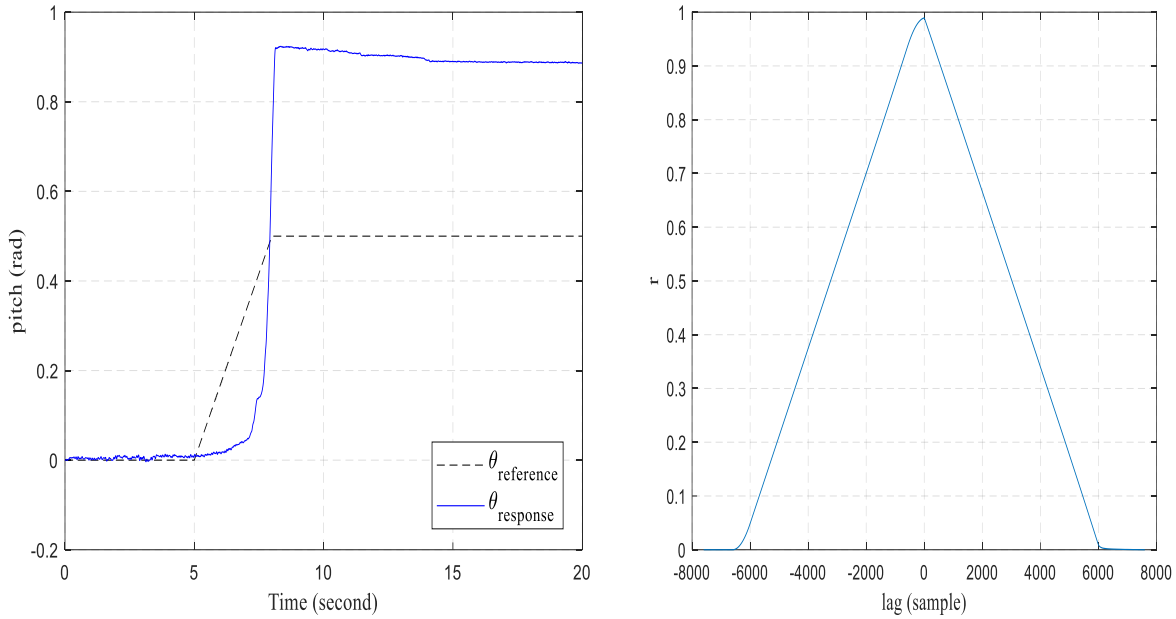


c The wind disturbance 4.2m/s

Figure IV.6 The wind disturbance.



a sinusoidal reference 1 rad/s and wind disturbance 4.2m/s



b ramp reference and wind disturbance 4.2m/s

Figure IV.7 The wind disturbance with tracking different reference command

IV.5 High Dynamic Performance Control System

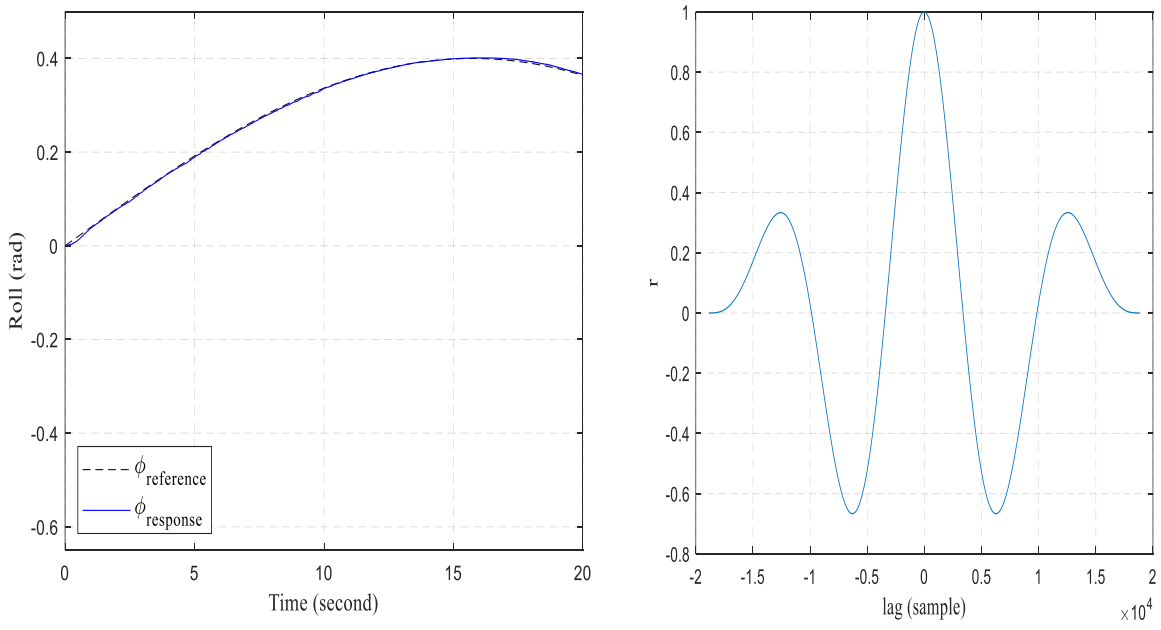
The purpose of the command is to orient the quadrotor in order to follow a predefined trajectory. in this section we will modify impedance control in block fly control in project asbQuadcopterStart by HDP control[87-89], this modification will serve us to reject external disturbances while guaranteeing the stability of the system as we will demonstrate below.

The quadrotor is a system under actuated, since the number of input signals (voltage of the motors which generate the speeds ω_i) is lower than the number of quantities to be controlled: three in rotation (ψ, θ, ϕ,).

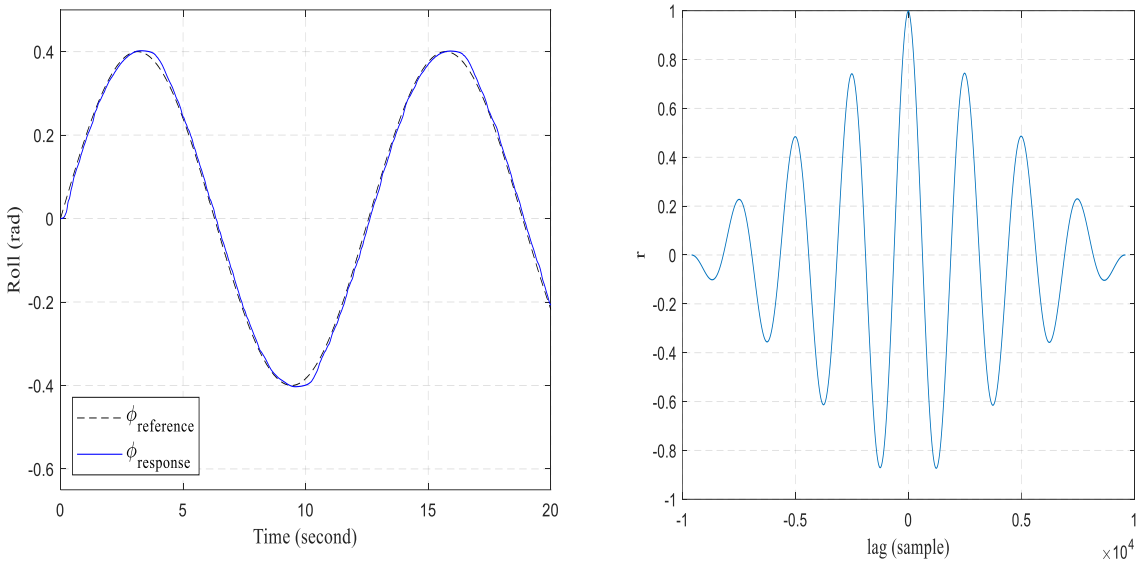
IV.6 Experiment Results

IV.6.1 Result experiment for Roll

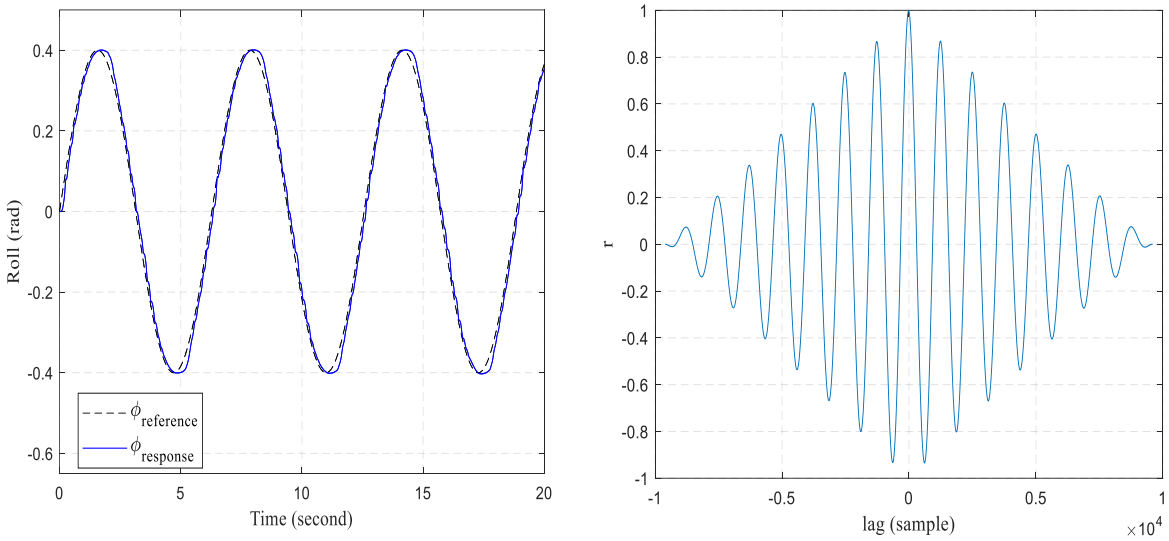
The first experiment was run to determine the control system's ability to track sinusoidal commands, the sinusoidal commands were selected to ensure the system is capable of circular maneuvers with increasing frequencies (i.e., 0.1 to 3 rad/s) figures IV.8. The control system was able to track the commands with high accuracy (maximum MAE= 0.4 rad). Moreover, the figures on the right, show the cross-correlation between the command and quadcopter response in all cases cross correlation at a maximum value around 2 samples lag. The correlation declines when the signal is shifted to the right or left in a periodic fashion. A similar, result is found when the control system was tested pitch (see figures IV.11). To show system ability to track reference with big tilt values (more than 10 degrees), a ramp command was generated to prove system stability around 90 degrees command. The quadcopter reached 90 degree after 11 seconds and maintain the stability until the experiment was stopped as is seen in figure IV.8.f. The cross-correlation shows the system is able to track the command with no lag, and the cross-correlation declines in a linear style when the signal is shifted to the lef/right. Similar, result is found when the control system was tested pitch (see figures IV.11.f).



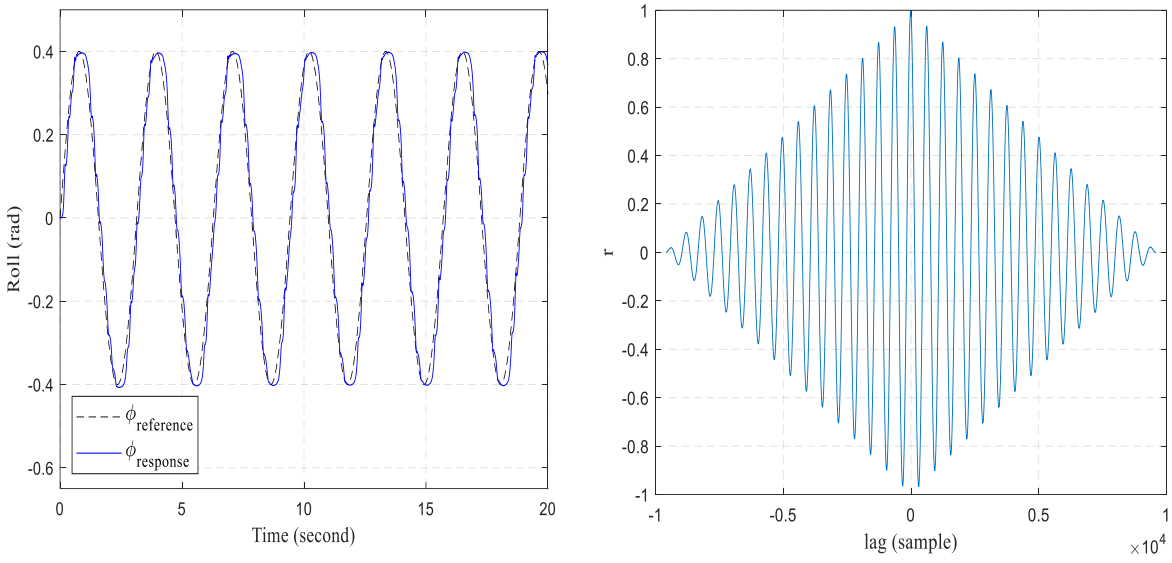
a sinusoidal reference 0.1 rad/s



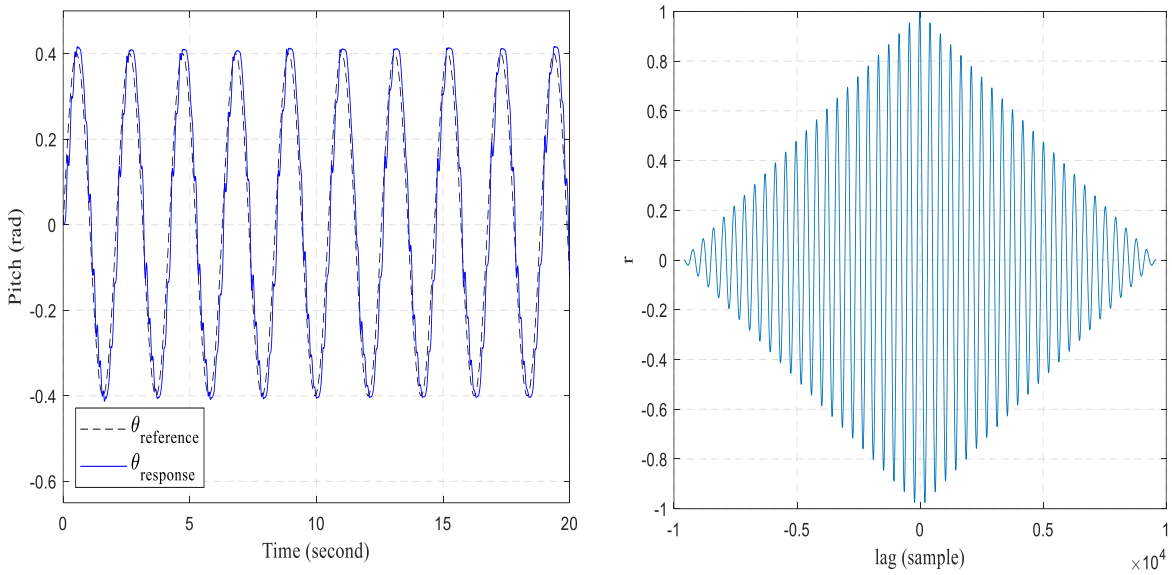
b sinusoidal reference 0.5 rad/s



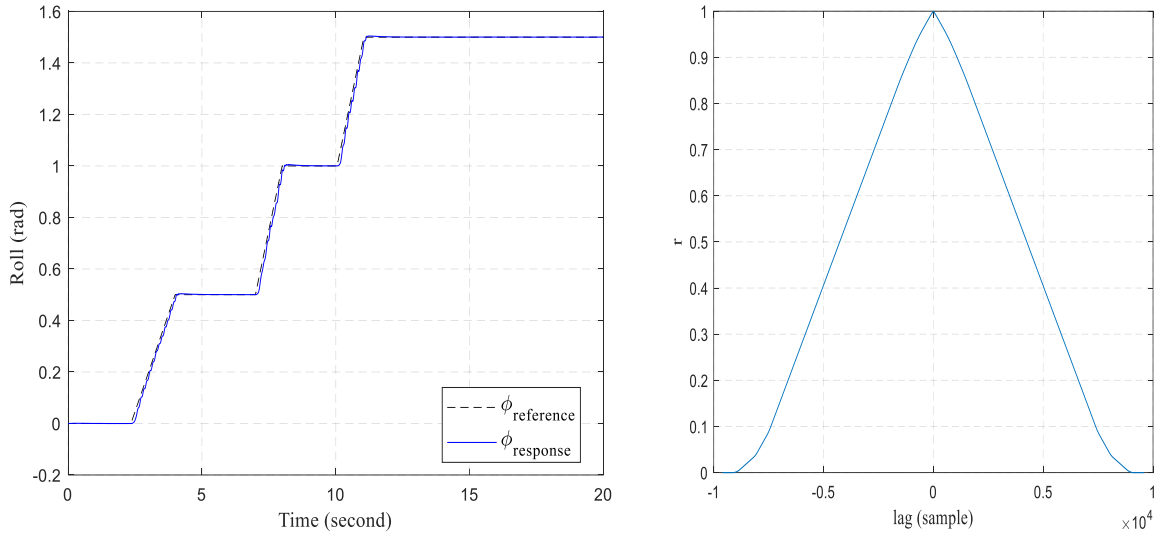
c sinusoidal reference 1 rad/s



d sinusoidal reference 2 rad/s



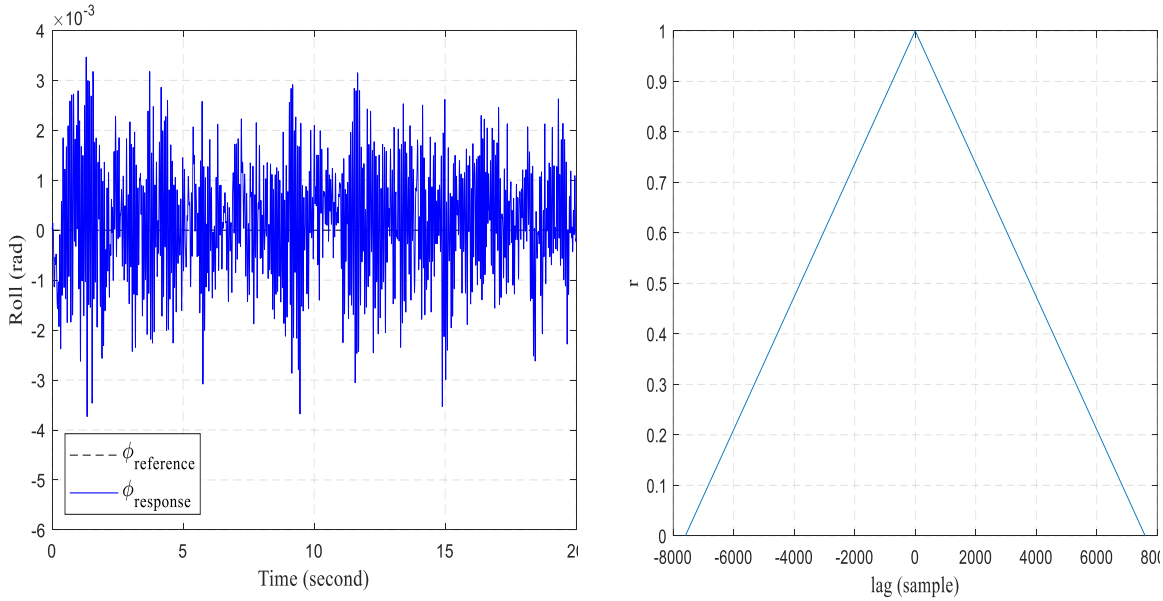
e sinusoidal reference 3 rad/s



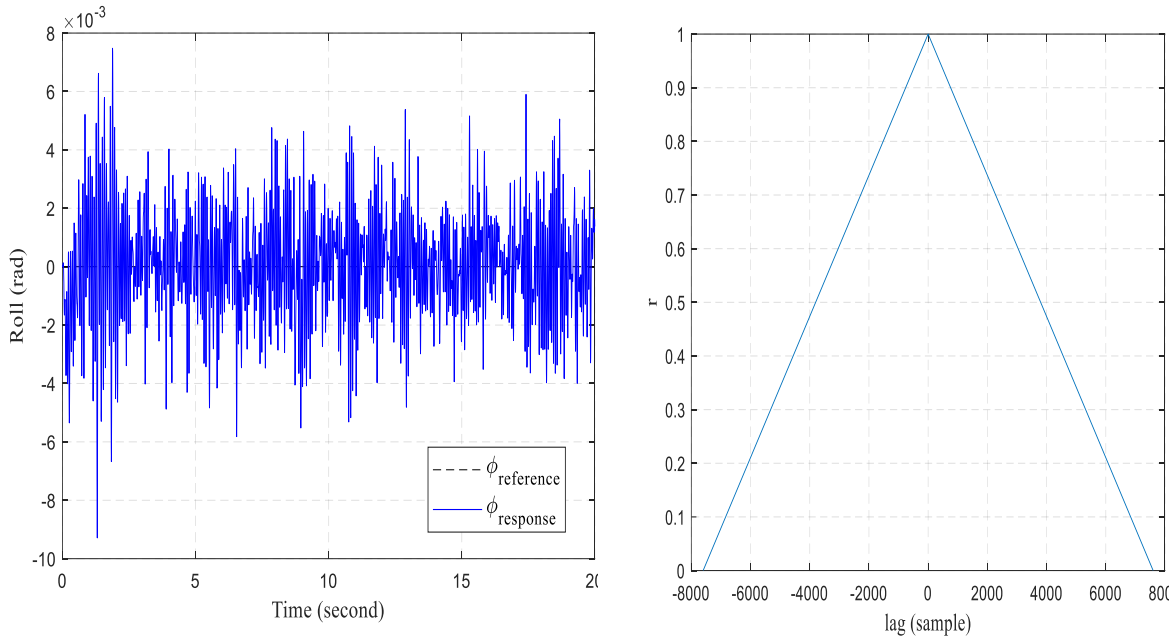
f ramp reference.

