

# Comparative Results on 3D Navigation of Quadrotor using two Nonlinear Model based Controllers

Y Bouzid<sup>1</sup>, H Siguerdidjane<sup>2</sup> and Y Bestaoui<sup>1</sup>

<sup>1</sup> IBISC Laboratory, université d'Evry Val d'Essonne, Université Paris-Saclay, Evry, France

<sup>2</sup> L2S, CentraleSupélec, Université Paris-Saclay, Gif sur Yvette, France

**Abstract.** Recently the quadrotors are being increasingly employed in both military and civilian areas where a broad range of nonlinear flight control techniques are successfully implemented. With this advancement, it has become necessary to investigate the efficiency of these flight controllers by studying their features and compare their performance. In this paper, the control of Unmanned Aerial Vehicle (UAV) quadrotor, using two different approaches, is presented. The first controller is Nonlinear PID (NLPID) whilst the second one is Nonlinear Internal Model Control (NLIMC) that are used for the stabilization as well as for the 3D trajectory tracking. The numerical simulations have shown satisfactory results using nominal system model or disturbed model for both of them. The obtained results are analyzed with respect to several criteria for the sake of comparison.

## 1. Introduction

Due to its structural simplicity and understandable flight principle, the quadrotor UAV is considered as a good case study to analyze and to design flight control tools. This has attracted the interest of many researchers in aeronautics and robotics and many studies have led to the development of sophisticated nonlinear control laws [1-2]. Thus, several control techniques have been proposed in the literature as for instance linear ones based on a simplified model (LQR and PID)[3-4]. Unfortunately, they are used for limited operation point (hovering, attitude stabilization...). Also, strong disturbances are poorly rejected by this type of controllers. Some nonlinear controls (sliding mode, feedback linearization, etc.) have been employed where the ability of robustness against the model uncertainties is improved (see [5]). However, this type of controllers presents a number of drawbacks and may be inappropriate in some cases of operation (dynamic not sustained by the actuators, high energy at the start-up, complex implementation...).

The control design has to take in consideration some issues of practical implementation as the computational complexity, power limitation, robustness to parameter uncertainties, sensor noise, external disturbances... In this paper, an improvement has been brought in order to simplify the existing controllers, and thus overcome some issues poorly tackled by the classic linear or nonlinear techniques, thanks to Nonlinear PID (NLPID) and Nonlinear Internal Model Control (NLIMC).

Usually the kind of control called “NLPID control” stands for the regulator for which the coefficients are not assumed to be constant. Unlike this latter, our control law is defined in a novel way. It consists of a nonlinear controller, which is derived from a method based upon sliding mode theory and combined with a PID structure. Thus, this structure is achieved using sliding surface instead of the tracking error between the reference and the measured signal.

Among, the large variety of control techniques available in the literatures [6-8], the well-known reference model based technique called IMC, is popular in industrial process control applications due



to its good tradeoff between the control performance and disturbance rejection capability [9]. The controller employs the flatness property instead the classic dynamic inversion technique to overcome some of the feedback linearization limitations.

The aim of this research work is to stabilize the vehicle while ensuring the tracking of prescribed 3D trajectories with good performance and to ensure a given level of robustness with respect to disturbances. Therefore, two approaches are compared in order to check out the features of each of them with respect to the environment disturbance (noise, wind...) and to the type of prescribed reference trajectories (constant, time varying) using several criteria and scenarios.

This paper is organized as follows: Section 2 introduces the design of the proposed nonlinear approaches. Section 3 presents the application of these techniques to the quadrotor and the control architecture used in this stage. In Section 4, we investigate the features and the characteristics of each technique with respect to some scenarios that confirm the effectiveness of the proposed approach where a comparison study is presented. The final section gives some conclusions.

## 2. Nonlinear controllers design

### 2.1. NLPID controller design

This section deals with the design of the NLPID controller, which may be summarized in two steps: First one, we apply the classic sliding mode approach and then the obtained controller is, in a second step, combined with the PID structure. Consider a class of nonlinear SISO system for  $t \in [0, \infty)$  given by:

$$(\Sigma_x) \begin{cases} \dot{x} = \mathcal{F}(x) + G(x)u \\ y = h(x) \end{cases} \quad (1)$$

where  $x \in \mathbb{R}^n$  is an  $n$ -dimensional state vector,  $u \in \mathbb{R}$  is a scalar input,  $y \in D_y \subset \mathbb{R}$  is a scalar output,  $\mathcal{F}: D_x \rightarrow \mathbb{R}^n$  and  $G: D_x \rightarrow \mathbb{R}^n$  are  $n$ -dimensional vector functions sufficiently smooth on a domain  $D_x \subset \mathbb{R}^n$  and  $h(x)$  the output scalar function.

**Assumption 1:** There exists a diffeomorphism  $\Gamma: D_x \rightarrow \mathbb{R}^n$  such that  $D_\xi = \Gamma(D_x)$  contains the origin and  $\xi = \Gamma(x)$  is a change of variables that transforms the nonlinear system into an equivalent system:

$$(\Sigma_\xi) \begin{cases} \dot{\xi}_i(t) = \xi_{i+1}(t) \\ \dot{\xi}_n(t) = f(\xi) + g(\xi)u \\ y = \xi_1 \end{cases} \quad (2)$$

$i = 1 \dots n-1$ . Where  $f(\xi), g(\xi)$  denote continuous nonlinear functions and  $g(\xi)$  is nonsingular for all  $\xi \in D_\xi$ . Now, let us consider a general sliding surface form

$$s(e) = \left( \frac{d}{dt} + \lambda_y \right)^{n-1} e(t) \quad (3)$$

where  $e = y - y_r$  represents the tracking error between a reference trajectory  $y_r$  and the output  $y$  of a system and  $\lambda_y$  a positive constant. By expansion, equation (3) is written as

$$s(e) = e^{(n-1)}(t) + \sum_{i=0}^{n-2} \beta_i e^{(i)}(t) \quad (4)$$

where  $\beta_i, i = 0, \dots, n-2$  are positive tuning parameters provided that they are chosen in order to render the equilibrium,  $s(e) = 0$ , asymptotically stable. The first time derivative of  $s(e)$  is

$$\dot{s}(e) = e^{(n)}(t) + \sum_{i=0}^{n-2} \beta_i e^{(i+1)}(t) \quad (5)$$

The objective of control tracking is to design a control law such that the output  $y$  can track the desired trajectory  $y_r$ . In this regard, where *Assumption 1* holds, using the last component of system (2), equation (5) becomes

$$\dot{s}(e) = f(\xi) + g(\xi)u - y_r^{(n)}(t) + \sum_{i=0}^{n-2} \beta_i e^{(i+1)}(t) \quad (6)$$