Figure IV.8 track different frequencies command a sinusoidal reference

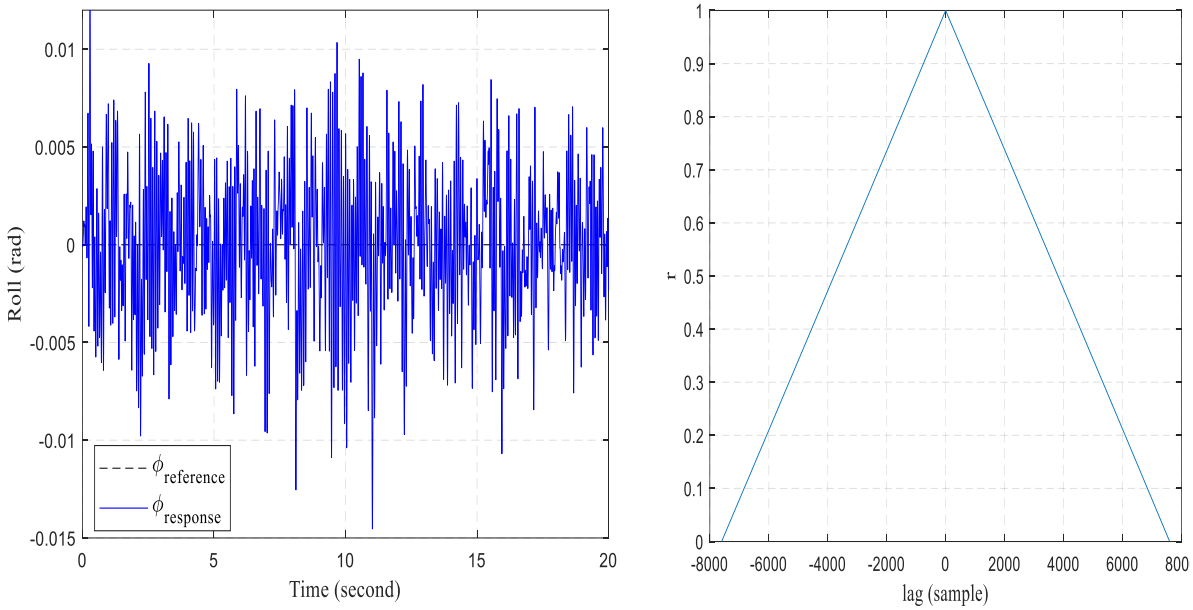
The performance of the control system with mechanical perturbation was tested with a concentrated wind source with variable speeds (2.2, 3.2, and 4.2 m/s). The control system shows good performance with a maximum tracking error of 0.086 degrees at 4.2 m/s wind speed sees Figure IV.9. Furthermore, the system shows good tracking for sinusoidal and ramp command with maximum wind disturbance (4.2 m/s) show Figure IV.10. Similar, result is found when the control system was tested pitch (see figures IV.12 and figures IV.13).



a The wind disturbance 2.2m/s

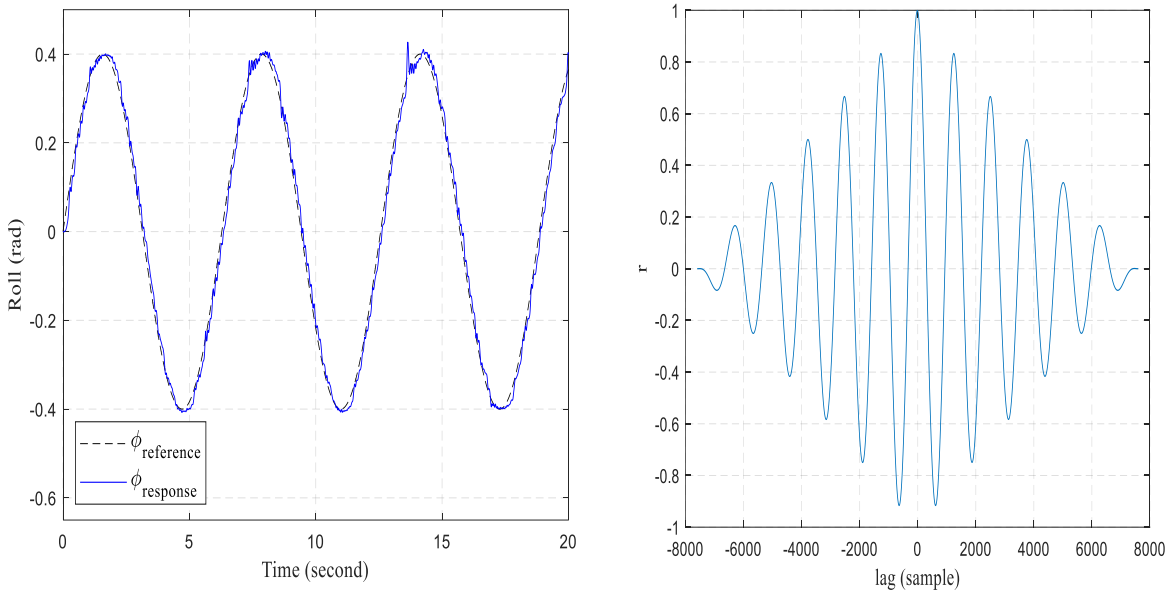


b The wind disturbance 3.2m/s

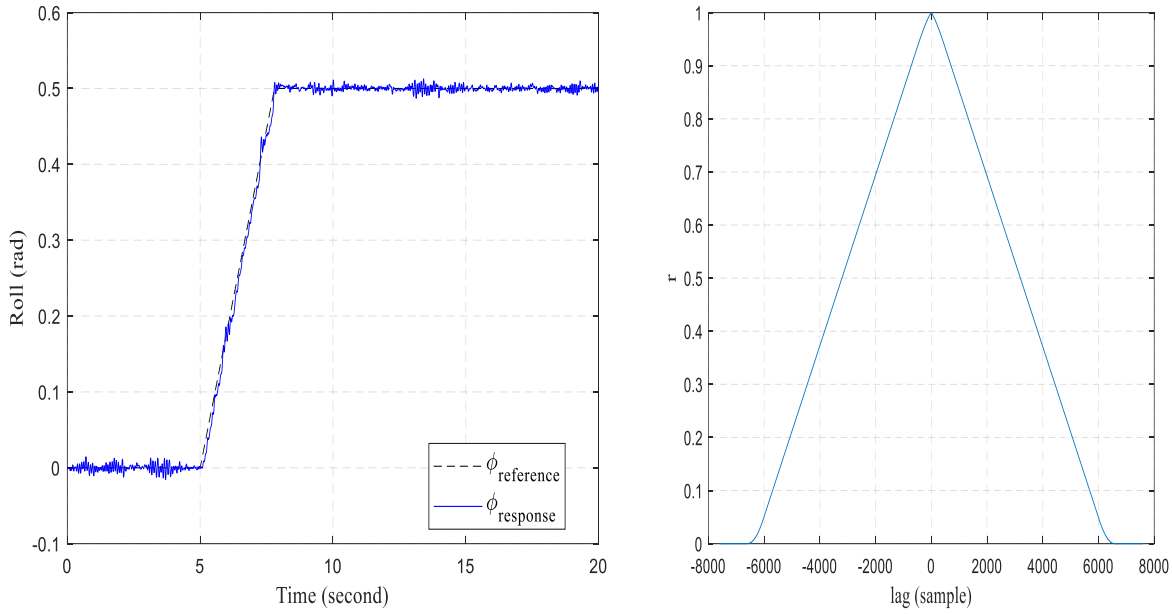


c The wind disturbance 4.2m/s

Figure IV.9 The wind disturbance.



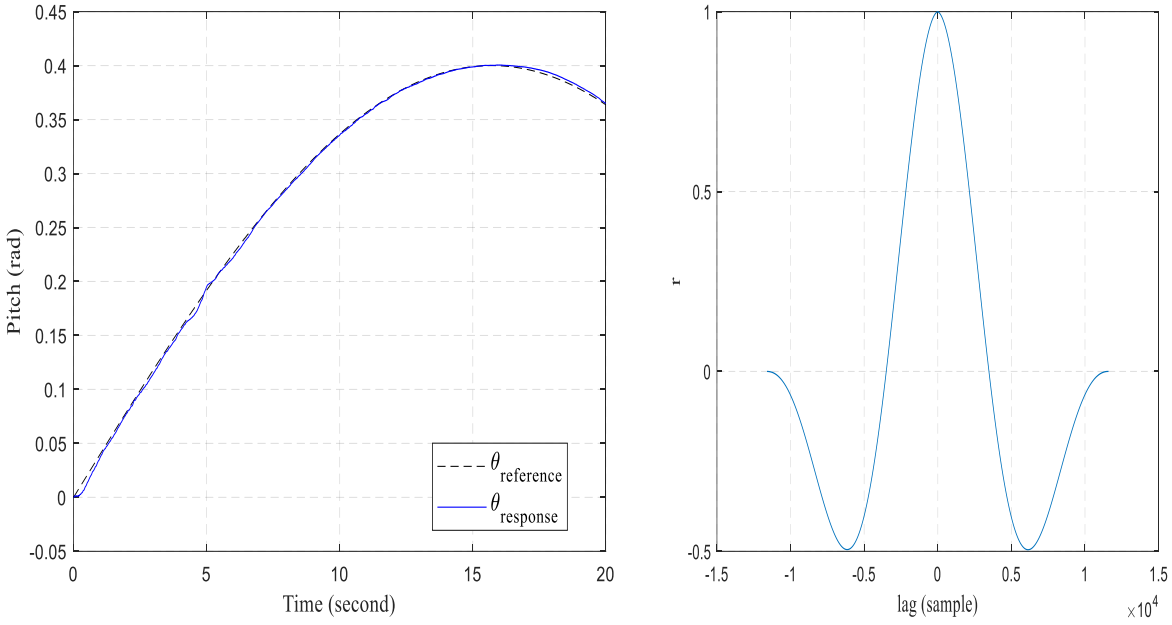
a sinusoidal reference 1 rad/s and wind disturbance 4.2m/s



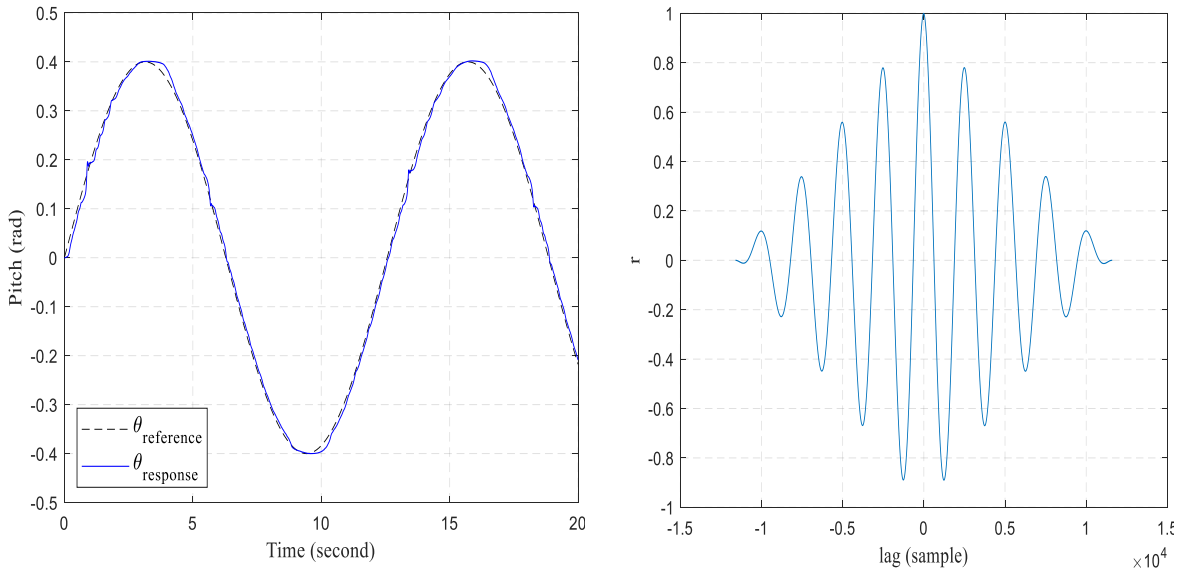
b ramp reference and wind disturbance 4.2m/s

Figure IV.10 The wind disturbance with tracking different reference command

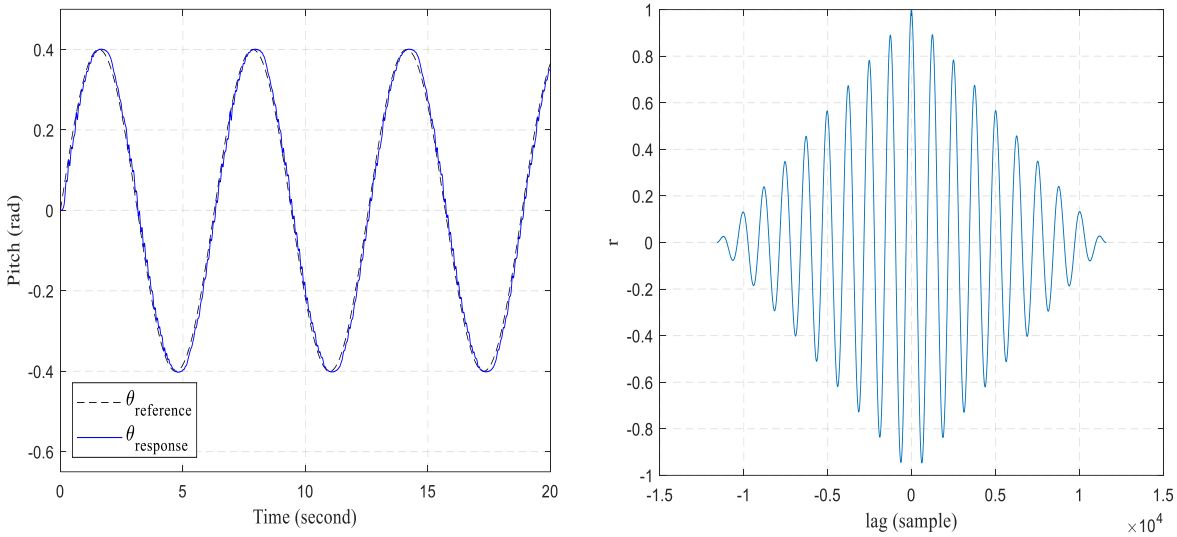
IV.6.2 Result experiment for Pitch



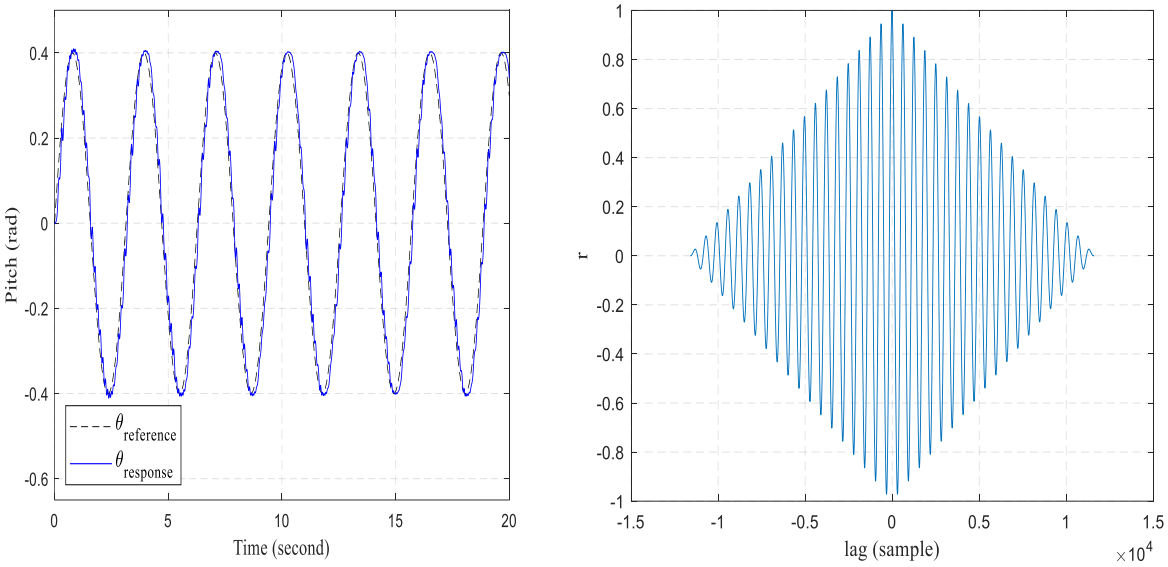
a sinusoidal reference 0.1 rad/s



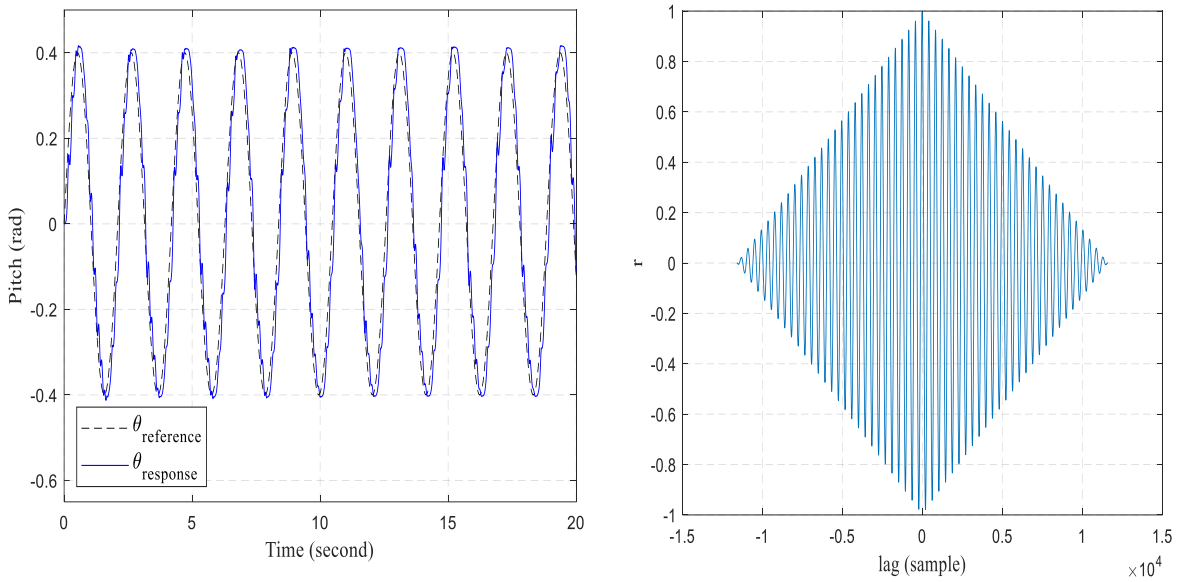
b sinusoidal reference 0.5 rad/s



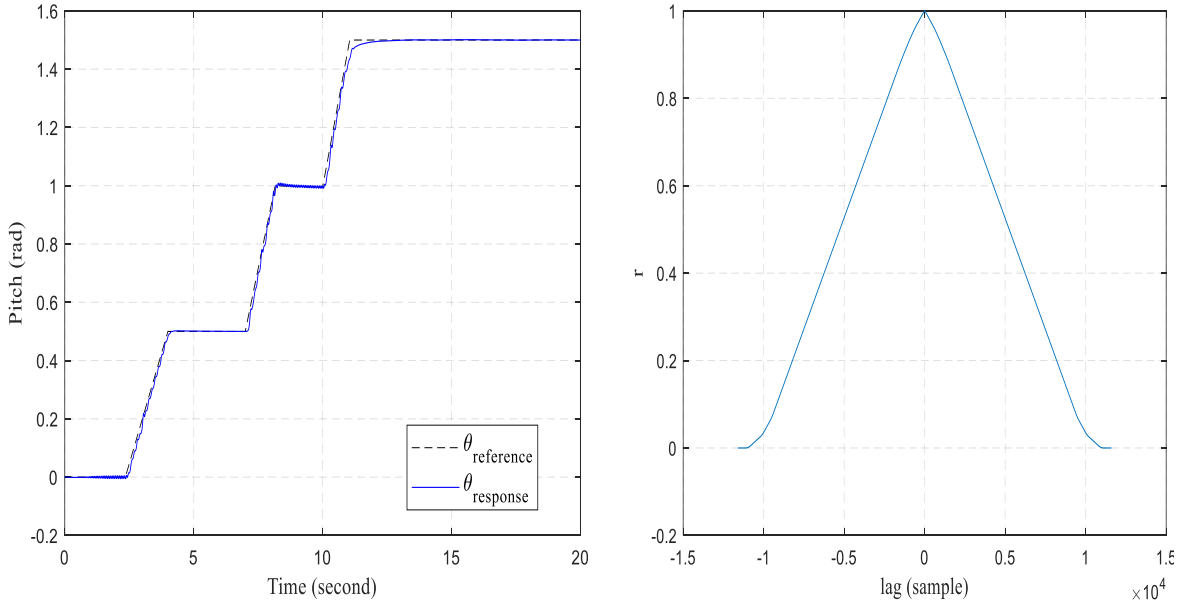
c sinusoidal reference 1 rad/s



d sinusoidal reference 2 rad/s

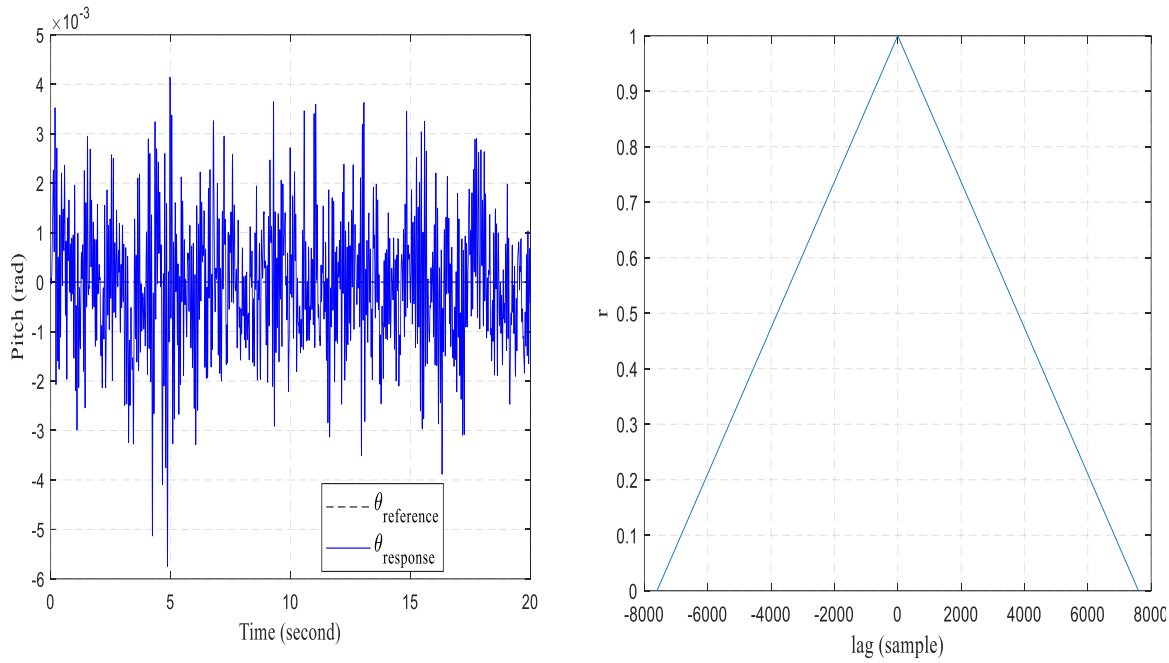


e sinusoidal reference 3 rad/s

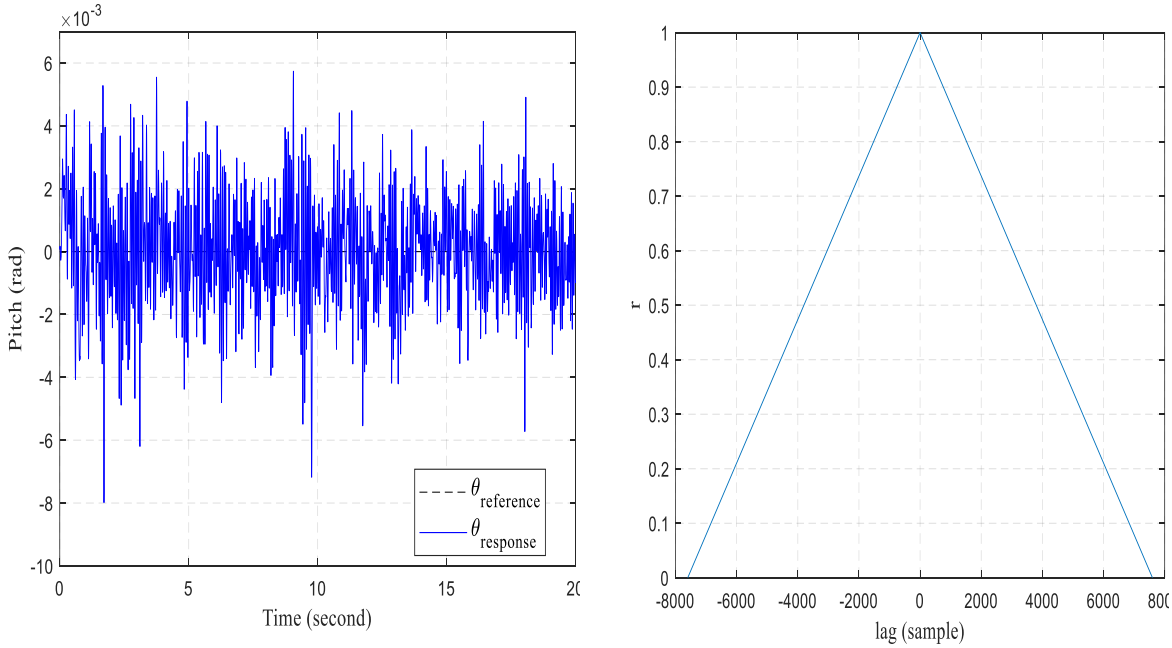


f ramp reference.

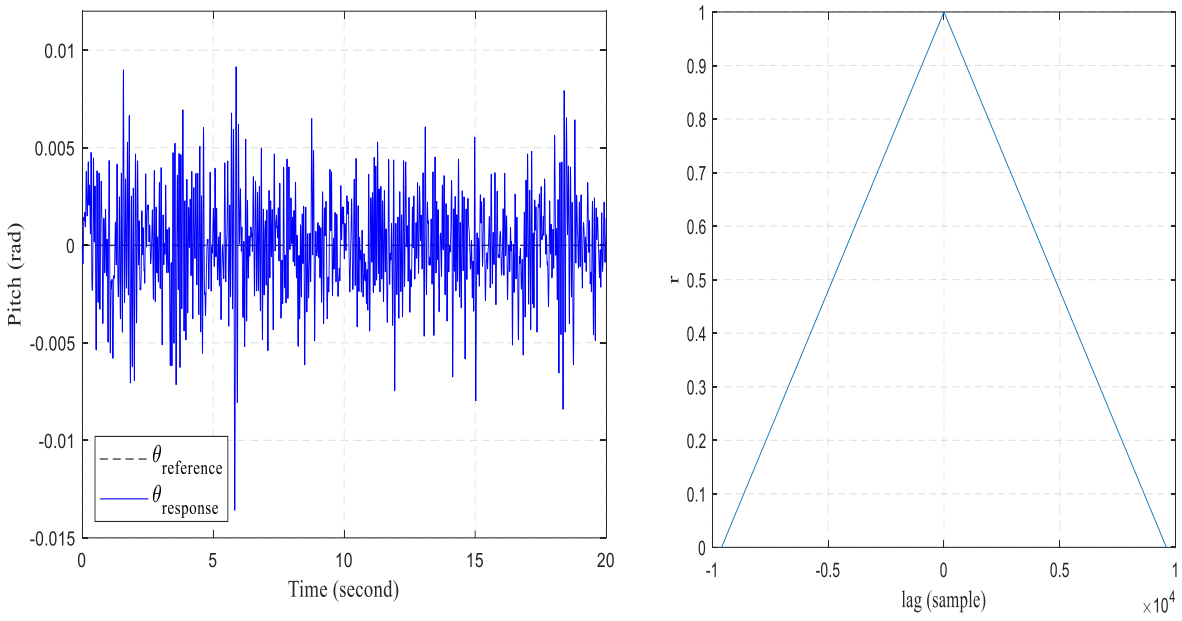
Figure IV.11 track different frequencies command a sinusoidal reference and ramp reference.



a The wind disturbance 2.2m/s

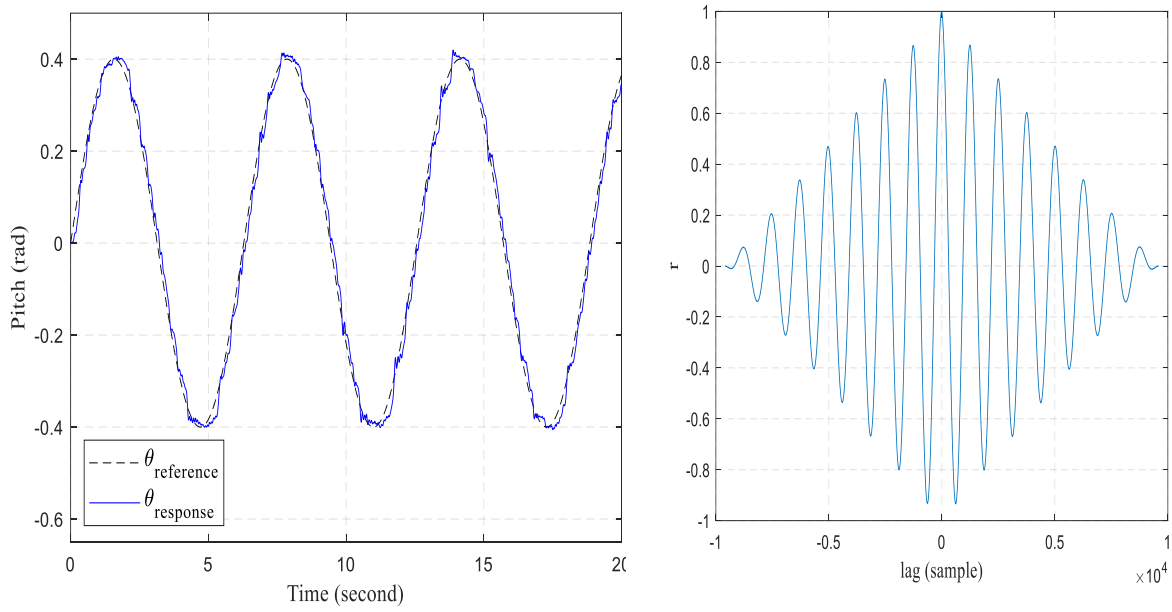


b The wind disturbance 3.2m/s

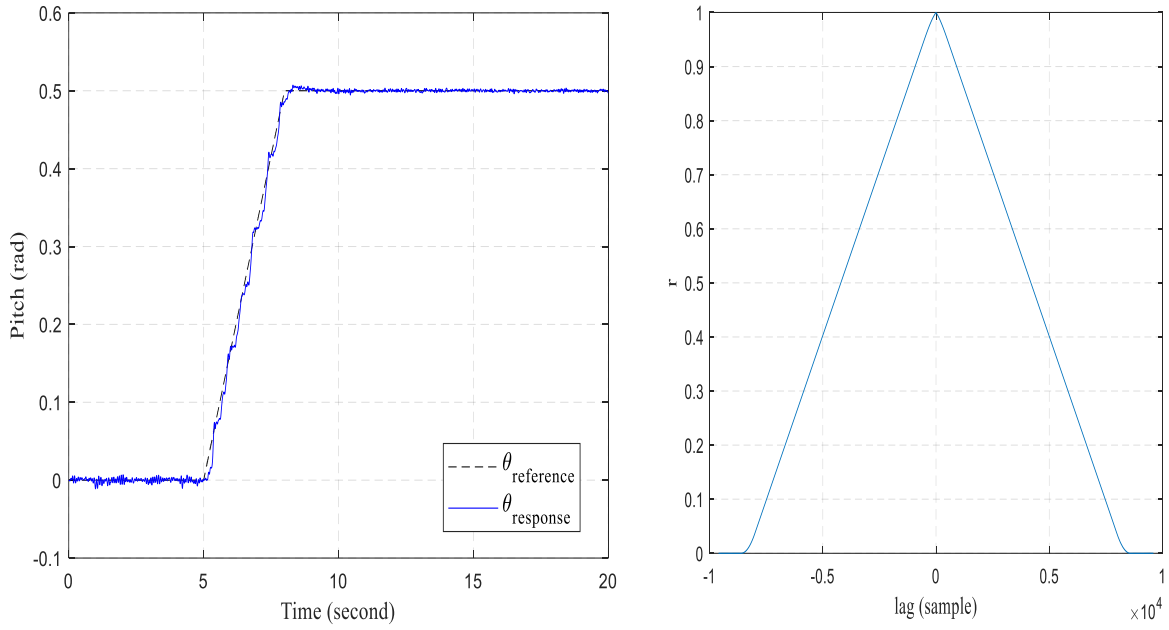


c The wind disturbance 4.2m/s

Figure IV.12 The wind disturbance.

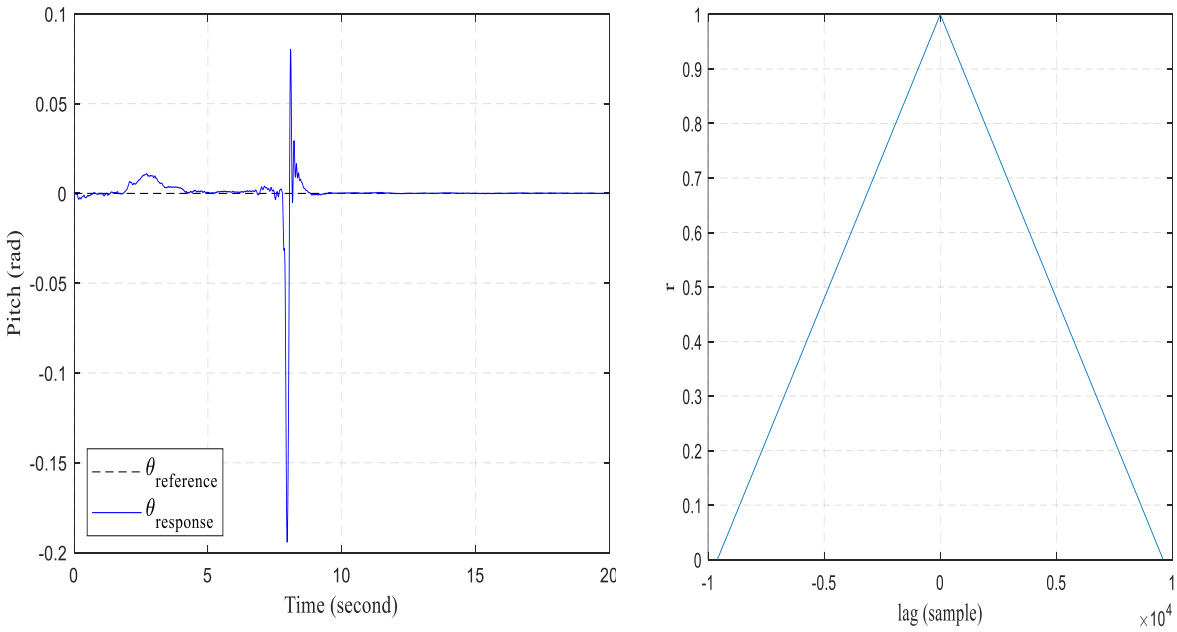


a sinusoidal reference 1 rad/s and wind disturbance 4.2m/s

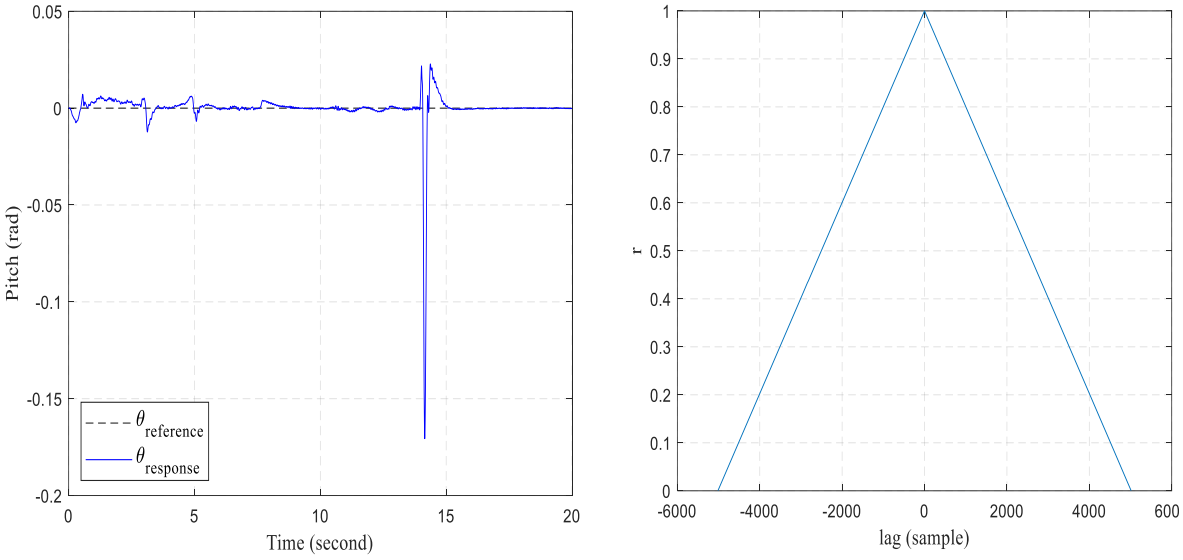


b ramp reference and wind disturbance 4.2m/s

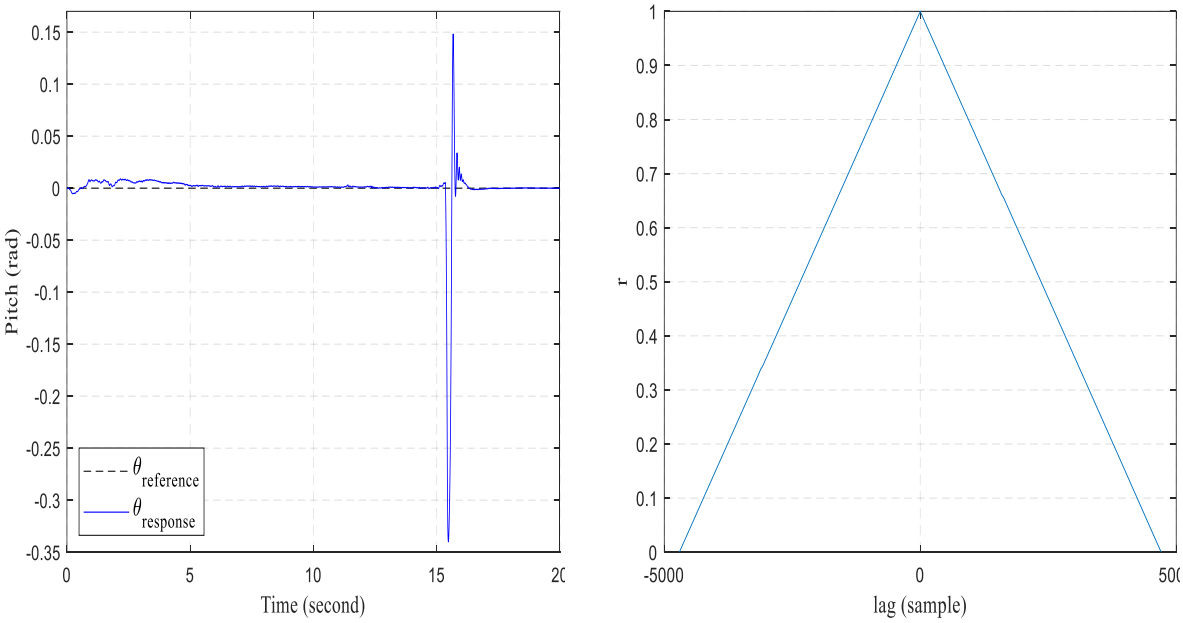
Figure IV.13 The wind disturbance with tracking different reference command



a added the mechanical loads 60g



b added the mechanical loads 75g



a added the mechanical loads 95g

Figure IV.14 added the mechanical loads and suddenly released

Final test is parashooting packages with zero angle command Figure IV.14, for this test three packages were selected (60, 75 and 95 g). The system was able to maintain the system stability after the loads were released. For the maximum load the system was disturbed with 30 degrees diversion from 0 position, and the quadcopter returned to 0 position after (1.80 seconds)

Table IV.1 comparative study between HDP and IC with the cross-correlation average r

			Roll				Pitch			
			HDP		IC		HDP		IC	
Type	Specification		RMSE	r /lag	RMSE	r /lag	RMSE	r /lag	RMSE	r /lag
0.4sin (ωt)	ω (rad/s)	0.1	0.3334	1/-14	16.6703	0.8269/0	0.2146	1/0	10.3860	0.9687/ -209
		0.5	1.1211	0.9993/0	33.0578	0.7245/-430	1.0532	0.9995/-7	30.1822	0.8851/ -428
		1	2.1456	0.9989/-12	30.7564	0.2069/342	1.8385	0.9989/-9	30.9070	0.8903/ -307
		2	4.4368	0.9971/-13	28.6265	0.0678/-140	3.8452	0.9987/ -12	28.3574	0.4063/ -136
		3	6.7652	0.9965/-14	28.1481	0.0837/-61	6.3181	0.9955/-13	28.1885	0.3553/ -83
Ramp	N/A		0.8049	1/0	10.1467	0.9155/1866	0.7613	1/0	12.2290	0.9965/0
Mechanical load/unload	Load (g)	60					0.8790	1/0		
		75					0.9784	1/0		
		95					2.6280	0.9997/0		
Wind disturbance	Speed (m/s)	2.2	0.1010	1/0	1.3762	1/0	0.1228	1/0	0.5843	1/0
		3.2	0.1769	1/0	1.1151	1/0	0.1524	1/0	0.5372	1/0
	.5sin (t)	4.2	0.3318	1/0	0.6722	1/0	0.2386	1/0	0.6611	1/0
			2.0960	0.9993/-13	29.8457	0.2803/866	2.3390	0.9982/-11	30.3972	0.8500/-319
	0.5 (rad)		0.4936	0.9999/0	29.3342	0.9380/ -39	0.4180	1/0	35.2364	0.9893/0

As shown in table VI.1 both control systems were tested for an increasing frequency reference as describe in the experiments designs. The position control system with HDP shows high resilience for high and low frequencies commands the cross-correlation between the reference and system response were on average r value **0.998** at **11** samples lag and **0.998** at **8** samples lag for rotation around x and y, respectively. In contrast, the impedance controller was

able to track low frequencies and at 1 rad/s command the control system acted to the reference as a disturbance and stabilize around an equilibrium with minimum error the cross-correlation between the reference and system response at low frequencies were on average r value **0.778** at **215** samples lag and **0.923** at **318** samples lag for rotation around x and y, respectively. Moreover, table 2 shows the superiority of the hierarchical position control for high precision tracking with average RMSE Appendix C equal to **2.96** (rad) compare to **27.45** (rad) when the impedance control system was implemented for rotation around x.

HDP control respond to mechanical disturbance is reported in experiment three where a mechanical loading and unloading on the system in which the system must keep tilt angle at zero. The system was able to respond robustly see figures IV.14. In this experiment, the behavior of the control system can be distinguished from the HDP control. The HDP react aggressively to unloading which can lead to unstable behavior while all the coupling components are affecting the system in real system operation.

Final experiment is designed to study system behavior under constant wind where the flow speeds were selected, the RMSE increases proportionally with the target speed increase for both degree of freedoms. The maximum error is **0.3318** and **0.2386** percent on roll and pitch, respectively.

IV.7 Conclusion

Two controls laws were dealt in this chapter, the impedance control suggested by the manufacturer, where we showed them to be ineffective in terms of robustness and tracking, Although, it is safer when the device is operating in a environment in which it is afraid of collision. Then we moved to High Dynamic Performance control of mambo parrot, The experimental results of the control of the high dynamic performance of roll and pitch show the high efficiency of this control, in terms of robustness and traceability.

Chapter V

Backstepping Geometric Control of Attitude for Quadrotor

V.1 Introduction

As we mentioned earlier, the efficiency of Backstepping geometric control in avoiding the singularity and ambiguity, also shows its efficiency when navigation depends on tracking wide angles, and this is what we will show in the simulation, the impedance control system has some limitations (i.e., tracking high-frequency command and load distance). In this chapter, the alternative control system was designed and implemented to show the limitation of the impedance control and focus on the performance of nonlinear control theory (i.e., geomagnetic backstepping control method).

V.2 Wide angles tracking

The control systems were tested for three degrees of freedom (roll, pitch, and yaw), as shown in Figure V.1, In order to test wide-angles tracking using geometric backstepping.

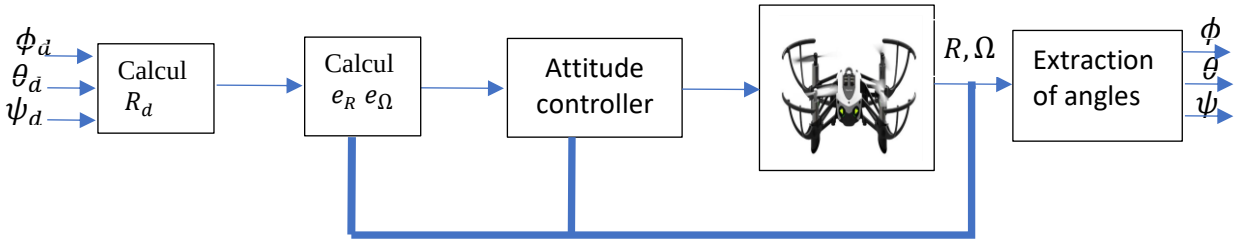


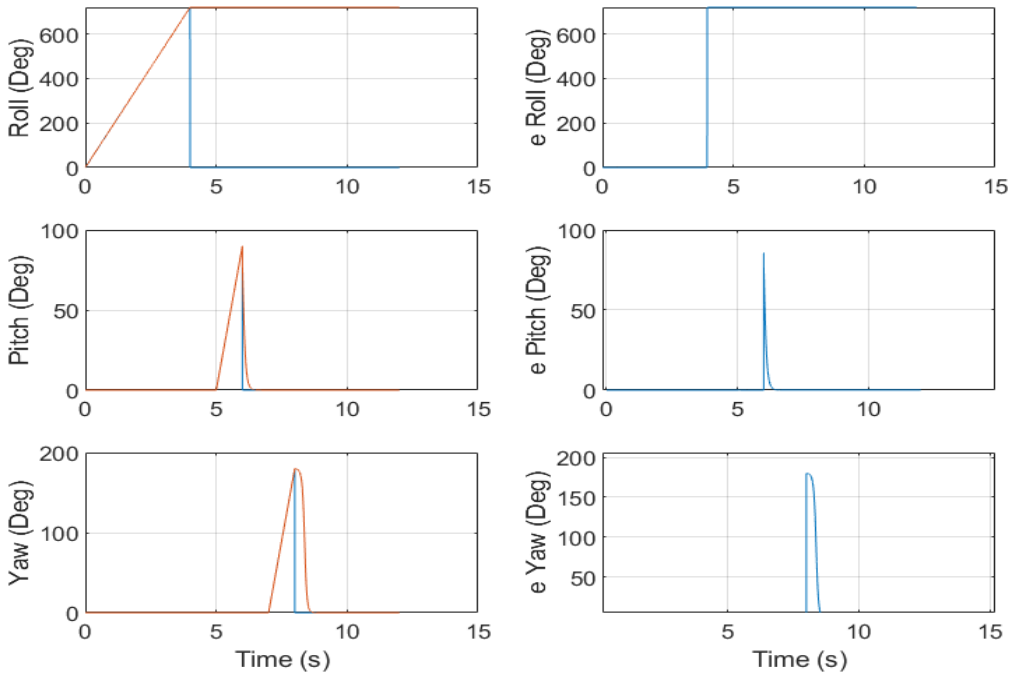
Figure V.1: Quad-rotor attitude control strategy

The initial hovering conditions are: $\Theta(0) = \Omega(0) = [0,0,0]^T$, $R(0) = I_{3\times3}$

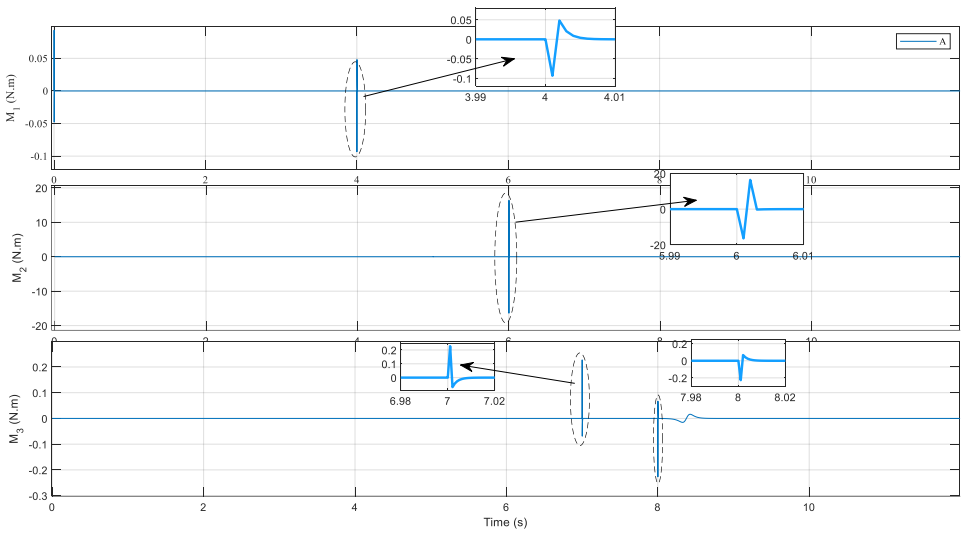
The desired trajectory $\Theta_d = [\phi_d,\ \theta_d, \psi_d]^T$ varies as follows:

First, the desired roll ϕ_d is ramp signal with inclination equal 720 in t = [0s 5s] after taking zero, the desired roll θ_d is ramp signal with inclination equal 90 in t = [5s 6s] after taking zero, the desired roll ψ_d is ramp signal with inclination equal 180 in t = [7s 8s] after taking zero,

V.3 RESULTS AND DISCUSSION



a. Altitude and its error



b- Different moments

Figure V.2 Flipping using BG control Without changing the parameters

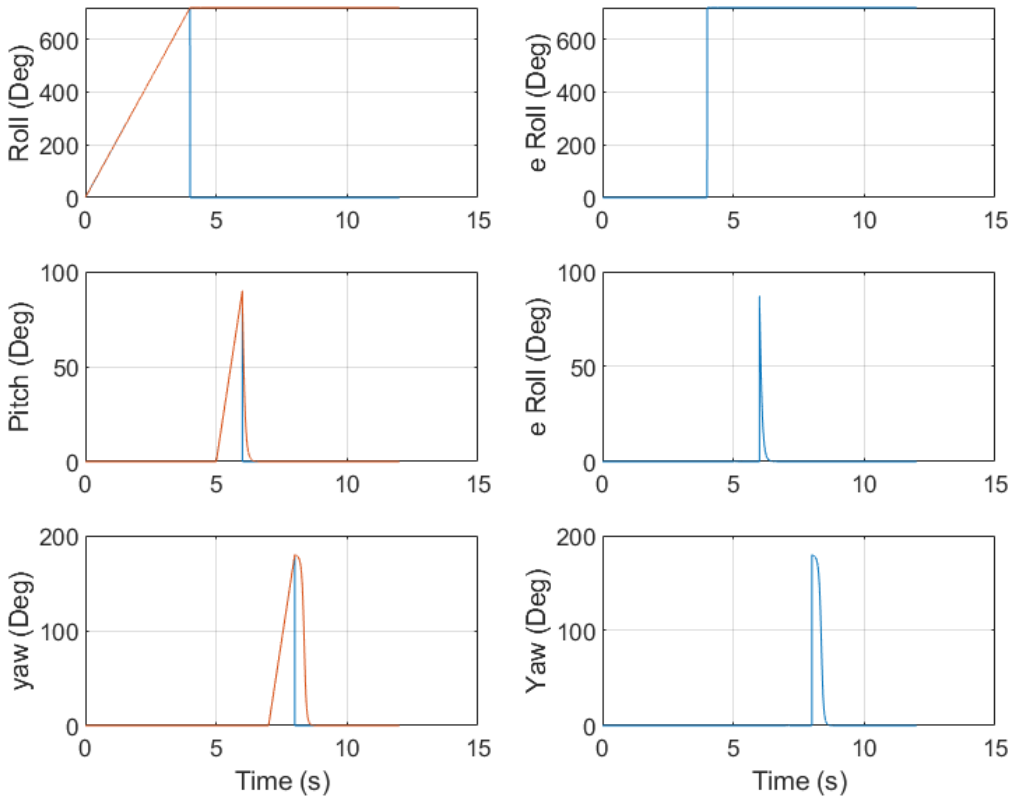


Figure V.3 Flipping using backstepping control with we apply an external force F_{ex}.

Simulation results of Figure V.2.a on the right side show the attitude of the components in three directions, which are almost the same as the desired values, and on the left side show its error, which is defined as $e_1 = \Theta - \Theta_d$. The components in the three directions converge to the desired states or stable states. We observed the roll, where it converges to the desired trajectory at small-time, pitch converges at t=1s, while Yaw converges in almost 2.5s. Figure V 3.b clearly shows the input control *M*. Figure V.3 shows the trajectory of the quadrotor after changing the mass and the inertia.

V.4 Geometric backstepping control of quadrotor

The purpose of this test is to orient the quadrotor in order to follow a predefined trajectory. in this section we will modify Impedence control in block fly control in project asbQuadcopterStart by geometric backstepping [90], this modification will serve us to reject external disturbances while guaranteeing the stability of the system as we will demonstrate below.

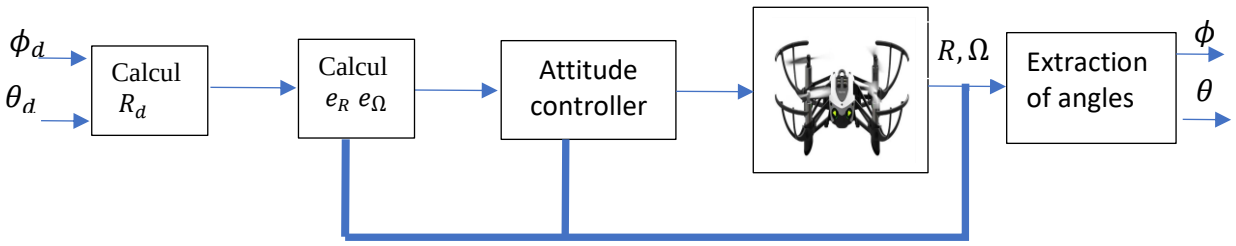
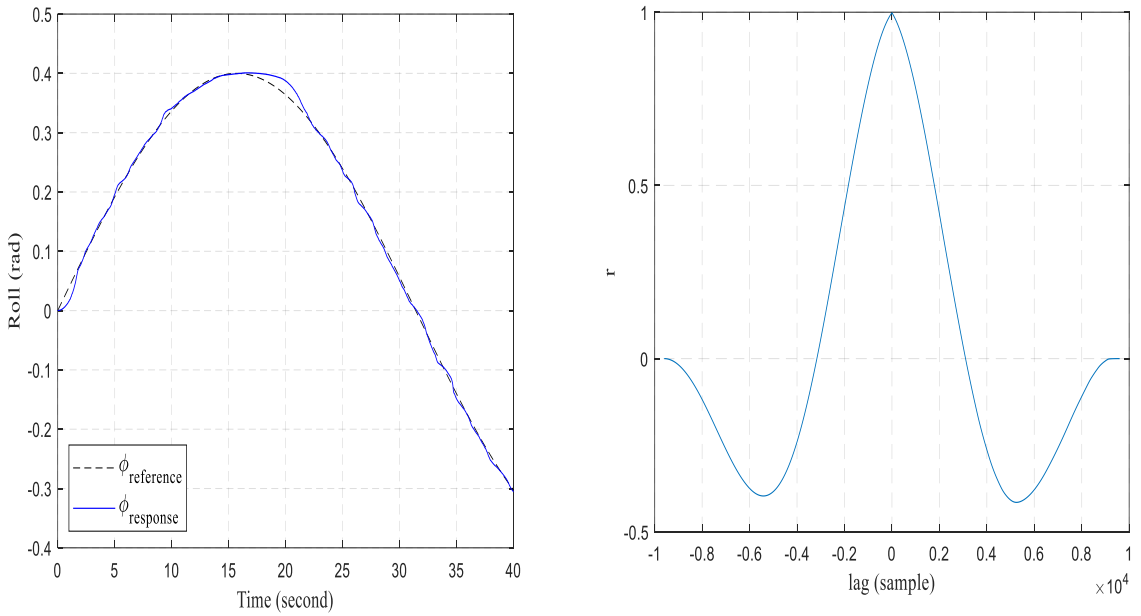


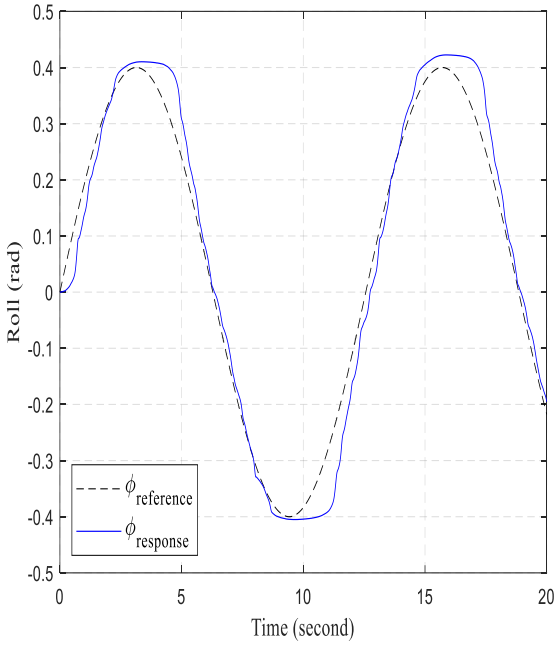
Figure V.4: Quad-rotor attitude control strategy

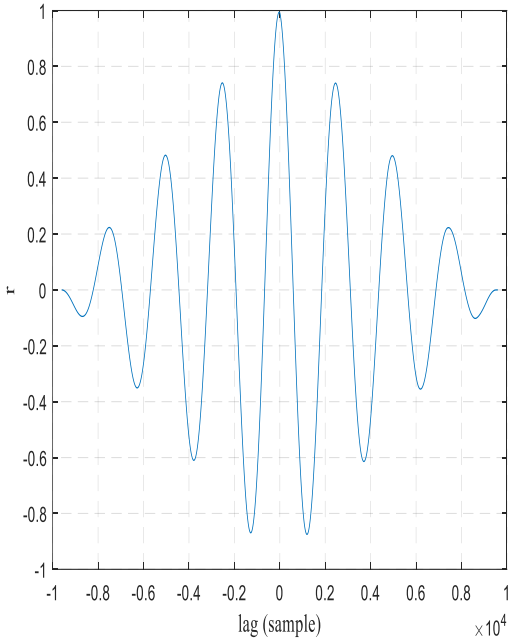
V.6 Experiment Results

V.6.1 Result experiment for Roll

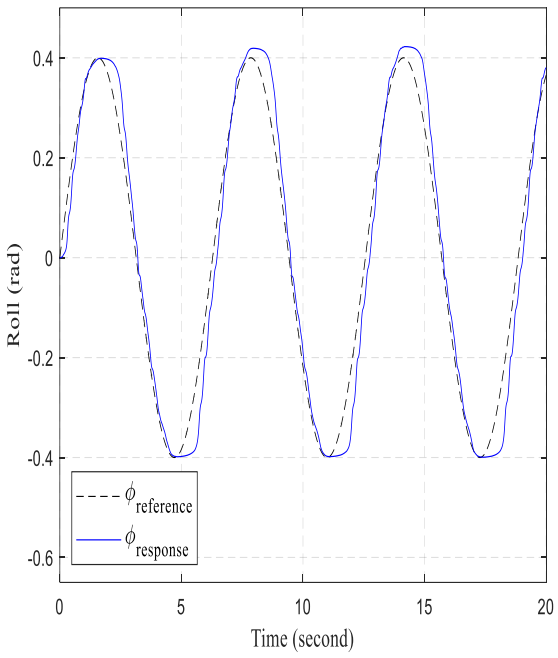


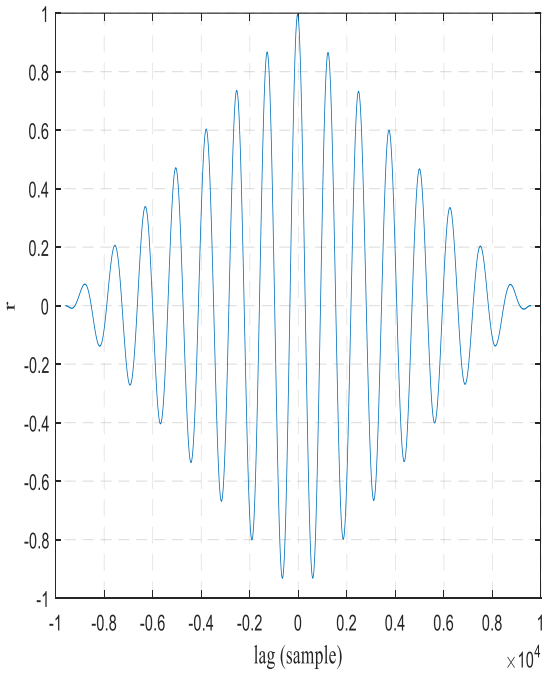
a sinusoidal reference 0.1 rad/s



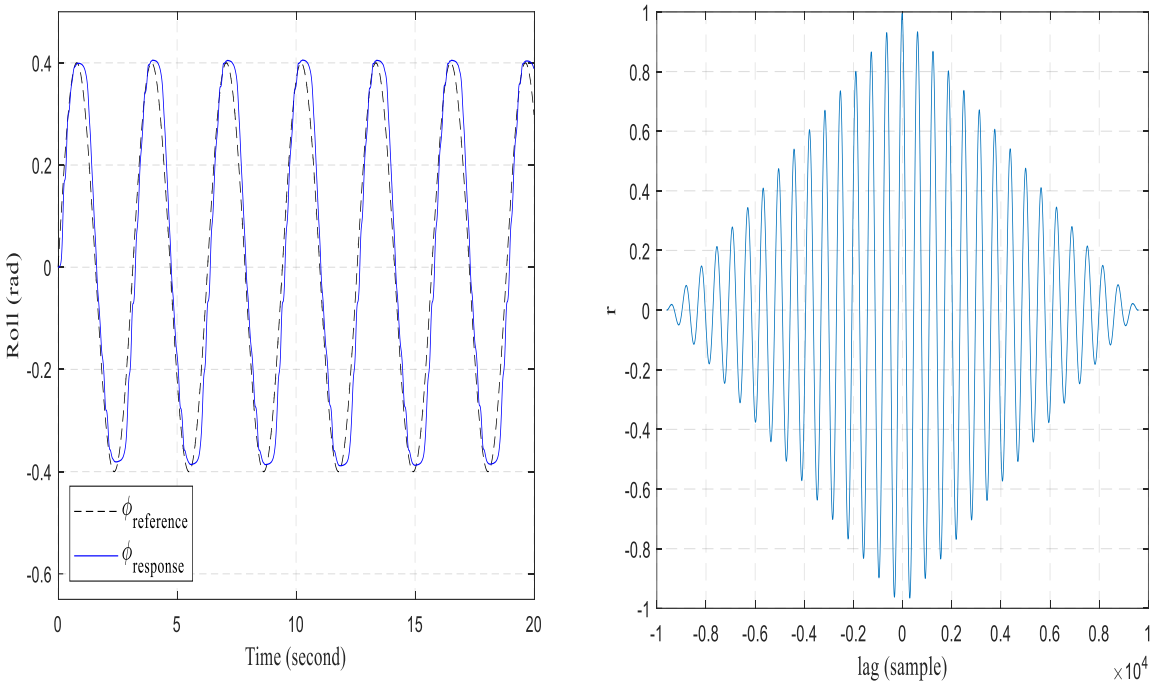


b sinusoidal reference 0.5 rad/s

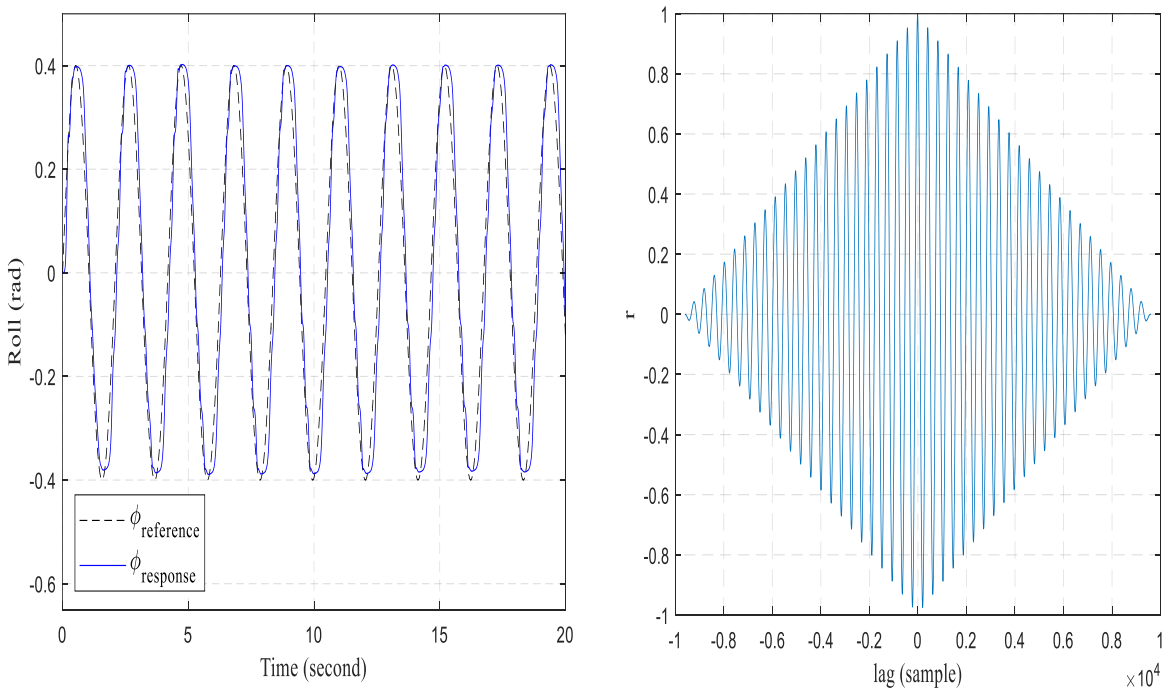




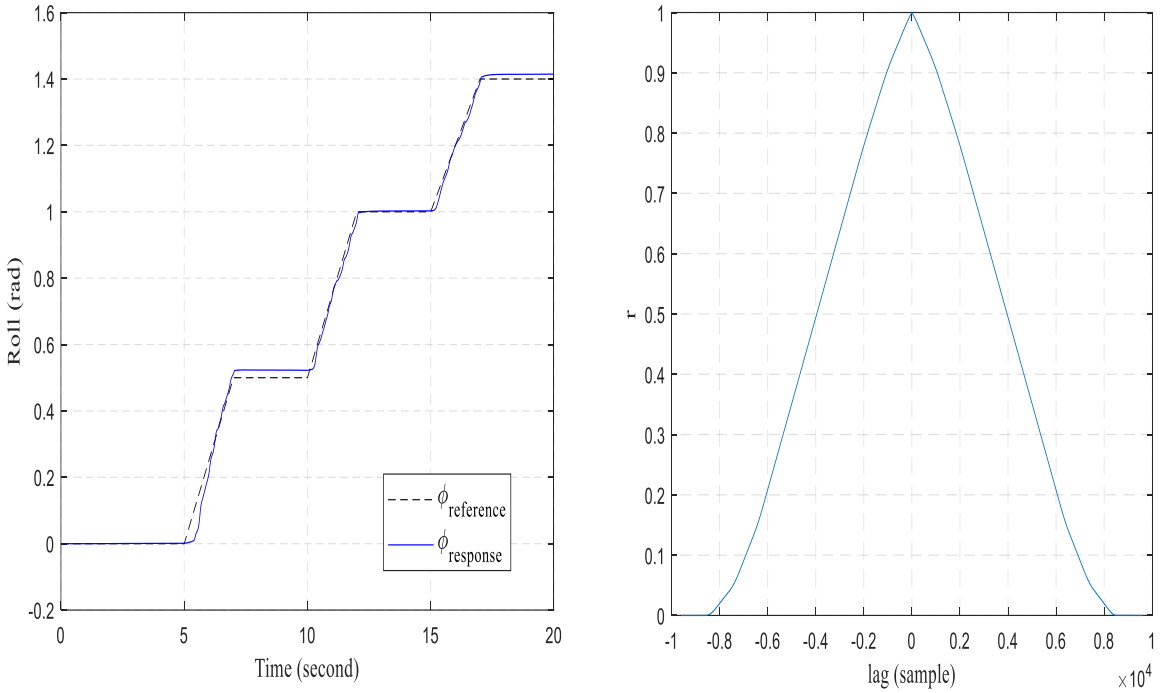
c sinusoidal reference 1 rad/s



d sinusoidal reference 2 rad/s



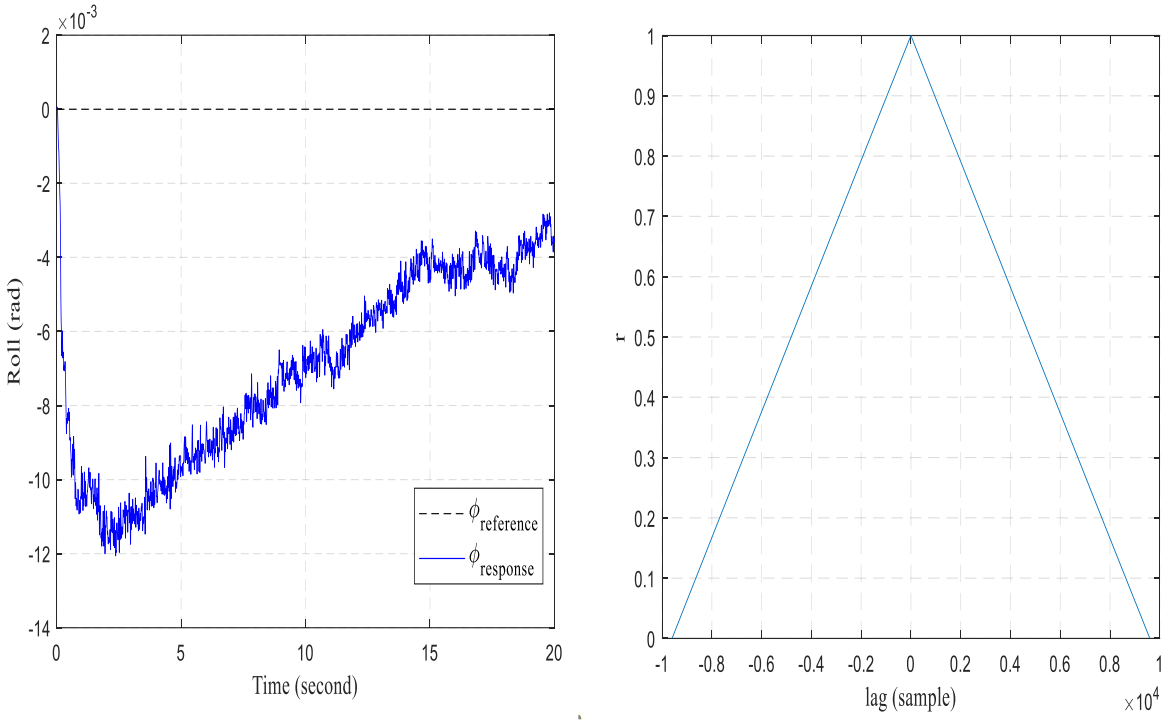
e sinusoidal reference 3 rad/s



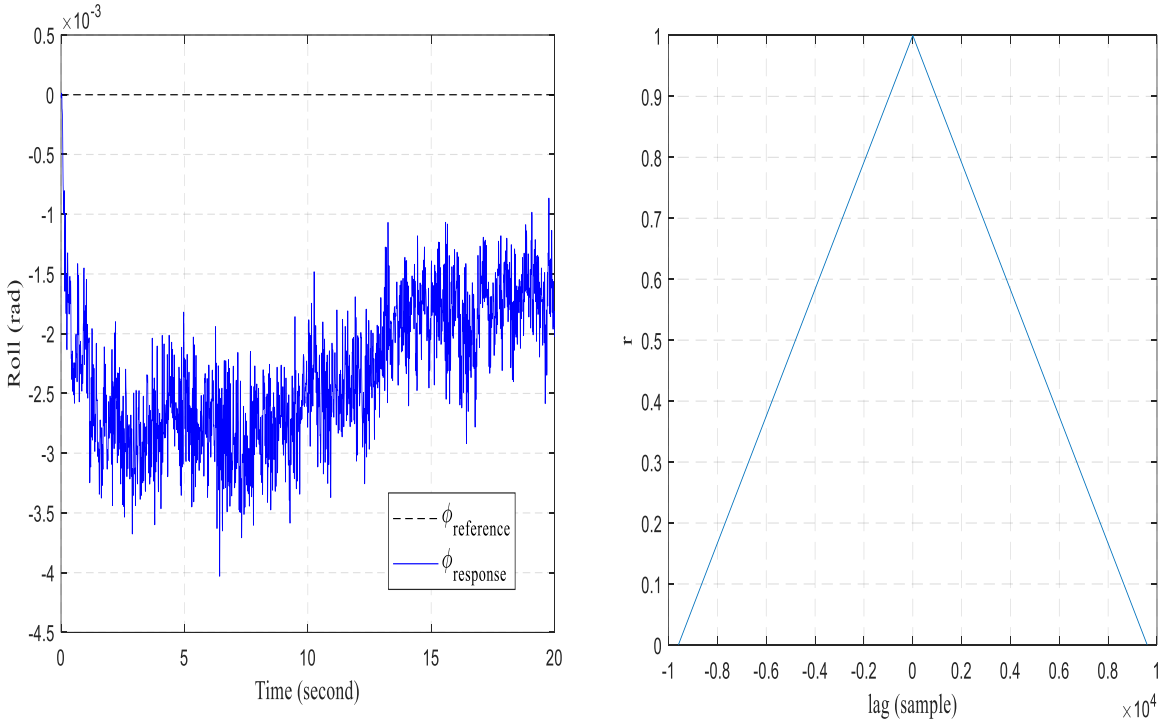
f ramp reference.