Given a positive definite Lyapunov function candidate

$$V = \frac{1}{2} s^2(e) \quad (7)$$

So, the first time derivative leads to

$$\dot{V} = s(e)\dot{s}(e) \quad (8)$$

Note that the reachability condition of sliding mode control ensures the asymptotic stability ( $\dot{V}$  must be negative definite). Thus

$$\begin{cases} \dot{s}(e) < 0 \text{ when } s(e) > 0 \\ \dot{s}(e) > 0 \text{ when } s(e) < 0 \end{cases} \quad (9)$$

Assume that

$$\dot{s}(e) = -\kappa\gamma(s) \quad (10)$$

with  $\kappa$  being a strictly positive constant and  $\gamma(s)$  is a function defined by

$$\begin{cases} \gamma(s) < 0 \text{ if } s < 0 \\ \gamma(s) > 0 \text{ if } s > 0 \end{cases} \quad (11)$$

In fact,  $\gamma(s)$  may take different forms as for instance: saturation function, tangent function, etc. Thus, the discontinuous function  $\text{sign}(s)$ , represents the ideal sliding mode regime.

From (6) and (10), it immediately follows that

$$u = \frac{-1}{g(\xi)} \left\{ \kappa\gamma(s) + \sum_{i=0}^{n-2} \beta_i e^{(i+1)} - y_r^{(n)}(t) + f(\xi) \right\} \quad (12)$$

The discontinuous term  $\gamma(s)$  gives the controller a good level of robustness. However, the fast variation of the sliding surface gives rise of vibration in the system during the flight phase consequently affecting the control accuracy. So, in order to improve the performance of the controller and limiting the effect of the chattering phenomena, involved through the term  $\dot{s}(e)$ , we combine a PID structure and the controller determined above, namely

$$u = \frac{-1}{g(\xi)} \left\{ K_p \left( s(t) + \frac{1}{T_i} \int_0^t s(\tau) \cdot d\tau + T_d \frac{ds(t)}{dt} \right) + \sum_{i=0}^{n-2} \beta_i e^{(i+1)} - y_r^{(n)}(t) + f(\xi) \right\} \quad (13)$$

$K_p, T_d$  and  $T_i$  denote the proportional gain, the derivative and integral time constants respectively.

## 2.2. NLIMC controller design

The IMC controllers are used to be devoted to the linearized models and then implemented on the real plants [10]. The difference between the model and the real process gives rise to the poor performance of the closed-loop system. Therefore, the linear IMC scheme has been extended to non-linear versions by different approaches [11]. Moreover, due to the complexity of nonlinear systems, the development of a general IMC nonlinear strategy has serious difficulties. To accomplish IMC control, it requires the inverse of the plant. this inversion is more involved and analytical solutions are too much complex to obtain. Motivated by these gaps and requirements (inversion, open loop structure, linear model, computational complexity...), we develop in the following, a novel and simple analytic formulation for the design of NLIMC synthesized for nonlinear Single-Input, Single-Output (SISO) systems that involves a closed loop control structure.

The direct mission is to investigate the controller that implies the dynamic model inverse and occurs in closed loop control architecture. Thus, our procedure split into two steps in order to accomplish the controller design. The first step is to formulate a direct relationship between the output and the tracking error. In other words

$$y^{(i)}(t) = \Theta_i(e, e, \dot{e}, \dots, e^{(i-1)}) \quad i = 0, \dots, n \quad (14)$$

where  $\Theta_i$  are scalar smooth functions. For this purpose, we consider a surface defined in the state-space  $D_x$  as

$$\dot{e} + \frac{1}{\mu} e = 0 \quad (15)$$

where  $\mu$  is a positive time tuning parameter providing a compromise between performance and robustness.

**Assumption 2:** The reference trajectories allowed for this study are considered piecewise constant in domain  $D_y \subset \mathbb{R}$ .

From equation (15), we have  $\dot{e}(t) = -\frac{1}{\mu} e(t)$  and Assumption 2 allows writing:  $y^{(1)}(t) = \frac{e(t)}{\mu}$ . Then, by successive time derivatives,  $y^{(i)}(t)$  can be written as

$$y^{(i)}(t) = \frac{e^{(i-1)}(t)}{\mu} \quad i = 2, \dots, n. \quad (16)$$

In addition, equation (16) can be written under general form Introducing two parameters,  $(\alpha_1, \beta_1) \in \mathbb{R} \times \mathbb{R}$  verifying  $\alpha_1 + \beta_1 = 1$ , we obtain

$$y^{(i)}(t) = \frac{\alpha_1}{\mu} e^{(i-1)}(t) + \frac{\beta_1}{\mu} (y_r - y(t))^{(i-1)} \quad i = 2, \dots, n$$

Recall that *Assumption 2* holds, then

$$y^{(i)}(t) = \frac{\alpha_1}{\mu} e^{(i-1)}(t) - \frac{\beta_1}{\mu} y^{(i-1)}(t) \quad i = 2, \dots, n$$

By recurrence, we obtain a general form

$$y^{(i)}(t) = \sum_{k=1}^{k=i} (-1)^{k+1} \frac{\alpha_{(k)} \prod_{j=1}^k \beta_{(j-1)}}{\mu^k} e^{(i-k)}(t) \quad i = 2, \dots, n \quad (17)$$

with initial term  $y^{(1)}(t) = \frac{e(t)}{\mu}$

$(\alpha_{(k)}, \beta_{(k)}) \in \mathbb{R} \times \mathbb{R}$  verifying  $\alpha_{(k)} + \beta_{(k)} = 1$ , with  $\alpha_{(1)} = 1$  and  $\beta_{(0)} = 1$ . For more detail, the reader may refer to [12].