Figure V.5 track different frequencies command a sinusoidal reference and ramp reference.

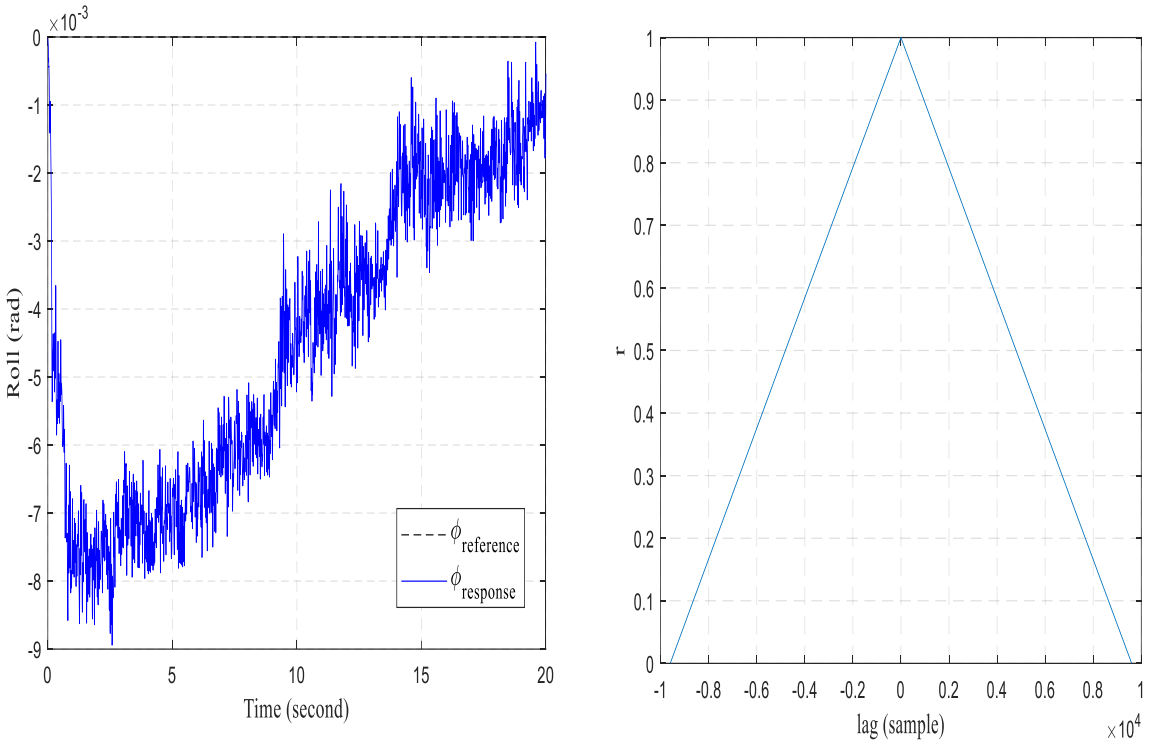
In this experiment, we replaced the impedance control with the geometric backstepping control, to determine the ability of the control system to keep track of sinusoidal commands, were selected to ensure that the system is capable of circular maneuvers at increasing frequencies (i.e. 0.1 to 3 radians/sec) Figures V.5. The control system was able to follow commands with average accuracy. Furthermore it. A similar result was found when the pitch control system was tested (see Figures V.8). To demonstrate the ability of the system to trace the reference with large tilt values (>10°), a regression order was constructed to demonstrate the stability of the system within 90°. The quad plane reached 90° after 17 seconds and remained stationary until the experiment was stopped as shown in Fig. V.5.f. A similar result was found when the pitch control system was tested (see Figures V.8.f).



a The wind disturbance 2.2m/s

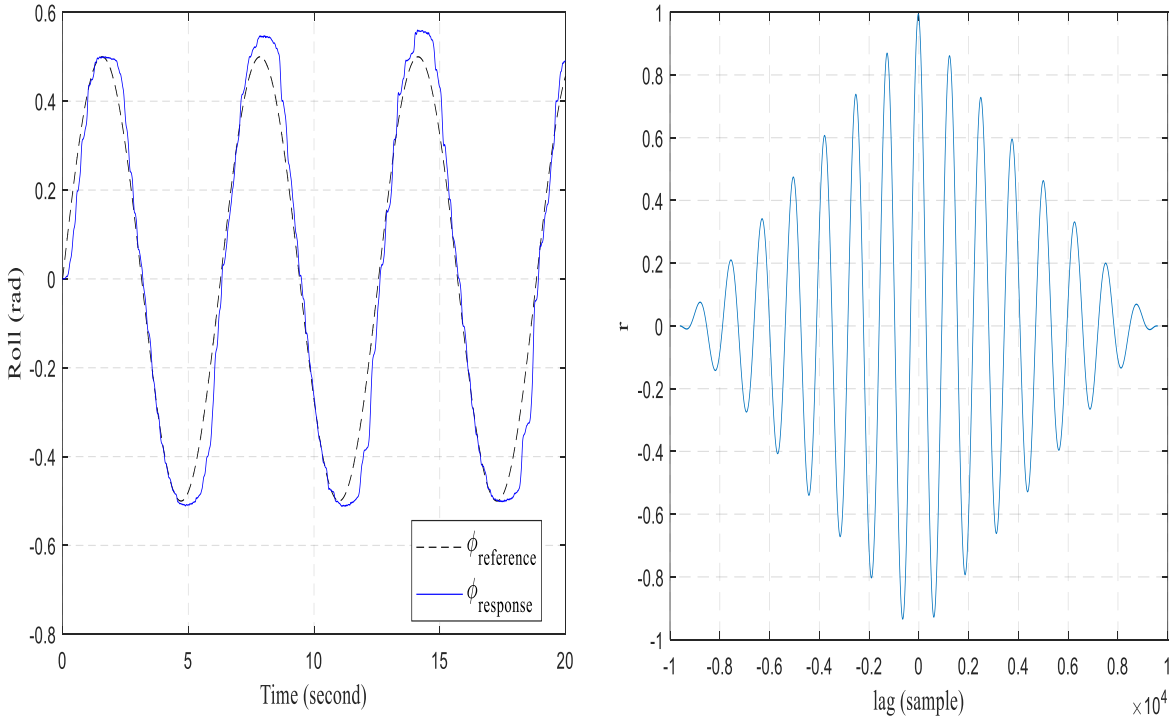


b The wind disturbance 3.2m/s

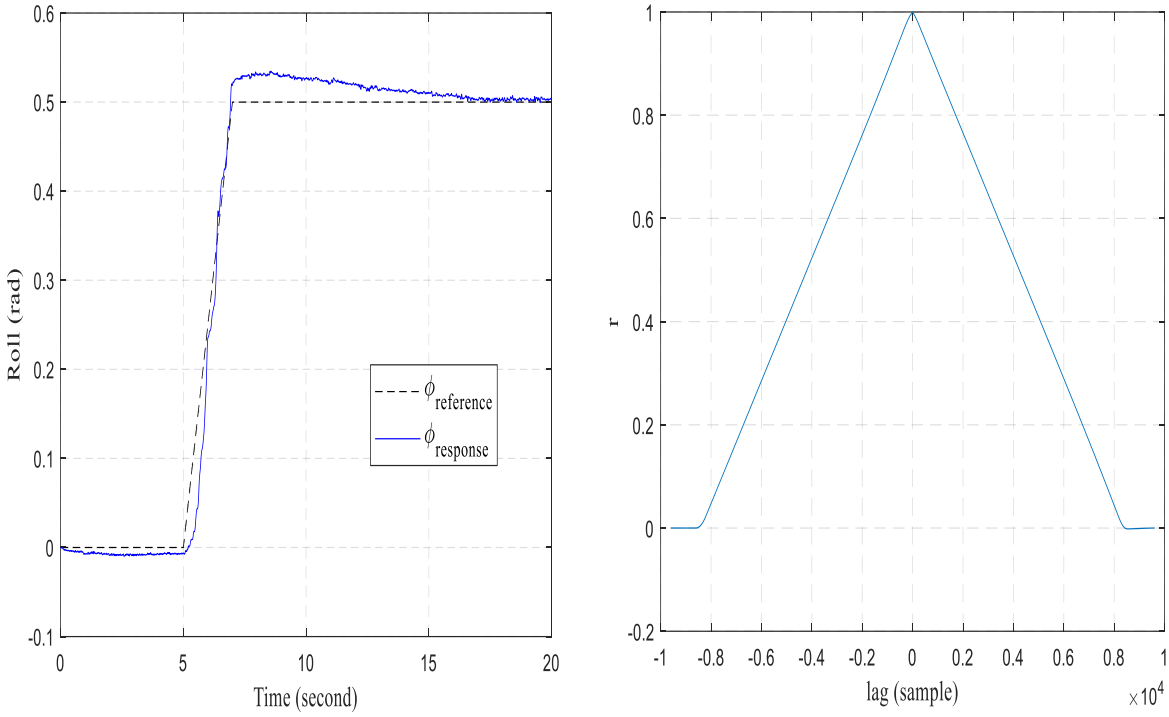


c The wind disturbance 4.2m/s

Figure V.6 The wind disturbance.



a Sinusoidal reference 1 rad/s and wind disturbance 4.2m/s

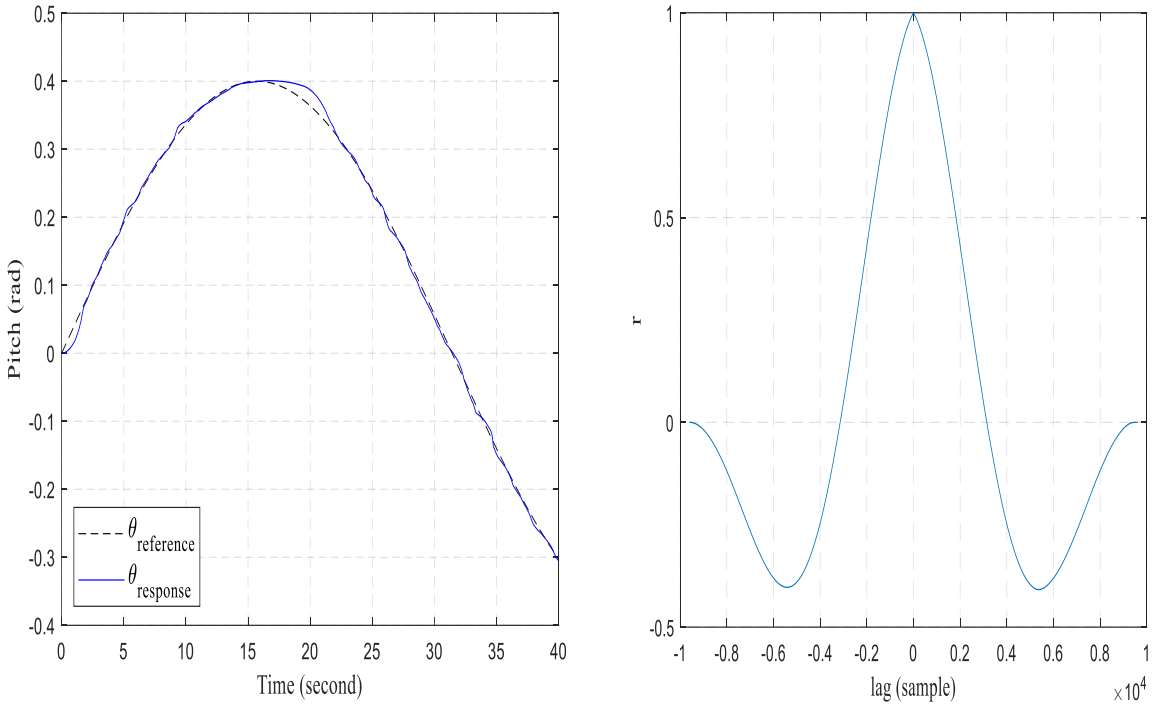


b ramp reference and wind disturbance 4.2m/s

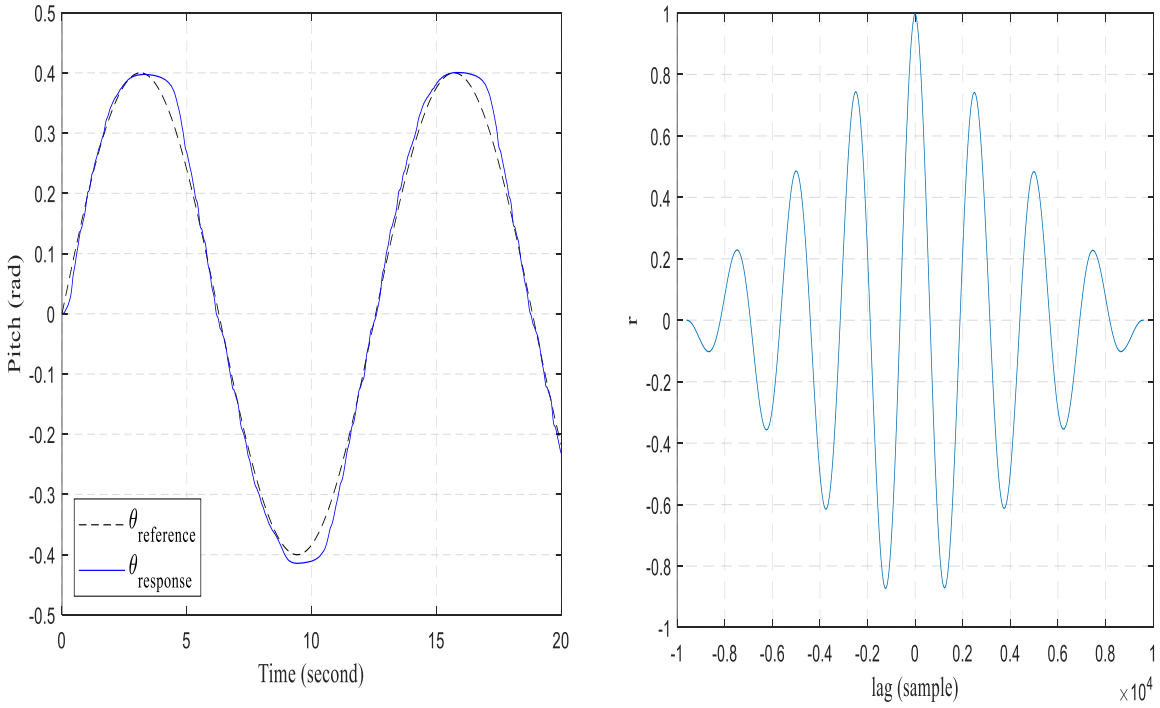
Figure V.7 The wind disturbance with tracking different reference command

To test the efficiency of the performance of the control system with mechanical turbulence with a concentrated wind source at variable speeds. The control system shows good performance with maximum tracking error of 0.086° at 4.2 m/s wind speed as shown in Fig. V.6. Moreover, the system shows good sinusoidal and slope order tracking with maximum wind turbulence (4.2 m/s) as shown in Fig. V.7. The results were also good for pitch control figures V.9 and figures V.10.

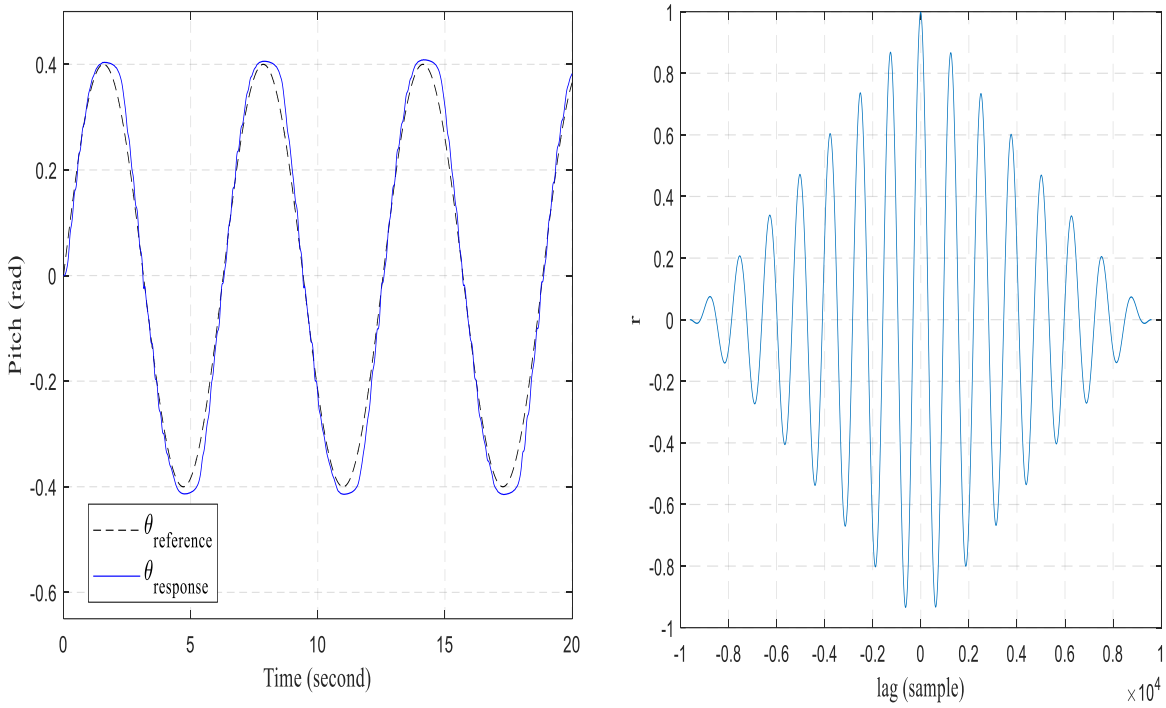
V.6.2 Result experiment for Pitch



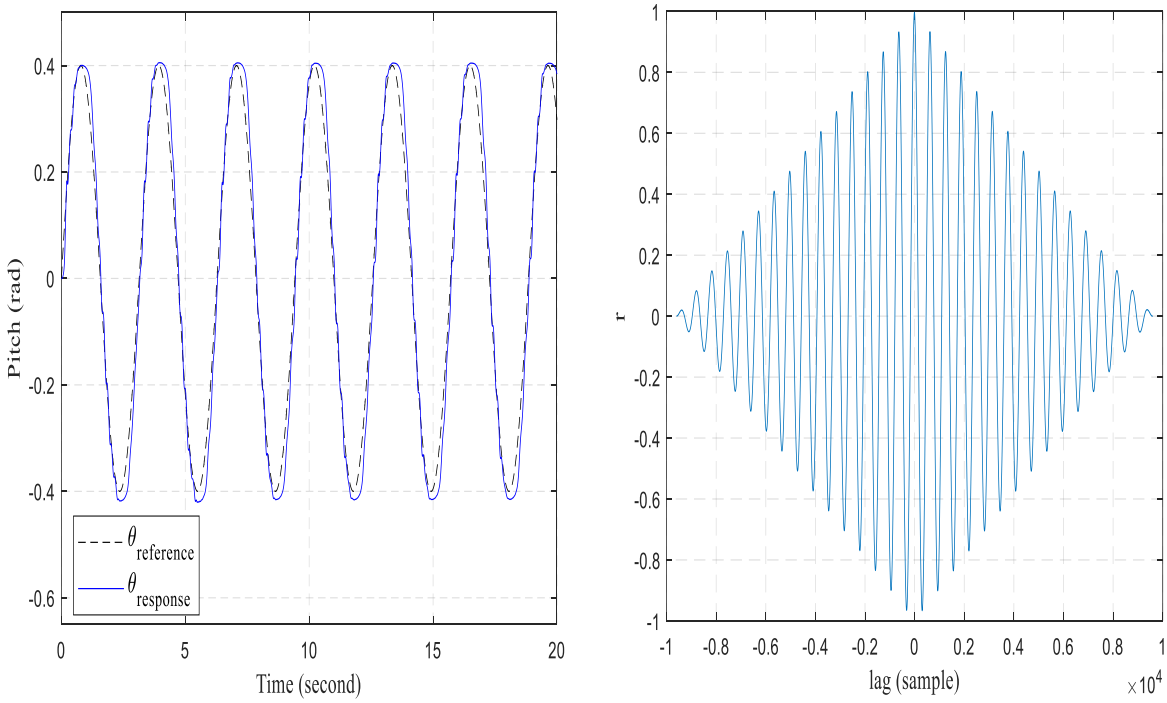
a sinusoidal reference 0.1 rad/s



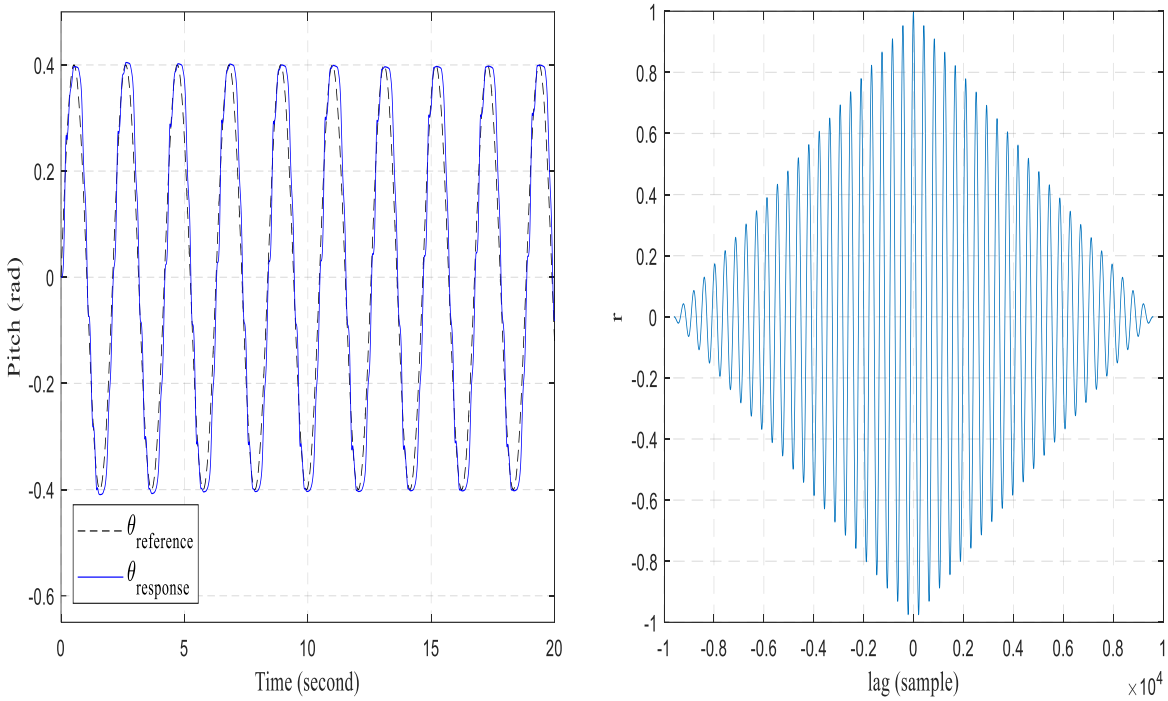
b sinusoidal reference 0.5 rad/s



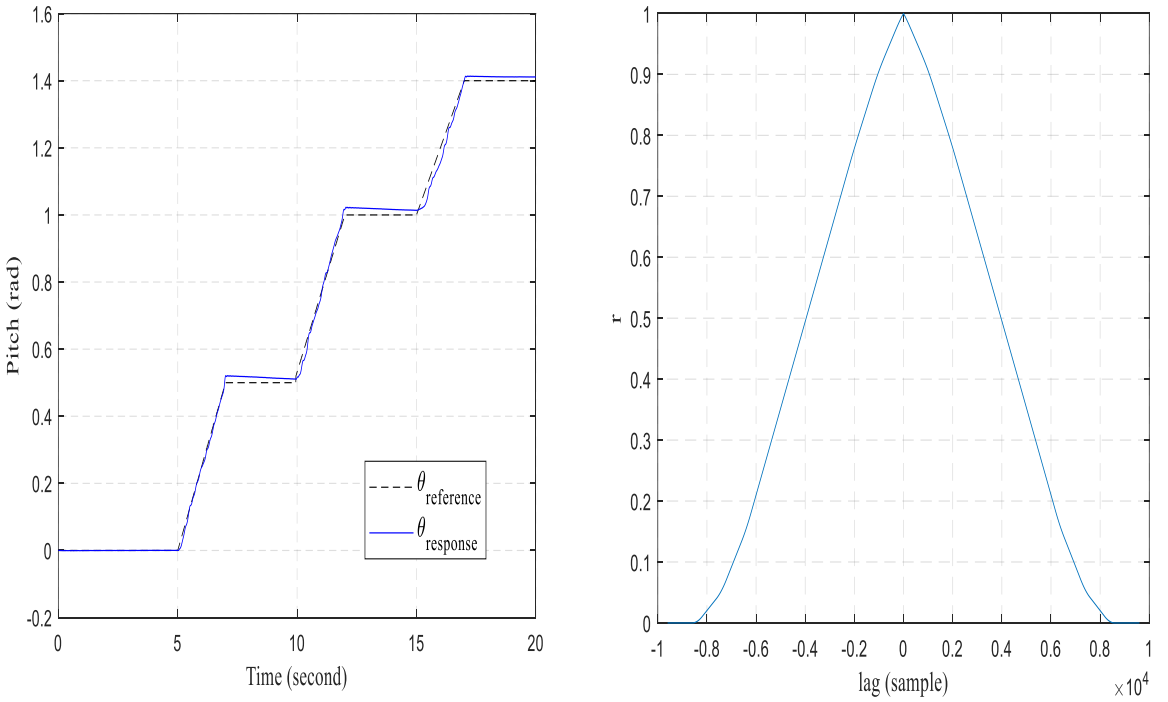
c sinusoidal reference 1 rad/s



d Sinusoidal reference 2 rad/s

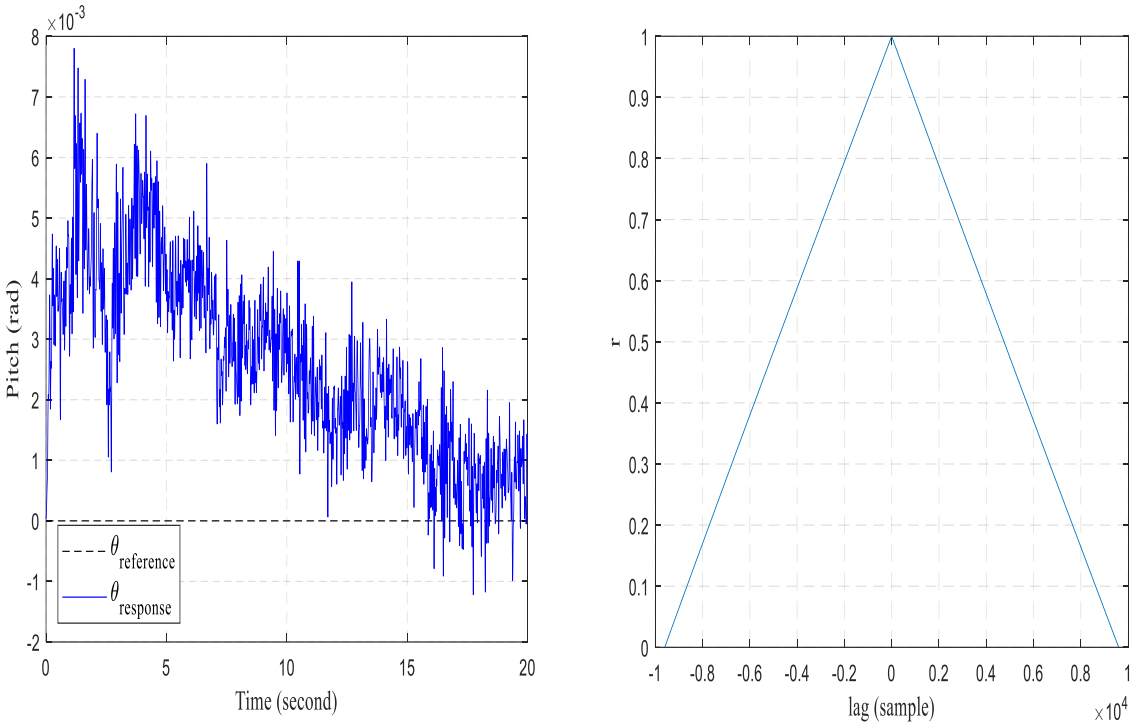


e sinusoidal reference 3 rad/s

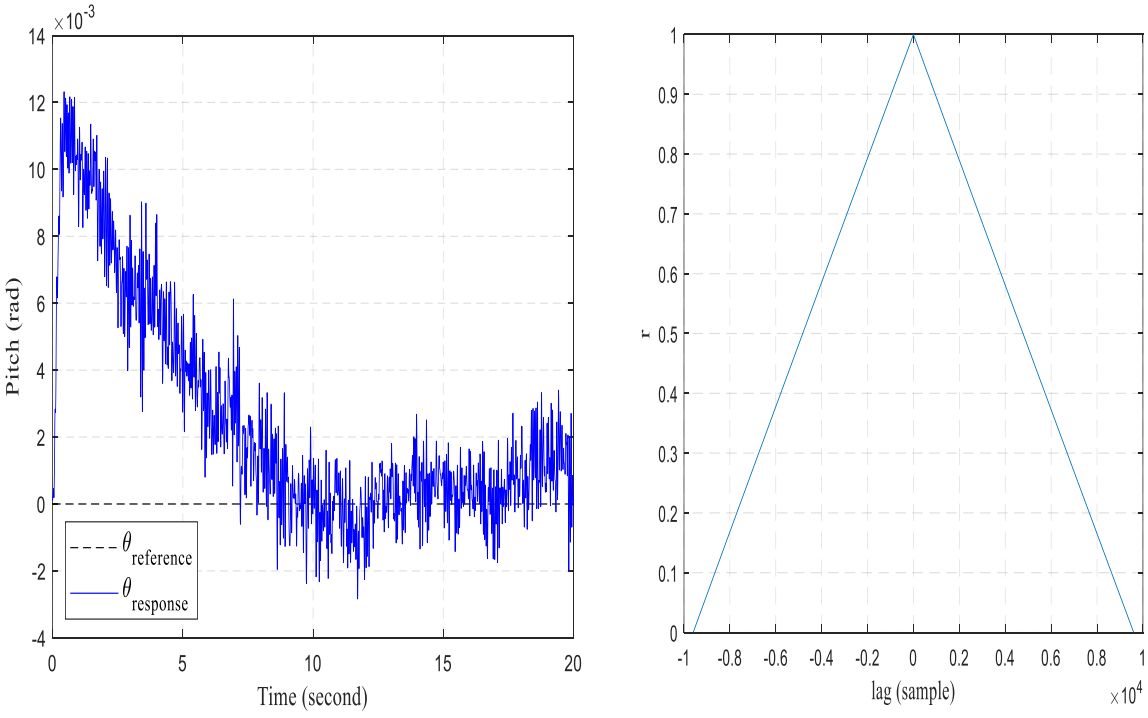


f Ramp reference.

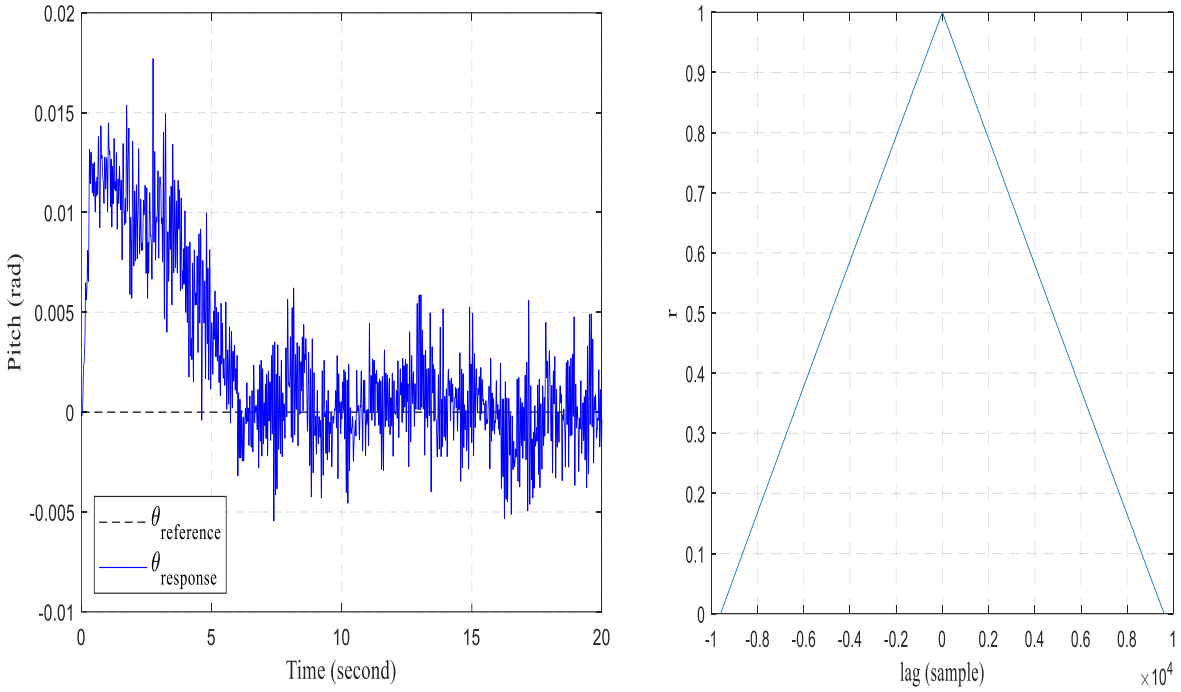
Figure V.8 track different frequencies command a sinusoidal reference and ramp reference.



a The wind disturbance 2.2m/s

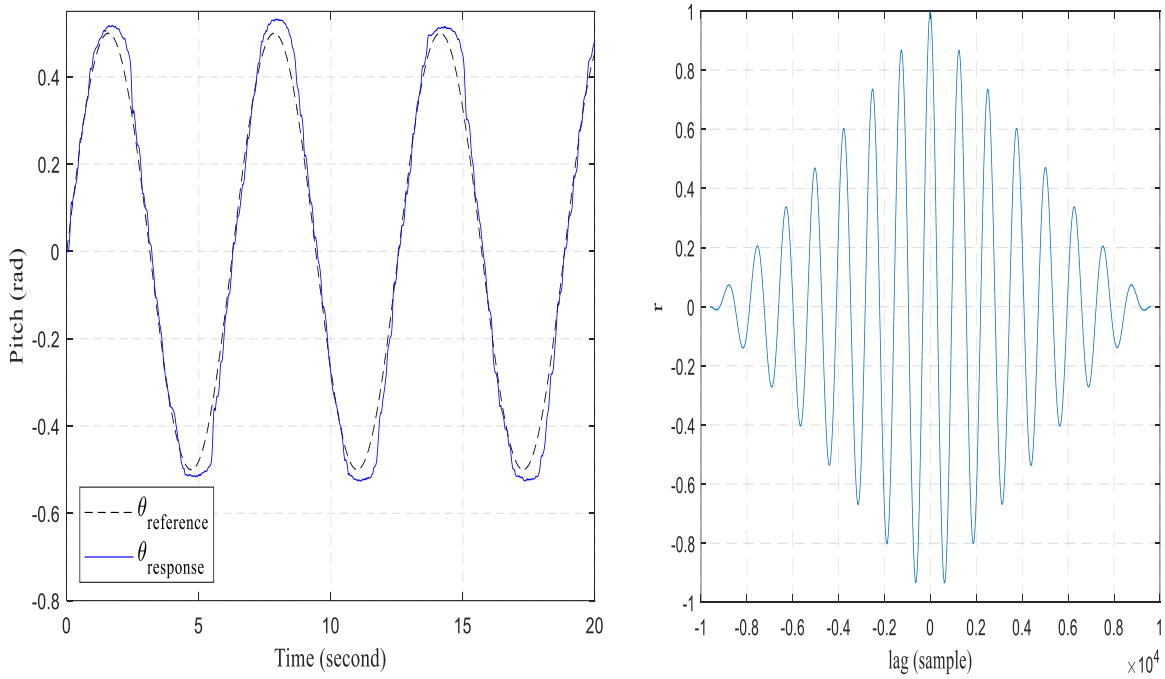


b The wind disturbance 3.2m/s

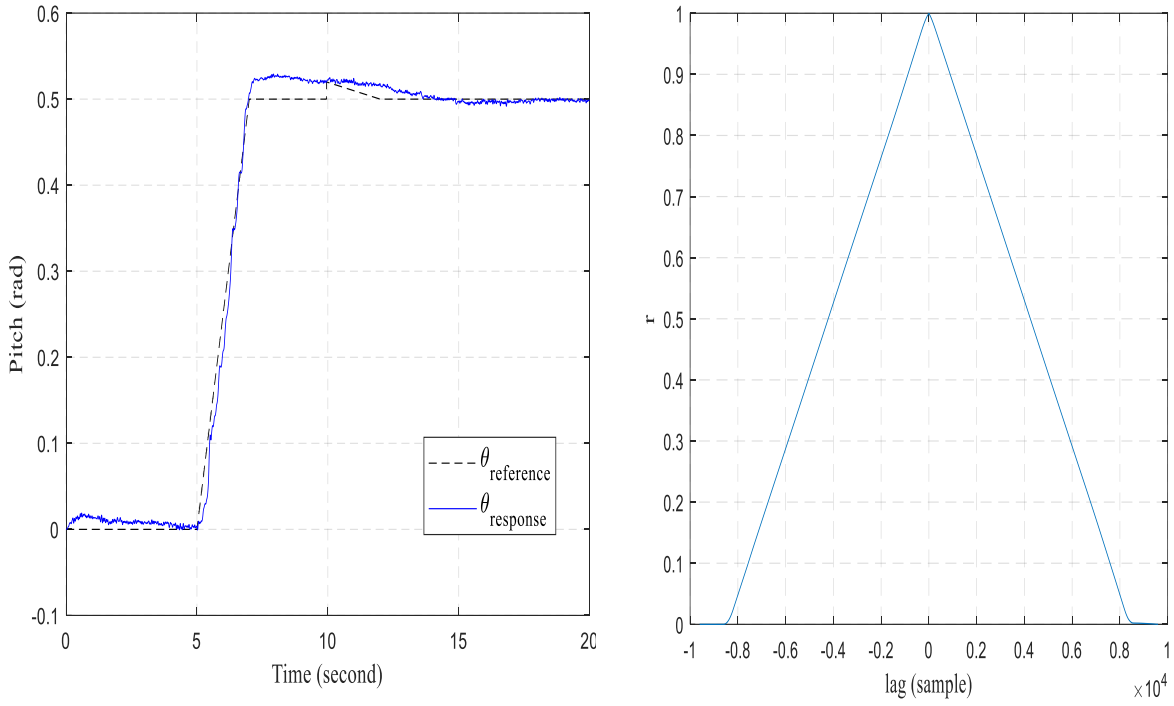


c The wind disturbance 4.2m/s

Figure V.9 The wind disturbance.

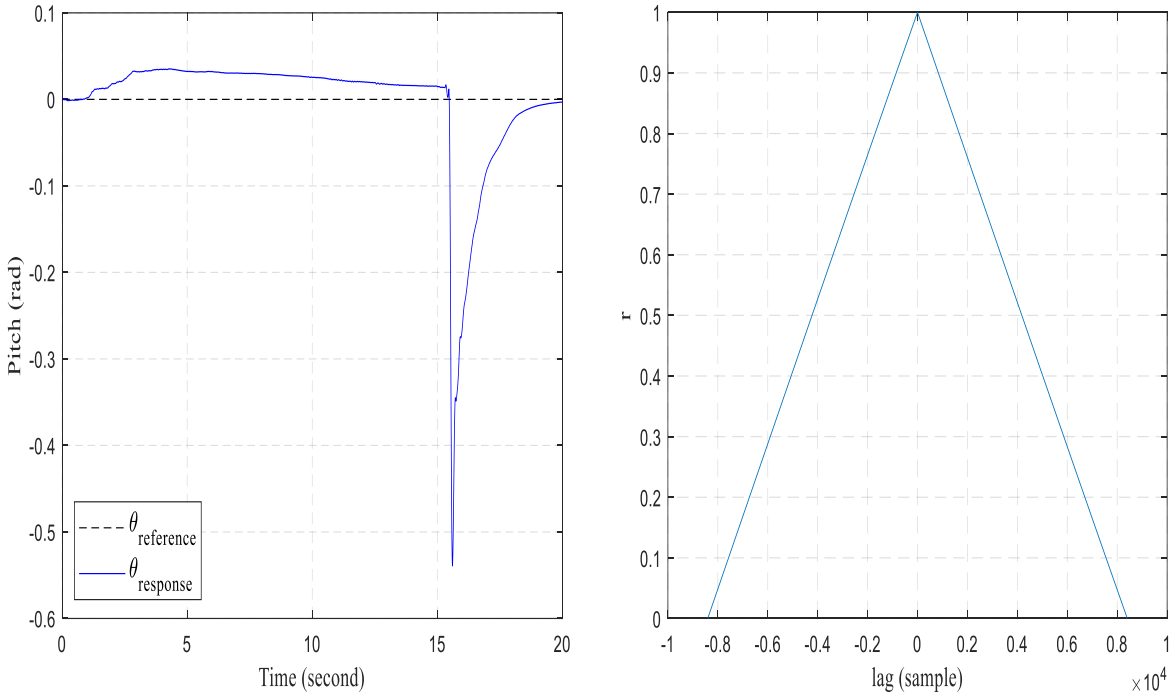


a sinusoidal reference 1 rad/s and wind disturbance 4.2m/s

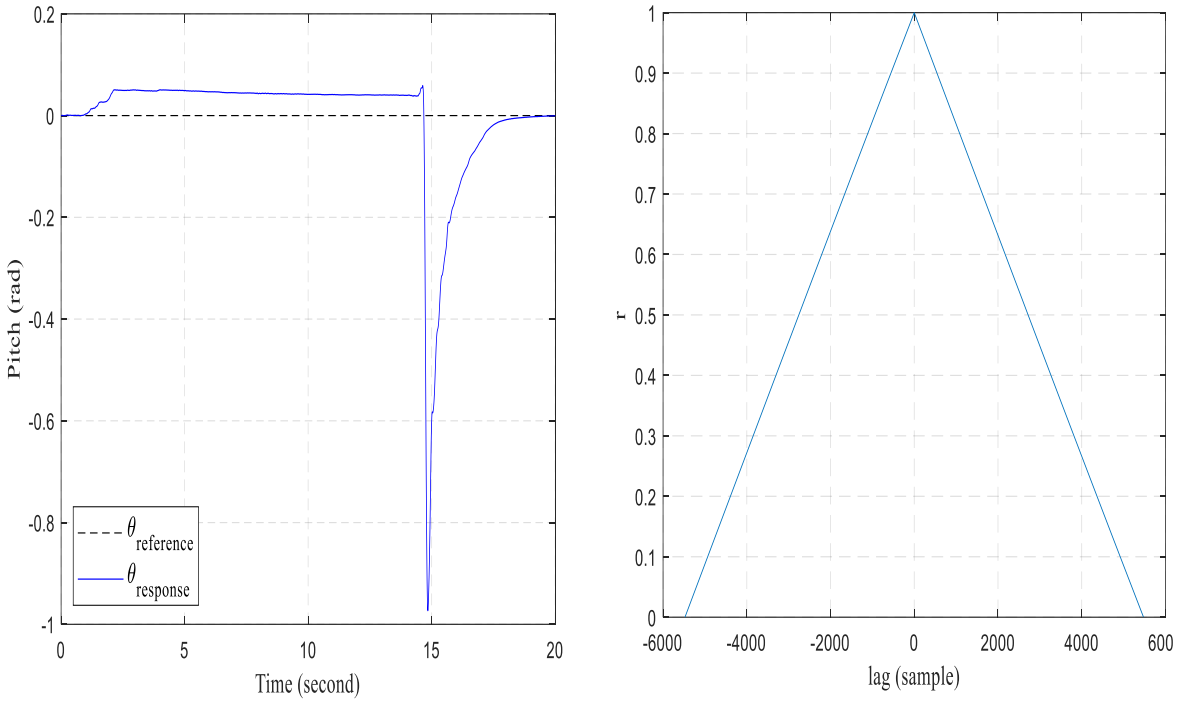


b ramp reference and wind disturbance 4.2m/s

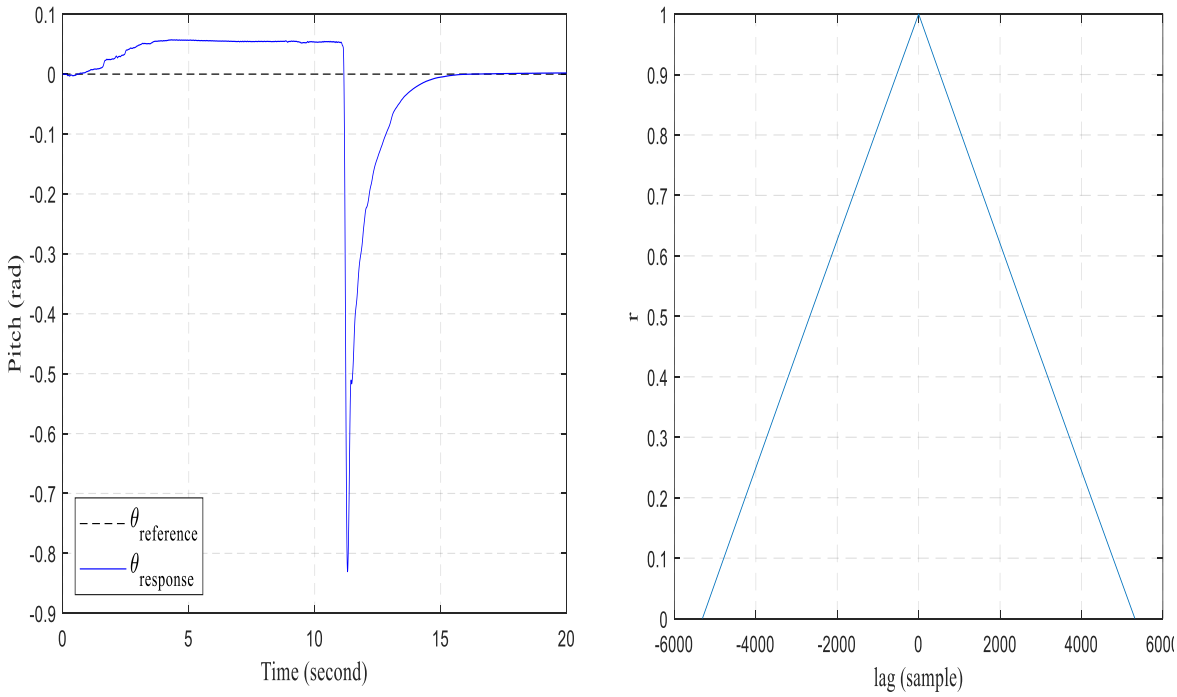
Figure V.10 The wind disturbance with tracking different reference command



a added the mechanical loads 60g



b added the mechanical loads 75g



c added the mechanical loads 95g

Figure V.11 added the mechanical loads and suddenly released

Like previous experiments, in this test we use bundle parashooting with zero angle command. The control was able to maintain system stability after the loads were released. For

maximum load, the system was perturbed by 49 degrees shift from position 0, and the quadrotor returned to position 0 after (3 s) see figure V.11.