The inversion of the nonlinear model requires the relationship between the input  $u$  and the output  $y$ . A broad range of techniques already published investigating this classic problem. So that in this second step, we exploit the flatness property. The flat output and its time derivatives describe the system dynamics, since their knowledge is sufficient to compute all the variables  $x(t), u(t)$  and their time derivatives. Then, the state-space model (1) may be summarized by the following differential equation.

$$y^{(n)}(t) + \rho(y, y^{(1)}, \dots, y^{(n-1)}, u, u^{(1)}, \dots, u^{(\sigma-1)}) + \vartheta(y, y^{(1)}, \dots, y^{(n-1)}, u, u^{(1)}, \dots, u^{(\sigma-1)})u^{(\sigma)} = 0 \quad (18)$$

where  $\rho$  and  $\vartheta$  are two nonlinear multi-variable functions and  $r = n - \sigma$  denotes the relative degree of the system ( $\sigma \leq n$ ). Our objective is to design controller that drives the system output to track a desired trajectory. Let us write again equation (18) in terms of the tracking error by using first relationship (17).

$$y^{(n)}(e, \mu, \alpha_{(k)}, \beta_{(k)}) + \rho(e, \alpha_{(k)}, \beta_{(k)}, \mu, u, u^{(1)}, \dots, u^{(\sigma-1)}) + \vartheta(\alpha_{(k)}, \beta_{(k)}, e, \mu, u, u^{(1)}, \dots, u^{(\sigma-1)})u^{(\sigma)} = 0 \quad (19)$$

The controller then is obtained by

$$u^{(\sigma)}(t) = - \frac{y^{(n)}(e(t), \mu, \alpha_{(k)}, \beta_{(k)}) + \rho(e(t), \alpha_{(k)}, \beta_{(k)}, \mu, u, u^{(1)}, \dots, u^{(\sigma-1)})}{\vartheta(\alpha_{(k)}, \beta_{(k)}, e(t), \mu, u, u^{(1)}, \dots, u^{(\sigma-1)})} \quad (20)$$

By integration, the control effort  $u(t)$  is computed.

### 2.3. Discussion and properties

In this section, we discuss the main features, the advantages and the limitations of the proposed controllers. Controller (13) exhibits several benefits, and can be split up into three parts: The first part is designed in order to compensate the system dynamics (2) (feedback linearization). The second part consists of a linear combination of the tracking error dynamics,  $\sum_{i=0}^{i=n-2} \beta_i \varepsilon_y^{(i+1)}$ , and the  $n^{th}$  derivative of the reference trajectory,  $y_r^{(n)}(t)$ , that allows the linearization of the closed loop system after the act of the first part. The remaining part includes the PID structure, which represents the additional control needed for control accuracy and disturbance rejection. From equation (13), we observe the absence of discontinuities associated to jumps in the control action, which clearly eliminates the chattering problem. In addition, plant-model mismatch and the steady state errors are canceled by adding the integral action that penalizes the deviations between the output and its set point. However, the derivative term of the proposed controller induces a higher order derivative one with respect to that needed for classical “feedback linearization”. Unfortunately, this is a drawback because of the additional noise and derivative estimation inaccuracy.

The classic dynamic inversion (feedback linearization) technique involves the cancelation of the nonlinearities. Moreover, it is applied for a specific class of systems especially the canonical one (2).

However, if the model is inaccurate or contains some varying variables that are highly influenced by the environment changes (temperature, pressure ...), this cancelation may be not exact and engenders another equilibrium points of the closed loop system or causes the instability. This drawback is tackled by the NLIMC controller where the plant-model mismatch is canceled by using the tracking error in the inversion process. Therefore, the controller presents a good level of robustness, especially, with respect to the model uncertainties. Moreover, using the flatness property allowing the system behaves in the original coordinates where the characteristic can be specified naturally. NLIMC approach uses only the considered outputs and their dynamics  $(y(t), \dot{y}(t), \dots, y^{(m)}(t))$ , while the dynamic inversion in the NLPID involves beside the outputs all the states of the system. In this last case and for real implementation purpose, the control loop requires a broad range of sensors or estimators that affect the accuracy of the control because of the noise and the uncertainties especially that the feedback linearization is well-known by its lack of robustness. The NLPID uses the inversion of the system dynamics and this leads to numerical problem nearby the singularities. The NLIMC uses tunable gains  $\mu$  that allows to overcome such issue. Unfortunately, for high order systems, NLIMC suffers from the problem of numerical noise and derivative estimation inaccuracy. In addition, as the NLIMC is designed for piecewise reference trajectories, the steady state errors are poorly canceled for time varying reference trajectories. These properties are shown down through a series of numerical simulations.

### 3. Quadrotor application

#### 3.1. Dynamics background and control architecture

The quadrotor has four rotors with twin-bladed propellers. The system operates in two coordinate frames: the inertial fixed frame and the body frame. Let  $\eta = (\varphi, \theta, \Psi)^T$  describes the orientation of the aerial vehicle (Roll, Pitch, Yaw) and  $\chi = (x, y, z)^T$  denotes its absolute position. We use a simplified model that is derived by Bouzid et al. in [12] and describes the quadrotor in-flight behavior. It may decompose into translational subsystem and rotational subsystem as

$$\ddot{\chi} = \begin{pmatrix} u_1 \frac{c_\psi s_\theta c_\varphi + s_\psi s_\varphi}{m} \\ u_1 \frac{s_\psi s_\theta c_\varphi - c_\psi s_\varphi}{m} \\ -g + u_1 \frac{c_\theta c_\varphi}{m} \end{pmatrix} \quad (21), \quad \ddot{\eta} = \begin{pmatrix} \dot{\theta} \dot{\Psi} \left( \frac{I_y - I_z}{I_x} \right) + \frac{u_2}{I_x} \\ \dot{\phi} \dot{\Psi} \left( \frac{I_z - I_x}{I_y} \right) + \frac{u_3}{I_y} \\ \dot{\phi} \dot{\theta} \left( \frac{I_x - I_y}{I_z} \right) + \frac{u_4}{I_z} \end{pmatrix} \quad (22)$$