Table V.1 Study of BGC with the cross-correlation overage r

			BGC			
			Roll		pitch	
Types	Specification		RMSE	r /lag	RMSE	r /lag
0.4sin (ωt)	ω (rad/s)	0.1	1.8676	0.9983/0	0.7191	0.9997/0
		0.5	4.6994	0.9941/ -26	2.1589	0.9988/0
		1	5.7364	0.9954/ -28	2.9930	0.9983/-6
		2	6.3225	0.9950/ -18	4.2792	0.9970/-8
		3	7.0975	0.9957/ -14	5.7878	0.9959/ -10
Ramp			1.6404	0.9999//0	1.0852	1/0
Mechanical load/unload	Load (g)	60			1.8145	1/0
		75			2.0424	1/0
		95			1.5606	1/0
Wind disturbance	Speed (m/s)	2.2	0.5032	1/0	2.1513	0.9999/0
		3.2	0.1800	1/0	1.1885	1/0
		4.2	0.3320	1/0	1.7019	1/0
	.5sin (t)		7.1108	0.9943/ -21	4.3464	0.9978/-10
	0.5(rad)		1.4760	0.9995/ 0	0.9839	0.9998/0

Table V.1 illustrates the statistical analysis of backstepping control system, and system behaviors against mechanical perturbation.

In experiment one the RMSE increase proportionally with the target frequency increase for both degree of freedoms. The maximum error is 7.097 and 5.787 percent on roll and pitch, respectively. The control system was able to follow all targets at five different frequencies with high correlation with minimum correlation of 0.994 and maximum lag of 28 samples.

Backstepping Geometric control respond to mechanical disturbance is reported in experiment three where a mechanical loading and unloading on the system in which the system must keep tilt angle at zero. The system was able to respond robustly see figures V.11. In this experiment, the behavior of the control system can be distinguished from the HDP control. The HDP react aggressively to unloading which can lead to unstable behavior while all the coupling components are affecting the system in real system operation. On the other hand, the backstepping geometric control system react smoothly to system unloading.

Final experiment is designed to study system behavior under constant wind where three flow speed were selected and the control system stabilized the system under all types of targets

as shown in table V.1 see figures V.9 and V .10. The system shows robust respond with very high correlation.

V.7 Conclusion

The results obtained were acceptable in terms of accuracy compared with the impedance control. But it did not reach the desired goal compared to the high dynamic performance. This latter was highly efficient, which we suspect is due to the quad-mount design that separates each corner individually. Therefore, in the future, we will use an aircraft carrier with three degrees of freedom for efficiency study backstepping geometric control.

General Conclusion

Autonomous robotic systems have become an interesting concept, which is developing in an accelerating pace in recent years. The autonomous systems research moved from being exclusive to a few institutions, to become a global trend which is being constantly enhancing by the research all over the world.

One of the recently developed area in autonomous robots is vertical tack-off and landing UAV. A common task for UAV is cooperative or surveillance missions to follow ground-located targets. In many cases, such targets can be dynamic and perform unpredictable movements, which must be considered by the UAV navigation algorithms in order to accurately track and maneuver.

The aim of this work is to contribute with an effective and suitable control techniques for controlling vertical tack-off and landing UAV (i.e., quadrotor). A successful control system would be able to regulate the motion (in 6-DoF) of the quadrotor for both small and large angles of rotation. Moreover, ensuring its stability and robust performance against all types of perturbations.

In first described the dynamic behavior of the robot, then studied the geometric mechanics of the rigid body in the Lie group in order to obtain a mathematical model in SO (3) for the command attitude and SE (3) for high-level control (i.e., translational motion). Second, a geometric backstepping control system was developed to ensure the stability of the system. The results obtained both by numerical simulation and practical implementation. The results show the superiority of the suggested control system compared with conventional control system.

During the development of this thesis, several points that can be improved have been detected, hence the perspectives and future works are numerous

- Apply the techniques proposed in this work to other UAV configurations, whether for fixed or rotary wings;
- To develop an observer taking into account the variety of rotation structures. Using models in SO (3) for attitude and SE (3) for position;
- The use of these commands for carrying out very specific missions, such as: carrying a load (where the System becomes quadrotor with suspended pendulum); cooperation between multi-agents of the quadrotor.

- The consideration of communication which is very important for experimental tests in real time, where increasing bandwidth and range is a challenging task.
- To properly estimate the source of energy to use for successful maneuvers.

References

References

[1] Putro NAS, Dharmawan A, Priyambodo TK. Quadrotor Control System with Hand Movement Sign as an Alternative Remote Control. International Journal of Robotics and utomation (IJRA). 2017;6(2):131-40.

[2] Yu J, Yan C, Huang M, editors. Research of consistency problem for quadrotor UAV system with leader-follower. 2019 Chinese Automation Congress (CAC); 2019: IEEE.

[3] Mahony R, Hamel TJEJoC. Adaptive compensation of aerodynamic effects during takeoff and landing manoeuvres for a scale model autonomous helicopter. 2001;7(1):43-57.

[4] Lasmadi L, Cahyadi AI, Hidayat R, Herdjunanto S. Inertial navigation for quadrotor using kalman filter with drift compensation. International Journal of Electrical and Computer Engineering. 2017;7(5):2596.

[5] Al Al AYD, Bachtiar A, Harinita D. An angle speed and thrust relationship of the quadcopter rotor. Indonesian Journal of Electrical Engineering and Computer Science. 2019;13(2):469-74.

[6] Jinyong L, Jun Z, Wenjie D, Xiaoying G, Chuntang Q, editors. Flying control system architecture of informationization for pervasive computing. 2010 First International Conference on Pervasive Computing, Signal Processing and Applications; 2010: IEEE.

[7] Goodarzi FA, Lee T, editors. Extended kalman filter on se (3) for geometric control of a quadrotor uav. 2016 International Conference on Unmanned Aircraft Systems (ICUAS); 2016: IEEE.

[8] Tan T, Yin J, Zhou H, editors. High Precision Attitude Control of Space Maneuvering Platform Based on Model Predictive and Inversive Method. 2019 IEEE International Conference on Unmanned Systems (ICUS); 2019: IEEE.

[9] Goodarzi F, Lee D, Lee T, editors. Geometric nonlinear PID control of a quadrotor UAV on SE (3). 2013 European control conference (ECC); 2013: IEEE.

[10] Zhang S, Chen C, editors. Backstepping based nonlinear integral sliding mode control for quadrotors under external disturbances. 2019 Chinese Control Conference (CCC); 2019: IEEE.

[11] Zhou L, Zhang B, editors. Quadrotor UAV Flight Control Using Backstepping Adaptive Controller. 2020 IEEE 6th International Conference on Control Science and Systems Engineering (ICCSSE); 2020: IEEE.

[12] Voronkov OY, editor Synergetic Synthesis of the Hierarchical Control System of the “Flying Platform”. 2019 III International Conference on Control in Technical Systems (CTS): IEEE.

[13] Zou Y, Yin Y, Song J, editors. Flight control of a flying-wing UAV based on active disturbance rejection control. 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE); 2017: IEEE.

[14] Hu M, Song J, Yang M, editors. Sliding mode control and control allocation for electromagnetic formation flying. 2016 35th Chinese Control Conference (CCC); 2016: IEEE.

[15] Ginting AH, Wahyunggoro O, Cahyadi AI. Attitude control of quadrotor using PD plus feedforward controller on SO (3). International Journal of Electrical and Computer Engineering (IJECE). 2018;8(1):566-75.

[16] Ji X, Li J, Niu Y, editors. Optimal attitude control of a rigid body on special orthogonal group using LGVI and FD-GMRES. 2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR); 2015: IEEE.
[17] Chen T-y, editor Quadrotor Attitude Control Using Special Orthogonal Matrix. 2020 International Automatic Control Conference (CACS); 2020: IEEE.
[18] Chenlu W, Zengqiang C, Qinglin S, Qing Z, editors. Design of PID and ADRC based quadrotor helicopter control system. 2016 Chinese Control and Decision Conference (CCDC); 2016: IEEE.
[19] Jiajin L, Rui L, Yingjing S, Jianxiao Z, editors. Design of attitude controller using explicit model predictive control for an unmanned quadrotor helicopter. 2017 Chinese Automation Congress (CAC); 2017: IEEE.
[20] Jiao Q, Liu J, Zhang Y, Lian W, editors. Analysis and design the controller for quadrotors based on PID control method. 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC); 2018: IEEE.
[21] Lee T, Leok M, McClamroch NH, editors. Geometric tracking control of a quadrotor UAV on SE (3). 49th IEEE conference on decision and control (CDC); 2010: IEEE.
[22] Lee T, editor Geometric tracking control of the attitude dynamics of a rigid body on SO (3). Proceedings of the 2011 American Control Conference; 2011: IEEE.
[23] Lee T. Robust adaptive geometric tracking controls on SO (3) with an application to the attitude dynamics of a quadrotor UAV. arXiv preprint arXiv:11086031. 2011.
[24] Mokhtari A, Benallegue A, editors. Dynamic feedback controller of Euler angles and wind parameters estimation for a quadrotor unmanned aerial vehicle. IEEE International Conference on Robotics and Automation, 2004 Proceedings ICRA'04 2004; 2004: IEEE.
[25] Madani T, Benallegue A. Adaptive control via backstepping technique and neural networks of a quadrotor helicopter. IFAC Proceedings Volumes. 2008;41(2):6513-8.
[26] Yin C, Shi X, Hu B, Cheng Y, Huang X, Zhang B, et al., editors. Trajectory tracking optimal control with backstepping method for quadrotor helicopter. 2017 Chinese Automation Congress (CAC); 2017: IEEE.
[27] Castillo P, Lozano R, Dzul A, editors. Stabilization of a mini-rotorcraft having four rotors. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat No 04CH37566); 2004: IEEE.
[28] Shulong Z, Honglei A, Daibing Z, Lincheng S, editors. A new feedback linearization lqr control for attitude of quadrotor. 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV); 2014: IEEE.
[29] Liu H, Wang X, Zhong Y. Quaternion-based robust attitude control for uncertain robotic quadrotors. IEEE Transactions on Industrial Informatics. 2015;11(2):406-15.
[30] Mayhew CG, Sanfelice RG, Teel AR, editors. Robust global asymptotic attitude stabilization of a rigid body by quaternion-based hybrid feedback. Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference; 2009: IEEE.
[31] Carino J, Abaunza H, Castillo P, editors. Quadrotor quaternion control. 2015 International Conference on Unmanned Aircraft Systems (ICUAS); 2015: IEEE.
[32] Lee T, Leok M, McClamroch NH. Control of complex maneuvers for a quadrotor UAV using geometric methods on SE (3). arXiv preprint arXiv:10032005. 2010.
[33] Lee T, editor Geometric controls for a tethered quadrotor UAV. 2015 54th IEEE conference on decision and control (CDC); 2015: IEEE.

[34] Gonzalez HnA. Robust tracking of dynamic targets with aerial vehicles using quaternion-based techniques 2019.
[35] Benaddy A, Bouzi M, Labbadi M, editors. Comparison of the different control strategies for Quadrotor Unmanned Aerial Vehicle. 2020 International Conference on Intelligent Systems and Computer Vision (ISCV); 2020: IEEE.
[36] Barcelos D, Kolaei A, Bramesfeld GJJoA. Aerodynamic Interactions of Quadrotor Configurations. 2020;57(6):1074-90.
[37] Pounds P, Mahony R, Corke P, editors. Modelling and control of a quad-rotor robot. Proceedings Australasian Conference on Robotics and Automation 2006; 2006: Australian Robotics and Automation Association Inc.
[38] Baldini A, Felicetti R, Freddi A, Monteriù A, Tempesta M, editors. Estimation of actuator faults in quadrotor vehicles: from theory to validation with experimental flight data. 2020 International Conference on Unmanned Aircraft Systems (ICUAS); 2020: IEEE.
[39] Idrissi M, Annaz FJIJoME, Research R. Dynamic Modelling and Analysis of a Quadrotor Based on Selected Physical Parameters. 2020;9(6).
[40] Marsden JE, Ratiu TS. Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems: Springer Science & Business Media; 2013.
[41] Carrillo LRG, López AED, Lozano R, Pégard C. Quad rotorcraft control: vision-based hovering and navigation: Springer Science & Business Media; 2012.
[42] Lozano R, Castillo P, Salazar S, Lara DJPJ, Irisa, France. Stabilisation de véhicules aériens à décollage vertical: théorie et application. 2005:67-73.
[43] Bouabdallah S, Murrieri P, Siegwart R, editors. Design and control of an indoor micro quadrotor. IEEE International Conference on Robotics and Automation, 2004 Proceedings ICRA'04 2004; 2004: IEEE.
[44] Bouabdallah S, Siegwart R, editors. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. Proceedings of the 2005 IEEE international conference on robotics and automation; 2005: IEEE.
[45] Bouabdallah S, Siegwart R, editors. Towards intelligent miniature flying robots. Field and service robotics; 2006: Springer.
[46] Campa R, Camarillo K, Ceccarelli MJRm, M. Ceccarelli , In-Teh. Unit quaternions: A mathematical tool for modeling, path planning and control of robot manipulators. 2008:21-48.
[47] Kuipers JB. Quaternions and rotation sequences: a primer with applications to orbits, aerospace, and virtual reality: Princeton university press; 1999.
[48] Siminovitch DJJCiMRAEJ. Rotations in NMR: Part I. Euler–Rodrigues parameters and quaternions. 1997;9(3):149-71.
[49] Spring KWJM, theory m. Euler parameters and the use of quaternion algebra in the manipulation of finite rotations: a review. 1986;21(5):365-73.
[50] Galilei G. Dialogue concerning the two chief world systems, Ptolemaic and Copernican: Modern Library Classics; 2001.
[51] Ogunseye HA, Mondal H, Sibanda P, Mambili-Mamboundou H. Lie group analysis of a Powell–Eyring nanofluid flow over a stretching surface with variable properties. SN Applied Sciences. 2020;2(1):1-12.
[52] Barbaresco F. Lie group statistics and Lie group machine learning based on Souriau Lie groups thermodynamics & Koszul-Souriau-Fisher metric: new entropy definition as

generalized casimir invariant function in coadjoint representation. Entropy. 2020;22(6):642.
[53] Hashim HA. Systematic convergence of nonlinear stochastic estimators on the special orthogonal group SO (3). International Journal of Robust and Nonlinear Control. 2020;30(10):3848-70.
[54] Mei Y, Liao Y, Gong K, Luo D. Adaptive Fuzzy Modified Fixed-Time Fault-Tolerant Control on SE (3) for Coupled Spacecraft. Mathematical Problems in Engineering. 2021;2021.
[55] Bull K, Desaules J-Y, Papić Z. Quantum scars as embeddings of weakly broken Lie algebra representations. Physical Review B. 2020;101(16):165139.
[56] Lee T. Computational Geometric Mechanics and Control of Rigid Bodies 2008.
[57] Chaturvedi NA, Lee T, Leok M, McClamroch NHJJoNS. Nonlinear dynamics of the 3D pendulum. 2011;21(1):3-32.
[58] Dahlgren J. Robust nonlinear control design for a missile using backstepping. Institutionen för systemteknik; 2002.
[59] Sakunthala G, Narayanan A, Dasgupta S, editors. A novel method to introduce built in robustness in backstepping. AIP Conference Proceedings; 2019: AIP Publishing LLC.
[60] Härkegård O, Glad STJIPV. Flight control design using backstepping. 2001;34(6):283-8.
[61] CHEBBI A. Commande Backstepping d'une machine asynchrone sans capteur de vitesse: Université de Batna 2; 2011.
[62] Zhou J, Wen C. Adaptive backstepping control of uncertain systems: Nonsmooth nonlinearities, interactions or time-variations: Springer; 2008.
[63] Garcia O, Rojo-Rodriguez EG, Sanchez A, Saucedo D, Munoz-Vazquez A-JJR. Robust Geometric Navigation of a Quadrotor UAV on SE (3). 2020;38(6):1019-40.
[64] Lee T, Chang DE, Eun YJJoDS, Measurement,, Control. Semiglobal nonmemoryless attitude controls on the special orthogonal group. 2019;141(2).
[65] Akhtar A, Saleem S, Waslander SL, editors. Feedback Linearizing Controllers on SO (3) using a Global Parametrization. 2020 American Control Conference (ACC); 2020: IEEE.
[66] Labbadi M, Cherkaoui MJIt. Robust adaptive nonsingular fast terminal sliding-mode tracking control for an uncertain quadrotor UAV subjected to disturbances. 2020;99:290-304.
[67] Glida HE, Abdou L, Chelihi A, Sentouh C, Hasseni S-E-IJND. Optimal model-free backstepping control for a quadrotor helicopter. 2020:1-20.
[68] Guendouzi A, Hamerlain M, Saadia NJIJoR. A Robust Adaptive Nonlinear Control Design via Geometric Approach for a Quadrotor. 2020:1-11.
[69] Gamagedara K, Lee T, Chang DE, editors. Attitude observer on SO (3) with time-varying reference directions. 2019 18th European Control Conference (ECC); 2019: IEEE.
[70] Ghasemi K, Ghaisari J, Abdollahi FJIJoR, Control N. Robust formation control of multiagent systems on the Lie group SE (3). 2020;30(3):966-98.
[71] Bisheban M, Lee TJIToCST. Geometric Adaptive Control With Neural Networks for a Quadrotor in Wind Fields. 2020.
[72] Shi X-N, Zhou D, Zhou Z-G, Li RJIJoSS. Continuous adaptive-gain finite-time control for rigid body attitude dynamics on SO (3). 2019;50(1):178-90.
[73] Bodó Z, Lantos BJPPEE, Science C. Integrating Backstepping Control of Outdoor Quadrotor UAVs. 2019;63(2):122-32.

[74] Gamagedara K, Bisheban M, Kaufman E, Lee T, editors. Geometric controls of a quadrotor uav with decoupled yaw control. 2019 American Control Conference (ACC); 2019: IEEE.
[75] Kaplan MR, Eraslan A, Beke A, Kumbasar T, editors. Altitude and position control of parrot mambo minidrone with PID and fuzzy PID controllers. 2019 11th International Conference on Electrical and Electronics Engineering (ELECO); 2019: IEEE.
[76] Kun L, Inoue Y, Shibata K, Enguo C. Ambulatory estimation of knee-joint kinematics in anatomical coordinate system using accelerometers and magnetometers. IEEE Transactions on Biomedical Engineering. 2010;58(2):435-42.
[77] Wu J, Zhou Z, Fourati H, Cheng Y. A super fast attitude determination algorithm for consumer-level accelerometer and magnetometer. IEEE Transactions on Consumer Electronics. 2018;64(3):375-81.
[78] Roveda L, Maskani J, Franceschi P, Abdi A, Braghin F, Tosatti LM, et al. Model-based reinforcement learning variable impedance control for human-robot collaboration. Journal of Intelligent & Robotic Systems. 2020:1-17.
[79] Tiseo C, Merkt W, Babarahmati KK, Wolfslag W, Vijayakumar S, Mistry M, editors. Bio-mimetic adaptive force/position control using fractal impedance. 2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob); 2020: IEEE.
[80] Rashad R, Bicego D, Jiao R, Sanchez-Escalonilla S, Stramigioli S, editors. Towards Vision-Based Impedance Control for the Contact Inspection of Unknown Generically-Shaped Surfaces with a Fully-Actuated UAV. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2020: IEEE.
[81] Reinikka T, Roinila T, Sun J, editors. Measurement Device for Inverter Output Impedance Considering the Coupling Over Frequency. 2020 IEEE 21st Workshop on Control and Modeling for Power Electronics (COMPEL); 2020: IEEE.
[82] Zhou Z. A Front-End Amplifier With Current Compensation Feedback Input Impedance Booster for Neural Signal Applications. IEEE Access. 2020;8:178055-62.
[83] Mujica M, Benoussaad M, Fourquet J-Y, editors. Evaluation of Human-Robot Object Co-manipulation Under Robot Impedance Control. 2020 IEEE International Conference on Robotics and Automation (ICRA); 2020: IEEE.
[84] Abu-Dakka FJ, Saveriano M. Variable impedance control and learning--A review. arXiv preprint arXiv:201006246. 2020.
[85] Šumarac J, Rodić A, Jovanović K. Impedance Control of an Industrial Manipulator.
[86] Izadbakhsh A, Khorashadizadeh S, Ghandali S. Robust adaptive impedance control of robot manipulators using Szász–Mirakyan operator as universal approximator. ISA transactions. 2020;106:1-11.
[87] Zhao H, Wu S, Wen Y, Liu W, Wu X. Modeling and flight experiments for swarms of high dynamic UAVs: A stochastic configuration control system with multiplicative noises. Sensors. 2019;19(15):3278.
[88] Chang X, Rong L, Chen K, Fu W. LSTM-Based Output-Constrained Adaptive Fault-Tolerant Control for Fixed-Wing UAV with High Dynamic Disturbances and Actuator Faults. Mathematical Problems in Engineering. 2021;2021.
[89] Xia Y, Wan Y, Luo X, Wang H, Gong N, Cao J, et al. Chatter suppression in large overhang face milling using a toolholder with high dynamic performance. The International Journal of Advanced Manufacturing Technology. 2020;108:1713-24.

[90] Bouchaib A, Taleb R, Massoum A, Mekhilef S "Geometric control of quadrotor UAVs using integral backstepping" Indonesian Journal of Electrical Engineering and Computer Science Vol. 22, No. 1, April 2021, pp. 53~61 ISSN: 2502-4752, DOI: 10.11591/ijeecs.v22.i1.pp53-61

APPENDIX

APPENDIX A

A.1 Properties of the hat map $\circ_X$

operator $\circ_X : \mathbb{R} \to SO(3)$ is isomorphism between $\mathbb{R}^3$ the Lie algebra $SO(3)$ which is the set of 3×3 antisymmetric matrices given by:

$$u_X = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ u_2 & u_1 & 0 \end{bmatrix}$$

For $\Omega = [\Omega_1 \quad \Omega_2 \quad \Omega_3]^T \in \mathbb{R}^3$ The Lie bracket on $so(3)$ corresponds to the cross product on $\mathbb{R}^3$.

properties of the hat map are summarized

$$u_X v = u \times v = -u_X \times v = -v_X \times u$$

$$u_X^T u_X = (u^T u)I - uu^T$$

$$u_X v_X u_X = -(v^T u)u_X$$

$$u^T v = \frac{1}{2} tr[u_X v_X]$$

$$(u \times v)_X = u_X v_X - v_X u_{X=} vu^T - uv^T$$

$$tr[u_X A] = \frac{1}{2} tr[u_X (A - A^T)]$$

$$(Au)_X = u_X \left(\frac{1}{2} tr[A]I - A\right) + \left(\frac{1}{2} tr[A]I - A\right)^T u_X$$

For whoever $u, v \in \mathbb{R}^3$ and $A \in \mathbb{R}^{3\times3}$.