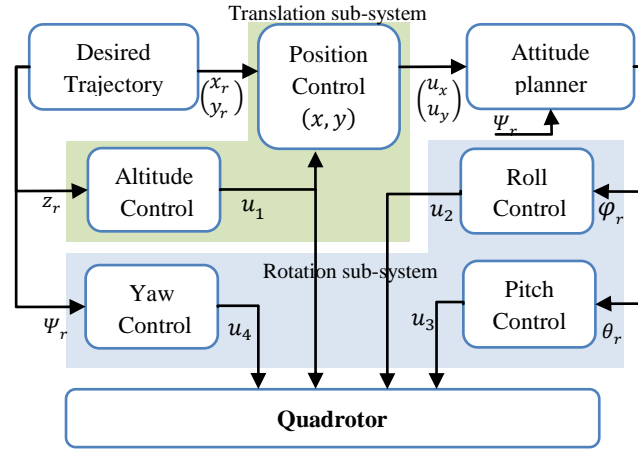
$s_{(\cdot)}$  and  $c_{(\cdot)}$  are abbreviations for  $\sin(\cdot)$  and  $\cos(\cdot)$  respectively.  $m$  denotes the mass,  $g$  the gravity acceleration,  $I = \text{diag}(I_x, I_y, I_z)$  is the diagonal inertia matrix,  $u_1$  is the total thrust of four rotors and  $\tau = (u_2, u_3, u_4)^T$  is the control torque. The system parameters are depicted in Table 1.

**Table 1.** Quadrotor parameters.

$m(kg)$	0.429	$I_y(kg.m^2)$	0.0029
$I_x(kg.m^2)$	0.0022	$I_z(kg.m^2)$	0.0048

These proposed techniques are herein applied to the quadrotor by taking care of having an adequate control structure (see Figure 1). The system has two dynamics, such that the first (slow outer loop) relates to the position control and generates the reference angles  $(\varphi_r, \theta_r)$  to the second one (fast inner loop) via attitude planner (24). This inner loop is related to the orientation control.  $x$  and  $y$  are controlled through two virtual drives  $(u_x, u_y)$  to allow the system to reach the prescribed references  $x_r$  and  $y_r$  respectively. The roll, pitch and yaw angles are controlled by  $(u_2, u_3, u_4)$  respectively, whereas the altitude is controlled by  $u_1$ . This control structure shown in Figure 1 is designed to ensure the tracking of any reference trajectory described by the absolute position with a given yaw angle. In fact, we will consider six inputs  $(u_x, u_y, u_1, u_2, u_3, u_4)$  and six outputs  $(x, y, z, \varphi, \theta, \Psi)$ . Notice that this control architecture allows applying the developed approaches that concern the single input single

output systems to the quadrotor. This choice is done in order to simplify the design and reduce the computational complexity.



**Figure 1.** Quadrotor control architecture.

### 3.2. Autopilot design

The tracking errors are defined as:  $e_x = x - x_r$ ,  $e_y = y - y_r$ ,  $e_z = z - z_r$ ,  $e_\phi = \phi - \phi_r$ ,  $e_\theta = \theta - \theta_r$  and  $e_\psi = \psi - \psi_r$ . Translation dynamics (21) can be divided into three other sub-systems along the three axis (X, Y, Z). Each sub-system has one input ( $u_x, u_y, u_1$ ) and one output ( $x, y, z$ ) respectively. We first start with  $u_1$  for the altitude motion. Once this command is calculated we then proceed in the same way for  $u_x$  and  $u_y$  by considering  $u_1$  as time varying parameter, with

$$\begin{cases} u_x = c_\psi s_\theta c_\phi + s_\psi s_\phi \\ u_y = s_\psi s_\theta c_\phi - c_\psi s_\phi \end{cases} \quad (23)$$

Then, the required reference angles of the roll and pitch rotations are given by

$$\begin{cases} \phi_r = \arcsin\left(\text{sat}(u_x \sin \psi_r - u_y \cos \psi_r)\right) \\ \theta_r = \arcsin\left(\text{sat}\left(\frac{u_x \cos \psi_r + u_y \sin \psi_r}{\cos \phi}\right)\right) \end{cases} \quad (24)$$

where the function  $\text{sat}(\cdot)$  is defined as

$$\text{sat}(a) = \begin{cases} 1, & a > 1 \\ a, & |a| \leq 1 \\ -1, & a < -1 \end{cases}$$

Similarly, system (22) can be divided into three sub-systems for the roll, pitch and yaw rotations. Each one of them has one input ( $u_2, u_3, u_4$ ) and one output ( $\phi, \theta, \psi$ ) respectively. For the NLIPID, we make the choice of the surfaces as

$$\begin{cases} s_x = \dot{e}_x + \beta_{0x} e_x \\ s_y = \dot{e}_y + \beta_{0y} e_y \\ s_z = \dot{e}_z + \beta_{0z} e_z \\ s_\phi = \dot{e}_\phi + \beta_{0\phi} e_\phi \\ s_\theta = \dot{e}_\theta + \beta_{0\theta} e_\theta \\ s_\psi = \dot{e}_\psi + \beta_{0\psi} e_\psi \end{cases} \quad (25)$$

with  $\beta_{0x}, \beta_{0y}, \beta_{0z}, \beta_{0\phi}, \beta_{0\theta}$  and  $\beta_{0\psi}$  are positive constants.

Applying (13), we obtain

$$\begin{aligned} u_x &= \frac{-m}{u_1} \left\{ K_{px} \left( s_x(t) + \frac{1}{T_{ix}} \int_0^t s_x(\tau) d\tau + T_{dx} \frac{ds_x(t)}{dt} \right) + \beta_{0x} \dot{e}_x - \ddot{x}_r \right\} \\ u_y &= \frac{-m}{u_1} \left\{ K_{py} \left( s_y(t) + \frac{1}{T_{iy}} \int_0^t s_y(\tau) d\tau + T_{dy} \frac{ds_y(t)}{dt} \right) + \beta_{0y} \dot{e}_y - \ddot{y}_r \right\} \end{aligned}$$

$$\begin{aligned}
u_1 &= \frac{-m}{c_\theta c_\varphi} \left\{ K_{pz} \left( s_z(t) + \frac{1}{T_{iz}} \int_0^t s_z(\tau) \cdot d\tau + T_{dz} \frac{ds_z(t)}{dt} \right) + \beta_{0z} \dot{\varepsilon}_z - g - \ddot{z}_r \right\} \\
u_2 &= -I_x \left\{ K_{p\varphi} \left( s_\varphi(t) + \frac{1}{T_{i\varphi}} \int_0^t s_\varphi(\tau) \cdot d\tau + T_{d\varphi} \frac{ds_\varphi(t)}{dt} \right) + \beta_{0\varphi} \dot{\varepsilon}_\varphi + \dot{\theta} \dot{\Psi} \left( \frac{I_y - I_z}{I_x} \right) - \ddot{\varphi}_r \right\} \\
u_3 &= -I_y \left\{ K_{p\theta} \left( s_\theta(t) + \frac{1}{T_{i\theta}} \int_0^t s_\theta(\tau) \cdot d\tau + T_{d\theta} \frac{ds_\theta(t)}{dt} \right) + \beta_{0\theta} \dot{\varepsilon}_\theta + \dot{\varphi} \dot{\Psi} \left( \frac{I_z - I_x}{I_y} \right) - \ddot{\theta}_r \right\} \\
u_4 &= -I_z \left\{ K_{p\psi} \left( s_\psi(t) + \frac{1}{T_{i\psi}} \int_0^t s_\psi(\tau) \cdot d\tau + T_{d\psi} \frac{ds_\psi(t)}{dt} \right) + \beta_{0\psi} \dot{\varepsilon}_\psi + \dot{\varphi} \dot{\theta} \left( \frac{I_x - I_y}{I_z} \right) - \ddot{\psi}_r \right\}
\end{aligned}$$

$K_{p()}, T_{i()}, T_{d()}$  denote the proportional gains, the integral and derivative time constants respectively of the NLPID structure.

For NLIMC, we apply (20)

$$\begin{aligned}
u_x &= \left( \frac{\alpha_{x(1)}}{\mu_x} \dot{e}_x - \frac{\beta_{x(1)}}{\mu_x^2} e_x \right) \frac{m}{u_1} \\
u_y &= \left( \frac{\alpha_{y(1)}}{\mu_y} \dot{e}_y - \frac{\beta_{y(1)}}{\mu_y^2} e_y \right) \frac{m}{u_1} \\
u_1 &= \frac{m \left( \frac{\alpha_{z(1)}}{\mu_z} \dot{e}_z - \frac{\beta_{z(1)}}{\mu_z^2} e_z + g \right)}{\cos \left( \frac{1}{\mu_\varphi} \int_0^t e_\varphi(\tau) d\tau + \varphi_0 \right) \cos \left( \frac{1}{\mu_\theta} \int_0^t e_\theta(\tau) d\tau + \theta_0 \right)} \\
u_2 &= I_x \left( \frac{\alpha_{\varphi(1)}}{\mu_\varphi} \dot{e}_\varphi - \frac{\beta_{\varphi(1)}}{\mu_\varphi^2} e_\varphi - \frac{e_\theta}{\mu_\theta} \frac{e_\psi}{\mu_\psi} \left( \frac{I_y - I_z}{I_x} \right) \right) \\
u_3 &= I_y \left( \frac{\alpha_{\theta(1)}}{\mu_\theta} \dot{e}_\theta - \frac{\beta_{\theta(1)}}{\mu_\theta^2} e_\theta - \frac{e_\varphi}{\mu_\varphi} \frac{e_\psi}{\mu_\psi} \left( \frac{I_z - I_x}{I_y} \right) \right) \\
u_4 &= I_z \left( \frac{\alpha_{\psi(1)}}{\mu_\psi} \dot{e}_\psi - \frac{\beta_{\psi(1)}}{\mu_\psi^2} e_\psi - \frac{e_\varphi}{\mu_\varphi} \frac{e_\theta}{\mu_\theta} \left( \frac{I_x - I_y}{I_z} \right) \right)
\end{aligned}$$

$\mu_x, \mu_y, \mu_z, \mu_\varphi, \mu_\theta$  and  $\mu_\psi$  denote tuning positive parameters.  $(\alpha_{(\cdot)(1)}, \beta_{(\cdot)(1)}) \in \mathbb{R} \times \mathbb{R}$  are chosen according to the stability requirements where  $\alpha_{(\cdot)(1)} + \beta_{(\cdot)(1)} = 1$ .

#### 4. Comparison study

For the sake of significant comparison, this study is done considering the same conditions and following the same protocol by investigating the performance of the controllers for several scenarios and trajectories. Thus, the parameters of NLPID and NLIMC are obtained through the optimization of the Integral Square Error (ISE) criterion.

$$ISE = \frac{1}{t_f - t_i} \int_{t_i}^{t_f} (q - q_r)^T (q - q_r) dt \quad (26)$$

where  $q$  is the output vector,  $q_r$  its reference trajectory.  $t_f$  and  $t_i$  denotes the final and the initial optimization instances respectively. For more efficient optimization, the parameters are optimized independently for each basic mode of flight. Thus, we consider a rotation of  $\frac{\pi}{2}$  radian around Z-axis, vertical flight, longitudinal flight and lateral flight of one meter along the Z, X and Y-axes respectively. The process of optimization holds during a flight of 20 seconds. The obtained control parameters for both NLPID and NLIMC are depicted in Table 2. Before studying the features of the controllers, we investigate their computational complexity via the number of additions and multiplications.

In this case, a quick view in the Table 3, allows checking out that the NLPID is more gourmand in terms of computation because of the considered number of derivations and the integrations compared to the NLIMC.