APPENDIX B

With the matrix $G \in \mathbb{R}^3$ give by $G = diag[g_1, g_2, g_3]$ are constant positive with $g_i \in R$ where $i = 1,2,3$

We have in the formula of Rodrigues $Q = R_d^T R = \exp(x_\times)\ Q \in SO(3)$ and $x \in \mathbb{R}^3$ we found

$$\rho = \frac{1 - cos\|x\|}{2\|x\|^2} \sum_{(i,j,k)\in\varsigma} (g_i + g_j) x_k^2$$

$$\|e_R\|^2 - \frac{(1-cos\|x\|)^2}{4\|x\|^4}\sum_{(i,j,k)\in\varsigma}(g_i-g_j)^2x_i^2x_j^2 + \frac{cos^2\|x\|}{4\|x\|^2}\sum_{(i,j,k)\in\varsigma}(g_i+g_j)^2x_k^2$$

With $\varsigma = \{(1,2,3),(2,3,1),(3,1,2)\}$

the upper value of :

$$\frac{\|e_R\|^2}{\rho} \leq \frac{1}{2h_1}(1-cos\|x\|)h_2 + \frac{1}{2h_1}(1+cos\|x\|)h_3 \leq \frac{h_2+h_3}{h_1}$$

$$(\frac{h_2+h_3}{h_1})\ \|e_R\|^2 \leq\ \rho$$

Where:

$$\begin{cases} h_1 = \min\{g_1+g_2, g_2+g_3, g_3+g_1\} \\ h_2 = \max\{(g_1-g_2)^2, (g_2-g_3)^2, (g_3-g_1)^2\} \\ h_3 = \max\{(g_1+g_2)^2, (g_2+g_3)^2, (g_3+g_1)^2\} \end{cases}$$

When we assumed $\alpha < 2$ and $g_1 = g_2 = g_3 = 1$ we find:

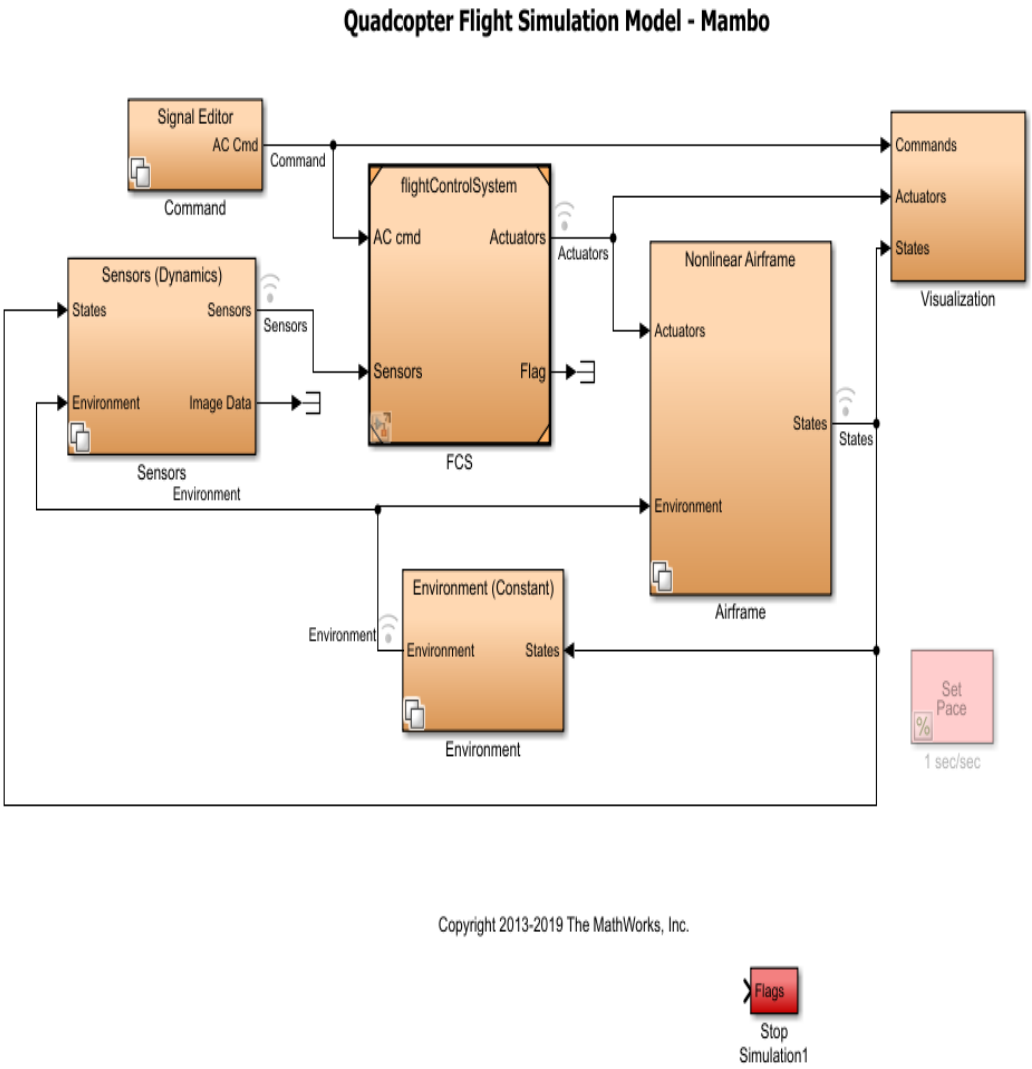
$$\frac{\|e_R{}^2\|}{2} \leq\ \rho \leq \frac{1}{2-\alpha}\|e_R{}^2\|$$

APPENDIX C

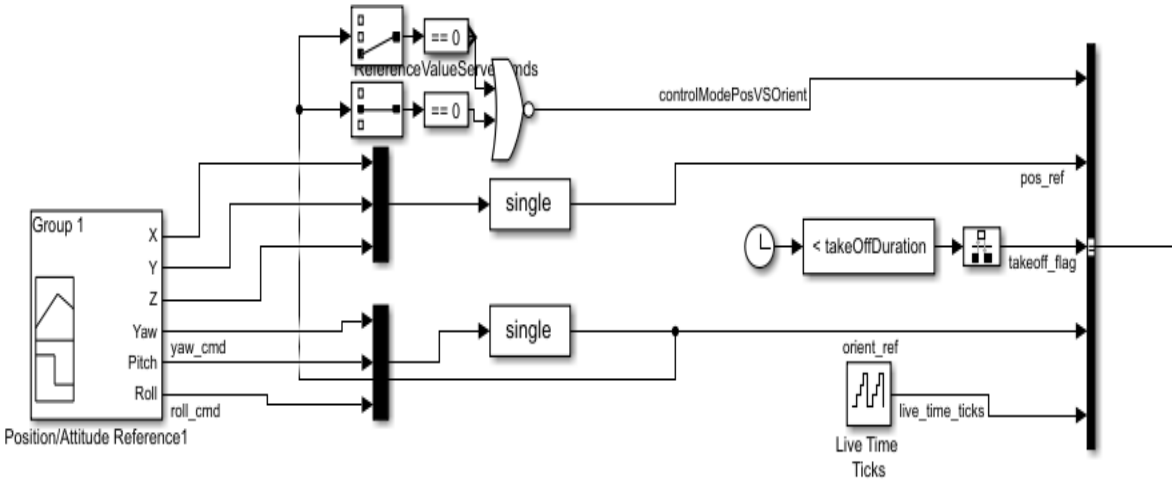
- Root Mean Square Error (RMSE) is a standard way to measure the error of a model in predicting quantitative data. Formally it is defined as follows:

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{(y_{refi}-y_i)^2}{n}}$$

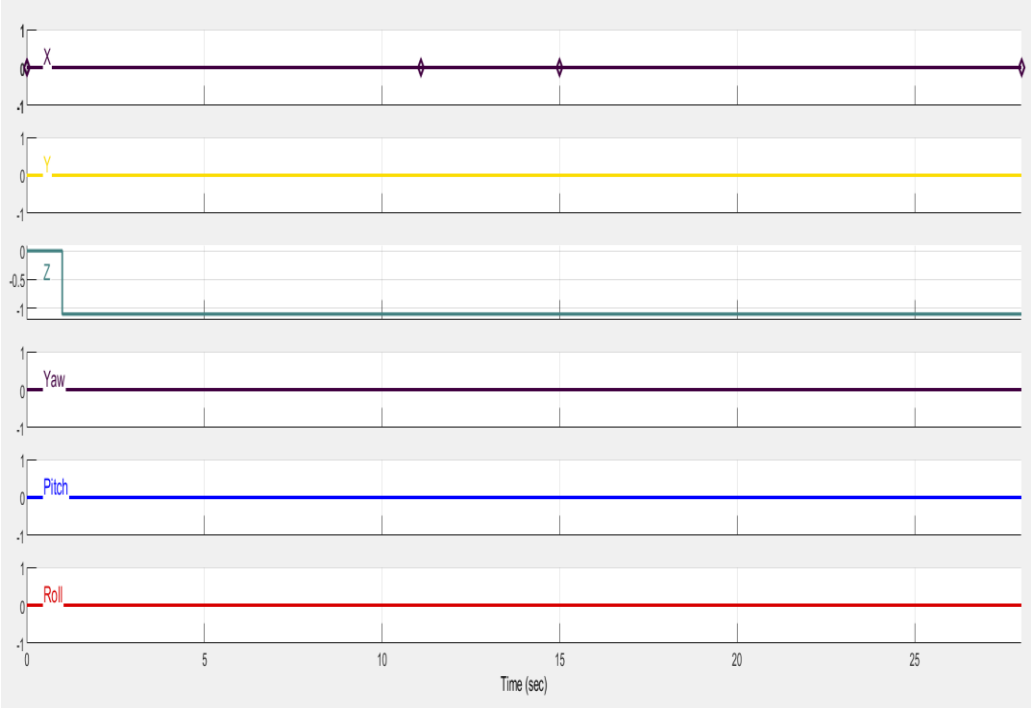
APPENDIX D Quadrotor Flight Simulation Model- Mambo



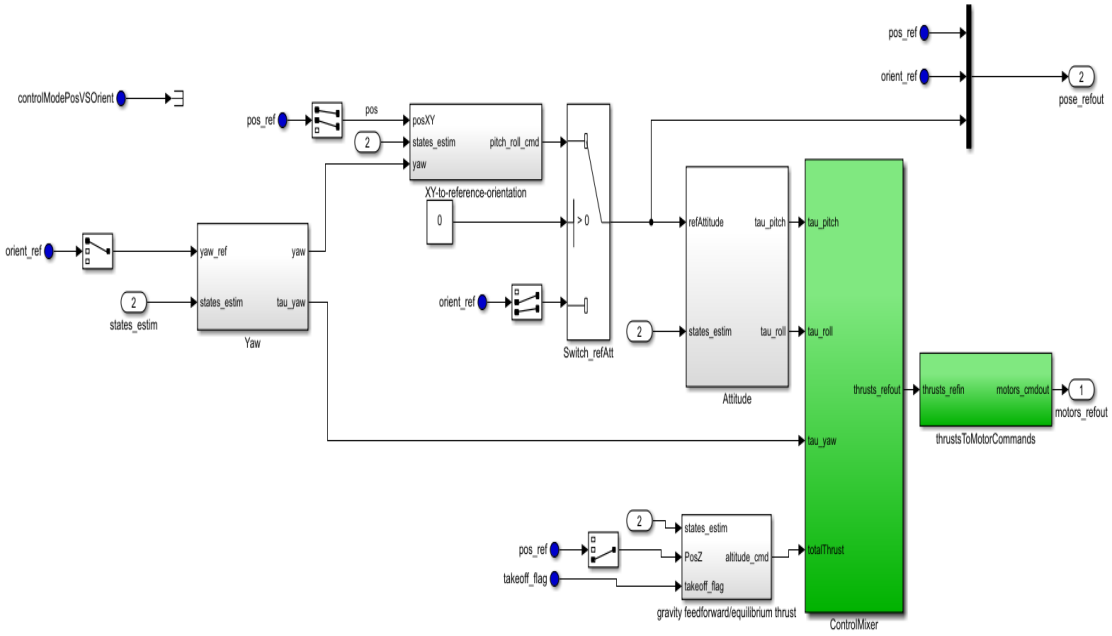
Quadrotor Flight Simulation Model- Mambo



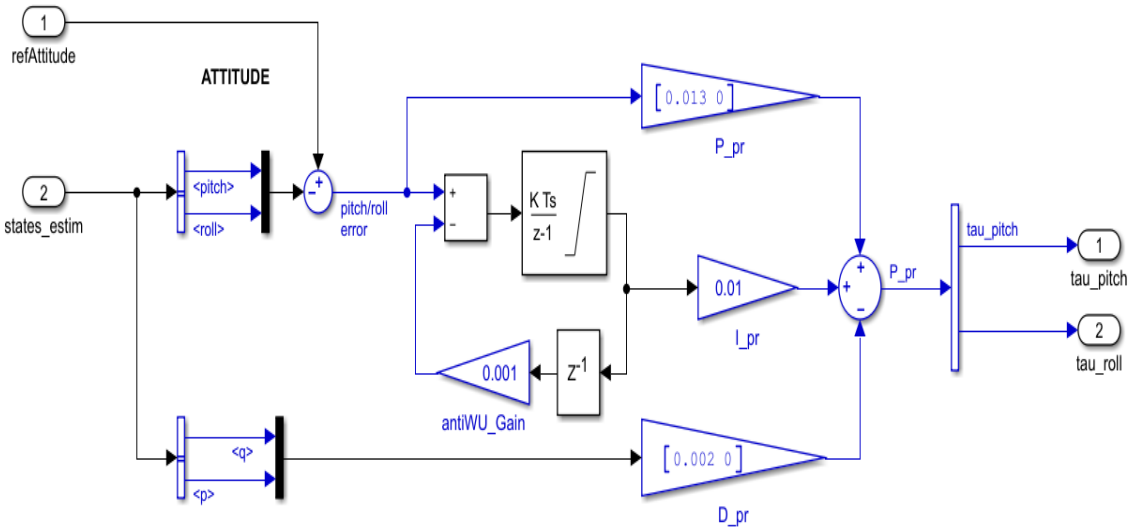
Command



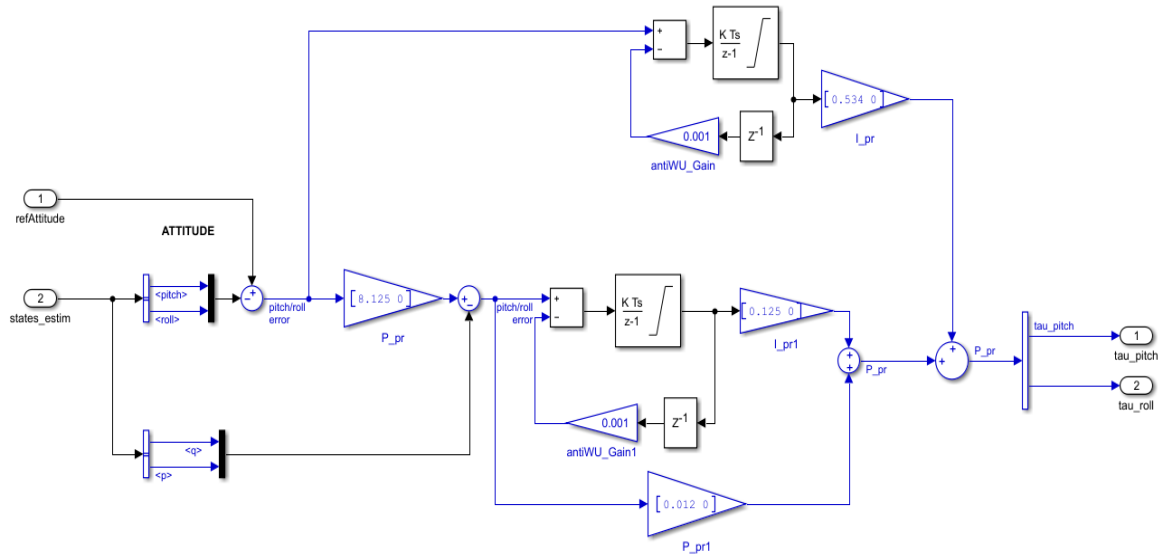
Position/Attitude Reference1



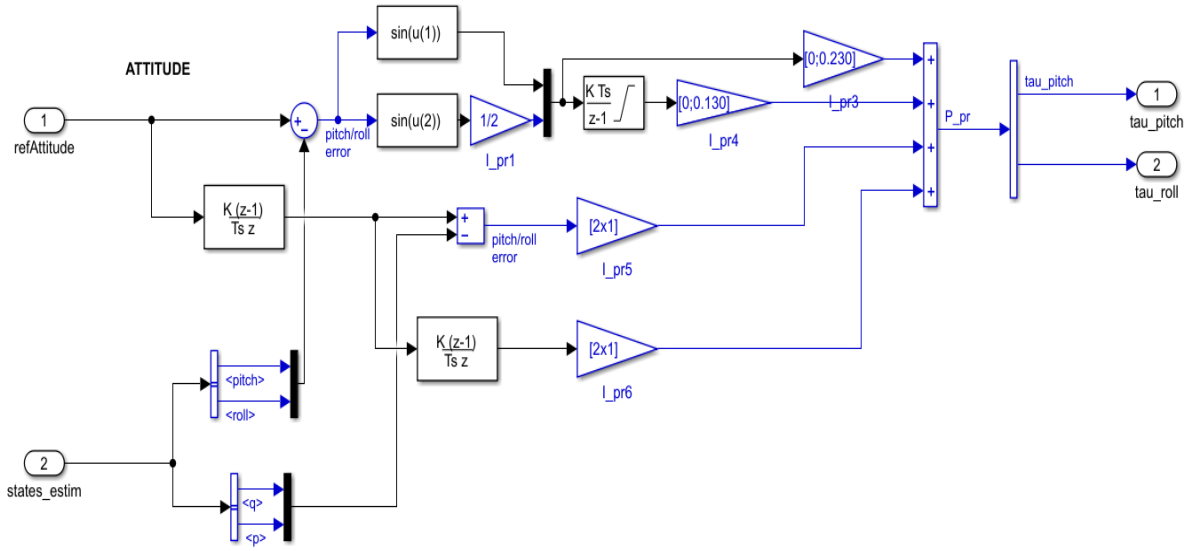
Flight Controller Block of IC



Attitude Block diagram of IC



Attitude Block diagram of PDH



Attitude Block diagram of BGC

الملخص

تتناول هذه الأطروحة النمذجة والتحكم في الروبوت الطائر ، وخاصة رباعي المحركات. تم تطوير النموذج الديناميكي وكذلك طرق التحكم مباشرة على (3) SO (مجموعة متعامدة خاصة) وعلى (3) SE (مجموعة إقليدية خاصة) ، على التوالي ، للارتفاع والموقف. ما سمح لنا بتجنب التفرد والغموض المرتبطين بتمثيلات المواقف الأخرى ، أثناء المناورات المعقدة للرباعي. في هذا السياق ، تم اقتراح عنصر تحكم تتبع جديد ؛ أطلقنا عليها اسم "التحكم "Backstepping الهندسي .هذا الأخير يضمن بساطة القوانين واستقرار السلوك. بالإضافة إلى ذلك ، تسمح طريقة التحكم ذات الصلة هذه للطائرة بدون طيار بمتابعة مسارها ، حتى في وجود اضطرابات داخلية وخارجية. لم يتم تقييم هذه التقنية فقط من خلال المحاكاة ، ولكن أيضًا من خلال الاختبارات التجريبية في الوقت الفعلي على طائر ببغاء مامبو المصغر. تمت مقارنة النتائج مع تلك التي تم الحصول عليها بطرق أخرى وأثبتت كفاءتها. يجب أن نلاحظ أننا نجحنا في تنفيذ التحكم في خوارزمية المواقف لببغاء مامبو من خلال استبدال التحكم في موقف "الصندوق الأسود" ، وهو الأمر الذي كان يتعذر الوصول إليه حتى وقت قريب. علاوة على ذلك ، تمكنا من تنفيذ طريقة أخرى ، من خلال إدخال التحكم في الأداء الديناميكي العالي من أجل زيادة ملاءمة المتانة. تم تقييم هذه التقنية أيضًا من خلال المحاكاة والتجريب في الوقت الفعلي. أثبتت النتائج كفاءتها وتفوقها فيما يتعلق بالموقف والسيطرة على الموقف

الكلمات الدالة

لروبوت الطائر ذو أربع محركات مامبو،مجموعة لي ((SO(3), SE(3)) مصفوفة الحركات الدائرية ، التحكم "Backstepping الهندسي ، التحكم في الأداء الديناميكي العالي

Abstrat

This thesis work deals with modelling and control of a flying robot, especially the quadrotor. The dynamic model as well as control methods have been directly developed on SO(3) (Special Orthogonal Group) and on SE(3) (Special Euclidean Group), respectively, for the attitude and the position. What allowed us to avoid the singularity and the ambiguity associated with other attitude representations, during complex manoeuvres of the quadrotor. In this context, a new tracking control is proposed; we named it the “Backstepping geometric control”. This latter ensures the simplicity of the laws and the stability of the behaviour. In addition, this pertinent method of control allows the drone to pursue its path, even in the presence of indoor and outdoor perturbations. This technique was not only evaluated through simulation, but also, through real time experimental tests on the Mambo parrot Minidrone. The results were compared to those obtained with other methods and they have proven their efficiency. We should note that we succeeded to implement our attitude algorithm control for the Mambo parrot by replacing its “black box” attitude control, a matter that was inaccessible until recently. Moreover, we managed to perform an another method, by introducing the High Dynamic Performance control in order to increase its robustness propriety. This technique, was, also, evaluated through simulation and experimentation in real time. The results proved its efficiency and superiority regarding the attitude and the position control.

Key words : Quadrotor, Mambo, LIE Group (SO(3), SE(3)), Rotation Matrix, Geometric Backstepping control, High Dynamic Perfermance control.

Résumé :

Ce travail de thèse traite la modélisation et la commande d’un robot volant, en particulier le quadrotor. Le modèle dynamique ainsi que les méthodes de contrôles ont été directement développés dans les ensembles *SO(3)* (*Special Orthogonal Group*) et *SE(3)* (*Special Euclidean Group*) respectivement pour l’attitude et la position. Cela nous a permis d’éviter la singularité et l’ambigüité, associées à d’autres représentations d’attitude, lors de manœuvres complexes du quadrotor. Dans ce contexte, une commande sous le nom «Backstepping géométrique » a été proposée, tout en garantissant la simplicité des lois et la stabilité du comportement. Cette méthode de contrôle pertinente permet au drone de poursuivre sa trajectoire, même en présence de perturbations internes et externes. Cette technique a été évaluée non seulement par simulation mais aussi en temps réel par des tests expérimentaux sur Mambo parrot Minidrone. Les résultats obtenus ont été comparés, à ceux obtenus avec d’autres méthodes, et ont bien prouvé leur validité. Notez que nous avons réussi à implémenter notre algorithme de commande d'attitude dans le Mambo parrot Minidrone en remplaçant son propre algorithme (commande impédance), chose qui était inaccessible jusqu'à récemment. De plus, nous avons réussi à appliquer une autre méthode, en introduisant le contrôle Haute Performance Dynamique afin d'augmenter sa propriété de robustesse. Cette technique, elle aussi, été évaluée par expérimentation en temps réel. Les résultats ont prouvé son efficacité et sa supériorité en matière de contrôle d'attitude.

Mots clés :

Quadrotor, Mambo, Groupe de Lie (SO(3), SE(3)), Matrice de rotation, Commande Backstepping géométrique, Commande Haute Performance Dynamique.