In order to compare, investigate and evaluate the control laws, various simulations have been performed and numerous scenarios have been proposed on the complete closed loop system (ideal case, parameters uncertainties, extra payload, noisy measurements and wing gust). The main metrics used for the comparison are:

- ISE: is computed for the three translations and the yaw angle that allows to compare the accuracy of the controllers for a given configuration in the space using (26). In fact, using ISE, we cannot distinguish the nature of these errors (overshoot, oscillations, steady state...). Therefore, we employ beside this metric a visual graphics that allow identifying the nature of these errors.
- Settling time ( $T_s$ ): it measures the rapidity of convergence of a closed loop time response.
- Consumed energy: This practical constraint makes the use of controllers that consume less energy more preferable. Herein, we use the following criterion

$$\int_{t_i}^{t_f} u_1^2 dt \quad (27)$$

**Table 2.** Controllers parameters.

	NLIMC	NLPID
$x$	$\mu_x = 0.5$	$k_{p_x} = 3.392, T_{i_x} = 0.7, T_{d_x} = 0.5$
$y$	$\mu_y = 0.5$	$k_{p_y} = 3.229, T_{i_y} = 0.7, T_{d_y} = 0.672$
$z$	$\mu_z = 0.1$	$k_{p_z} = 2.829, T_{i_z} = 3.488, T_{d_z} = 0.867$
$\varphi$	$\mu_\varphi = 3.9$	$k_{p_\varphi} = 0.731, T_{i_\varphi} = 0.37, T_{d_\varphi} = 0.439$
$\theta$	$\mu_\theta = 3.9$	$k_{p_\theta} = 5.161, T_{i_\theta} = 0.198, T_{d_\theta} = 0.439$
$\Psi$	$\mu_\Psi = 0.39$	$k_{p_\Psi} = 20.346, T_{i_\Psi} = 16.009, T_{d_\Psi} = 0.007$
	$\beta_{(\cdot)} = -0.5$	$\beta_{(\cdot)} = 1$

**Table 3.** Computational complexity of the autopilots.

Type	NLPID	NLIMC
Multiplication	49	56
Additions	43	29
Integration	6	2
Time derivation	First: 15, Second: 6	First: 6, Second: 0
Nonlinear functions	2	2

#### 4.1. Ideal case without disturbance

In this section, we investigate the efficiency of the controllers in the ideal case and the performance of the time responses of the closed loop system. Therefore, the quadrotor is supposed to flight in ideal environment (without disturbances) and has a perfect model. The proposed trajectories are composed of successive straight-line segments and circular trajectory.

- *Successive straight line segments trajectory*

The first proposed scenario allows us to test, independently, all the flight modes of the quadrotor. It is composed of successive straight-lines along the three axes X, Y, and Z and a rotation around Z-axis (yaw). This trajectory is used to overview the performance of the controllers for each type of flights: vertical flight ( $z$ ), longitudinal flight ( $x, \theta$ ), lateral flight ( $y, \varphi$ ) and yaw rotation ( $\Psi$ ). Moreover, we will test the behavior of the vehicle at the inter-connection points.

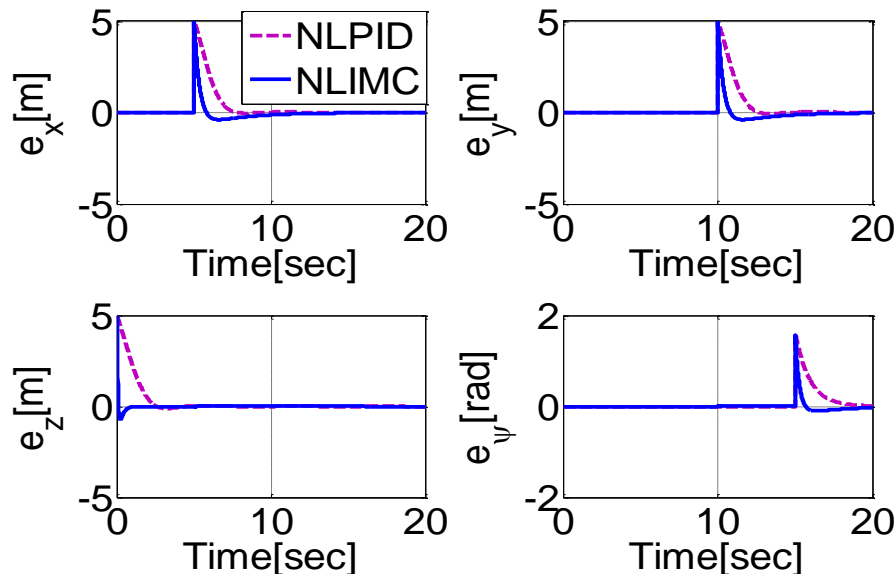
The control gains used herein are the same depicted in Table 2 and the obtained results are summarized in Table 4. The time responses of the tracking errors are presented in Figure 2.

**Table 4.** Controllers features.

Criteria		NLPID	NLIMC
$ISE$	X	0.7595	0.2062
	Y	0.7595	0.2062
	Z	0.7959	0.0420
	<i>Yaw</i>	0.0624	0.0158
	<i>Total</i>	2.3773	0.4702
Energy		$2 \cdot 10^3$	$1.2055 \cdot 10^5$
Settling time	$T_{s_x}$	2.165	3.17



(sec)	$T_{sy}$	2.179	3.16
	$T_{sz}$	2.193	0.545
	$T_{syaw}$	2.94	0.559
Maximal attitude angles (rad)	Roll	1.2494	0.3998
	Pitch	1.3581	0.4003

**Figure 2 :** Tracking errors.

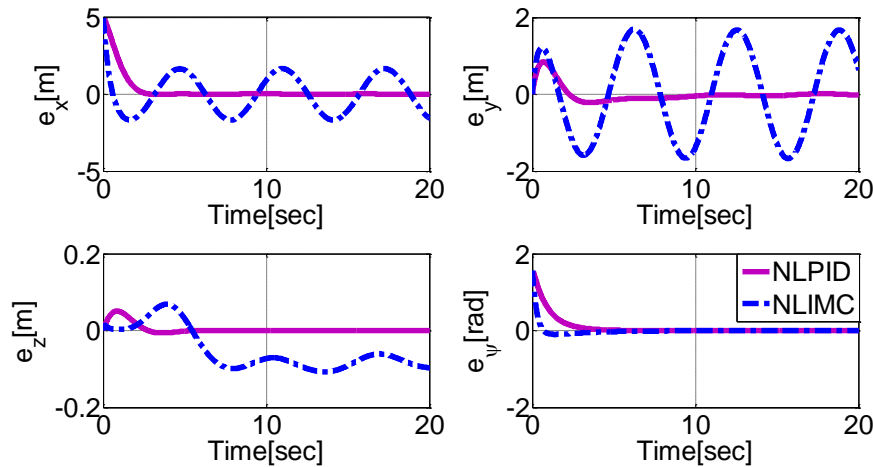
From Table 4, one can observe that the NLIMC ensures more accurate control compared to the NLPID. However, the consumed energy is of course greater. NLIMC has slower responses in the longitudinal and lateral motions compared to the NLPID and faster responses for the altitude and the yaw controls. It can be clearly seen in Figure 2 that a small overshoot appears when using the NLIMC especially for Z, which is reasonable because of the rapidity of convergence.

- Trajectory tracking

We consider a helix trajectory to check the efficiency of the controller in the tracking of time-varying reference trajectory especially that the NLIMC makes the assumption of piecewise constant trajectories. The obtained results are tabulated in Table 5. The time responses of the tracking errors are presented in Figure 3.

**Table 5.** Controllers features.

Criteria		NLPID	NLIMC
<i>ISE</i>	X	0.7692	1.4362
	Y	0.0401	1.2747
	Z	$1.46 \cdot 10^{-4}$	0.0057
	Yaw	0.0624	0.0158
	Total	0.8718	2.7324
Energy		457.8235	432.1417
Maximal attitude angles	Roll	0.9681	0.5911
	Pitch	0.8711	0.7244



**Figure 3 :** Tracking errors.

From Table 5, a remarkable degradation of accuracy for the NLIMC compared to the NLPID is reported where a steady state appears along the entire trajectory. The results are not strange because the NLIMC considers constant reference trajectories. However, in this stage, the controller consumes less energy compared to the NLPID.

#### 4.2. With disturbance

- *Parameters uncertainties:* It is so hard to express, exactly and explicitly, all the forces and moments due to the complexity of the aerodynamic effects. Naturally, there is a notable mismatch between the real dynamics and the model exploited for the control design. Moreover, due to the limit of the onboard electronic computational capacities, some forces and terms are simplified or neglected. The derived dynamic model of the quadrotor contains a number of parameters that are identified according to the real system. The inertia matrix can be influenced by the environment changes and lead to some uncertainties. Therefore, the controller should be able to ensure good performance even in presence of these uncertainties. In this test all these parameters are underestimated:  $I_{(c)} = 1.75 \cdot \tilde{I}_{(c)}$  where  $\tilde{I}$  denotes the parameters employed in the controllers.
- *Additional Payload:* Another test of robustness is proposed considering additional payload that represents 75% of the global mass. The payload is supposed added in the centre of mass.
- *Sensors noise:* In general, the signals provided by the sensors are noisy. The noise may be cause the vibration and oscillation of the vehicle and in certain cases the instability of the overall system. We consider thus an additive noise with all the outputs using the function `rand` of Matlab to generate stochastic values uniformly distributed between 0-1. The additive noise is selected equals to  $0.1 \cdot \text{rand}(\cdot)$  for the position and  $0.01 \cdot \text{rand}(\cdot)$  for the attitude.
- *Gust of wind:* The objective is to check the robustness of the controllers under the effect of gust of wind, without taking in consideration the model of the disturbed vehicle. Often the wind model is expressed in function of the velocities along the three axes.

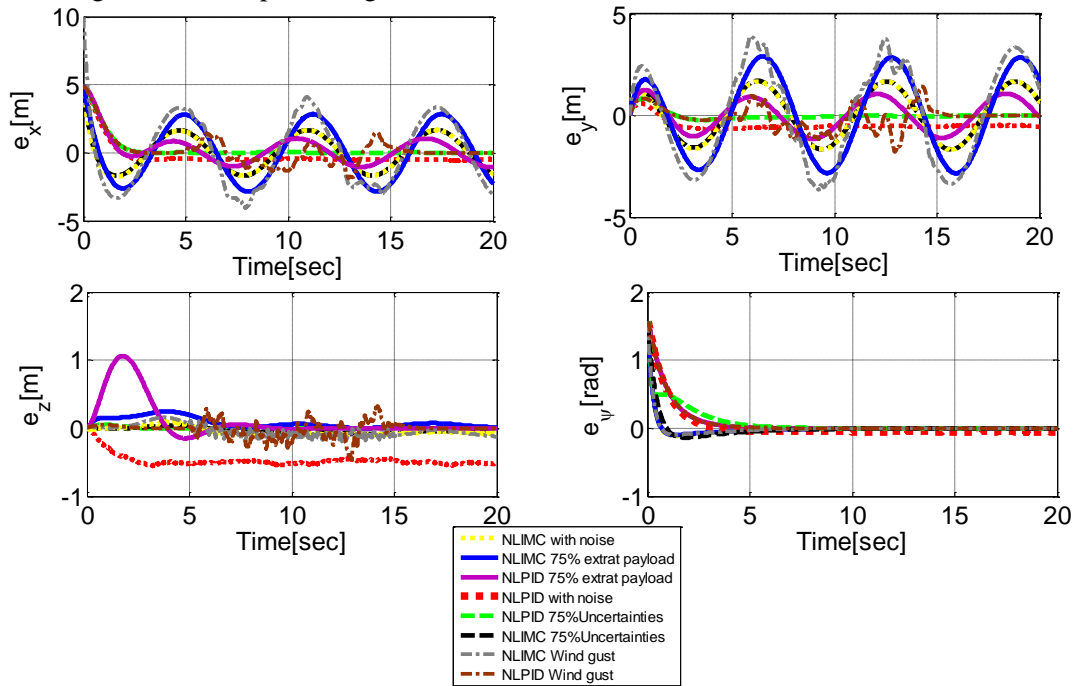
$$\begin{cases} u_w = \bar{u}_w + u_d \\ v_w = \bar{v}_w + v_d \\ w_w = \bar{w}_w + w_d \end{cases} \quad (28)$$

Where  $\bar{u}_w, \bar{v}_w, \bar{w}_w$  denote the mean wind velocities. The terms  $u_d, v_d$  and  $w_d$  stand for the deviations from the mean velocity. The mean wind speed is set according to the daily-use situation. Then turbulence gust models are superimposed on this mean wind where the data are collected from statistical properties (spectra, scale length, turbulence intensity...). The wind velocities along the inertial axes are generated via Dryden linear filters as

$$\begin{cases} \frac{u_d(s)}{g_u(s)} = H_u(s) \\ \frac{v_d(s)}{g_v(s)} = H_v(s) \\ \frac{w_d(s)}{g_w(s)} = H_w(s) \end{cases} \quad (29)$$

$g_u(s)$ ,  $g_v(s)$  and  $g_w(s)$  are zero-mean white Gaussian noise.  $s$  is the Laplace variable. In order to generate a turbulence signal, a band-limited white noise signal passes through an appropriate filter. For more details, the reader may refer to [13].

The obtained results through the previous scenarios are summarized by Table 6 and the time responses of the tracking errors are depicted Figure 4.



**Figure 4.** Tracking errors.

**Table 6.** Controllers features.

	criteria	NLPID	NLIMC
Uncertainties +75%	$ISE_{Total}$	0.8506	2.7404
	Energy	444.7810	390.6749
Additional Payload 75%	$ISE_{Total}$	1.8701	7.6439
	Energy	$1.4518 \cdot 10^3$	$31.2679 \cdot 10^3$
Sensors noise 0.1	$ISE_{Total}$	69.8358	3.3110
	Energy	$2.4688 \cdot 10^3$	$8.6112 \cdot 10^3$
Wind gust	$ISE_{Total}$	1.5800	3.0350
	Energy	484.3860	$3.7606 \cdot 10^3$

The controllers exhibit good level of robustness with respect to all the disturbances mentioned above. They consume more energy to keep the same performance compared to the ideal case except the parameters uncertainties where the consumed energy still the same. An overview of Table 6, we observe that the NLPID controller exhibits superior performance compared to the NLIMC controller. This superiority relevant to the poor capability of NLIMC to track time varying reference trajectories, which leads to a persistent steady state error along the entire trajectory (see Figure 4). However, also from Table 6, the NLPID exhibits poor capabilities to tackle the noise.

Through our analyze, we have concluded that the NLIMC is more robust than the NLPID that has better performance. From Figure 4 and for the altitude and the yaw angle, the NLIMC behaves better contrary to the longitudinal and lateral motion. We have concluded also that the attitude control subsystem is more sensitive to noise for both of them.

## 5. Conclusion

Two nonlinear approaches have been described namely Nonlinear Internal Model Control and Nonlinear PID. Numerical simulations were carried out in order to evaluate their effectiveness. The performance of the proposed approaches is demonstrated in multiple test scenarios. The settling time is shown to be quite fast and a good level of robustness is ensured with respect to parameters uncertainties, extra payload, wind gust and the noise. The NLIMC have superior performance for piecewise constant reference trajectories while NLPID is better for time varying reference trajectories. It is shown that the NLIMC guarantees better level of robustness compared to the NLPID with less computational complexity. To guarantee good performance, expect the case of parameter uncertainties, it was observed that the required energy increases where NLIMC consumes more than the NLPID. A rigorous proof of the stability of the whole system (translational and rotational) will be reported in the future works. Some issues also still open, among them is to reduce the consumed energy in presence of the disturbance where the controller is supposed to request more energy in order to keep the same performance as the ideal case.

## References

- [1] Pounds P, Mahony R., and Corke P 2010 Modelling and control of a large quadrotor robot *Control Engineering Practice*. 18(7), 691–699. <http://doi.org/10.1016/j.conengprac>.
- [2] Huang H, Hoffmann G. M, Waslander S. L and Tomlin C. J 2009 Aerodynamics and Control of Autonomous Quadrotor Helicopters in Aggressive Maneuvering *Proceedings of the 2009 IEEE International Conference on Robotics and Automation* (pp. 2408–2413). Piscataway, NJ, USA: IEEE Press.
- [3] Salih A. L, Moghavvemi M, Mohamed H. A. F and Gaeid K. S 2010 Modelling and PID controller design for a quadrotor unmanned air vehicle. In *IEEE International Conference on Automation Quality and Testing Robotics (AQTR)* (Vol. 1, pp. 1–5). <http://doi.org/10.1109/AQTR.2010.5520914>
- [4] Reyes-Valeria E, Enriquez-Caldera R, Camacho-Lara S and Guichard, J 2013 LQR control for a quadrotor using unit quaternions: Modeling and simulation. In *2013 International Conference on Electronics, Communications and Computing (CONIELECOMP)* (pp. 172–178). <http://doi.org/10.1109/CONIELECOMP>.
- [5] Mian A. A and Daobo, W 2008 Nonlinear Flight Control Strategy for an Underactuated Quadrotor Aerial Robot. In *IEEE International Conference on Networking, Sensing and Contro*. pp. 938–942. <http://doi.org/10.1109/ICNSC>.
- [6] Slotine J.-J.E and Li W 1991 *Applied nonlinear control* Prentice Hall.
- [7] Khalil H 2002 *Nonlinear Systems* Prentice Hall;
- [8] Utkin V.I 1992 *Sliding modes in control and optimization* Springer-Verlag.
- [9] Morari M and Zafriou E 1989 *Robust process control* Prentice Hall.
- [10] Economou C.G, Morari M 1986 Internal model control: extension to nonlinear systems *Ind. Eng. Chem. Process Des. Dev.* pp. 403-411.
- [11] Nahas E. P, Henson M. A, and Seborg D. E 1992 Nonlinear internal model control strategy for neural network models *Computers & Chemical Engineering*, vol. 16, no. 12, pp. 1039–1057.
- [12] Bouzid Y, Siguerdidjane H and Bestaoui Y 2016 Improved 3D trajectory tracking by Nonlinear Internal Model-Feedback linearization control strategy for autonomous systems 6th IFAC Symposium on System Structure and Control, Istanbul.
- [13] Aboudonia A, Rashad R. and El-Badawy, A 2015 Time domain disturbance observer based control of a quadrotor unmanned aerial vehicle In *2015 XXV International Conference on Information, Communication and Automation Technologies (ICAT)* (pp. 1–6).