Improved 3D trajectory tracking by Nonlinear Internal Model-Feedback linearization control strategy for autonomous systems

Yasser Bouzid, Houria Siguerdidjane, Yasmina Bestaoui

Improved 3D trajectory tracking by Nonlinear Internal Model-Feedback linearization control strategy for autonomous systems

Y. Bouzid*. H. Siguerdidjane**. Y. Bestaoui*

** IBISC, Université d'Evry-Val-d'Essonne, Université Paris-Saclay, Evry, France*
(e-mail : Yasser.Bouzid@ufrst.univ-evry.fr, Yasmina.Bestaoui@ufrst.univ-evry.fr)
***L2S, CentraleSupélec, Université Paris-Saclay, Gif-sur-yvette, France*
(e-mail : Houria.Siguerdidjane@centralesupelec.fr)

Abstract: This paper presents a nonlinear controller for 3D trajectory tracking of an autonomous helicopter. The main idea consists of combining feedback linearization controller together with a novel nonlinear IMC control. This approach allows more robustness, fast and good trajectory tracking. It is applied to a small, eight-rotor, Square-Shaped Octo-Rotor and has shown satisfactory results using adequate control architecture. The controller effectiveness is shown through numerical simulations and confirmed using a software simulator and real tests.

Keywords: Nonlinear control, Autonomous system, Trajectory tracking.

1. INTRODUCTION

Rotorcrafts have many applications because of their vertical landing/take-off capability and payload. Among these rotorcrafts, Octo-Rotor helicopter may usually afford a larger payload than conventional helicopter due to the eight rotors. For these advantages, the Octo-Copter has received much interest from students and researchers where some models have been presented for different shapes (Star-Shaped: Alwi 2013; Stocia et al. 2012 and 4Y-shaped: Adir et al. 2012; ...).

In order to achieve complicated missions, it is necessary to design controller such that the system will be able to follow predefined trajectories, particularly, in the presence of disturbances. This is the reason for why many studies have led to the development of nonlinear control laws. The feedback linearization control law is, as known, one of the most popular nonlinear control methods, which has been the subject of many books (see for example Slotine et al. 2012; Khalil 2002, Isidori 1985). In addition, the nonlinear control results permit a global asymptotic stability provided that no singular points exist. Many controllers allow good set-point tracking. However, for almost all processes control, disturbance rejection is much more significant. Hence, controller design that emphasizes disturbance rejection rather than a good set point tracking is of a great interest and a real design problem. Our focus is consequently pushed on this last point. Among the large variety of control techniques available in the literature, a model based control method, namely the Internal Model Control (IMC), is popular in industrial process control applications (Muhammad et al. 2010) due to its disturbance rejection capability and robustness (Morari et al. 1989). The aim of this research work is to stabilize the helicopter while ensuring the tracking of complex trajectories with a precise way, and also to ensure a given level of robustness with respect to structured and unstructured uncertainties. For this purpose, a novel Nonlinear IMC-Feedback linearization control is herein described by taking care of having an adequate control structure.

This paper is organized as follows: Section 2 introduces the dynamics of the Octo-Rotor and its operating principle. Section 3 presents the synthesis of the so-called nonlinear IMC- feedback linearization control and its application is developed in section 4. In section 5, the controller effectiveness is shown through numerical simulations and confirmed by experimental tests under different operating conditions. Finally conclusions are given.

2. OCTOROTOR DYNAMICS

In this section, we give a brief explanation of the system used to obtain the complete model, and which describes the UAV's behaviour. The Octo-Copter is controlled by the angular speeds of eight electric motors. Each motor produces a thrust and a torque, whose combination creates the main thrust, the yaw, the pitch and the roll torques acting on the UAV. As shown in Fig. 1, the system operates in two coordinate frames: the earth fixed frame $R_0(O_0, X, Y, Z)$ and the body frame $R_1\ (O_1, X_1, Y_1, Z_1)$. Let $\chi = (x, y, z)^T \in R^3$ be the absolute position of the system and $\eta = (\varphi, \theta, \Psi)^T \in \left]-\frac{\pi}{2}, +\frac{\pi}{2}\right[\times \left]-\frac{\pi}{2}, +\frac{\pi}{2}\right[\times]-\pi, +\pi[$ be the Euler angles (roll, pitch, and yaw) that describe the orientation of the aircraft. In order to obtain a usable model for the control synthesis of control laws, it is necessary to make a certain number of approximations and assumptions.

Assumptions:

1) The structure and propellers are rigid and perfectly symmetrical
2) The gyroscopic and ground effects are neglected.

3) The UAV has very small upper bounds on $|\varphi|$ and $|\theta|$ in such a way that the differences $|\varphi| - sin(\varphi)$ and $|\theta| - tan(\theta)$ are arbitrarily small.

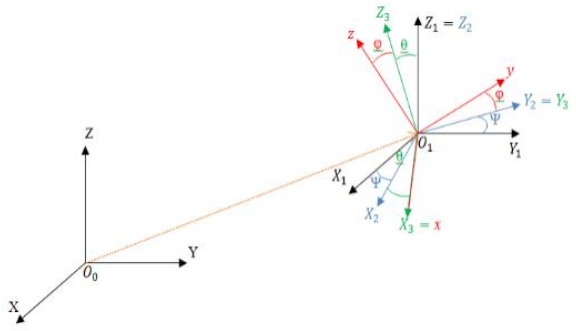

Fig. 1. Frames representation.

The dynamic model of the Octo-Rotor is classically derived from the Newton-Euler law. Gravity force acts on the center of mass in the negative Z direction in the earth frame. The global thrust is in the positive Z direction of the body frame. Therefore, the translational dynamic equations of the Octo-Copter can be expressed in earth frame as follows,

$$m\ddot{\chi} = (-mg + u_T R)e_z \quad (1)$$

, using the rotation matrix R ∈ SO(3):

$$R(\varphi,\theta,\Psi) = \begin{bmatrix} c_\Psi c_\theta & c_\Psi s_\theta s_\varphi - s_\Psi c_\varphi & c_\Psi s_\theta c_\varphi + s_\Psi s_\varphi \\ s_\Psi c_\theta & s_\Psi s_\theta s_\varphi + c_\Psi c_\varphi & s_\Psi s_\theta c_\varphi - c_\Psi s_\varphi \\ -s_\theta & c_\theta s_\varphi & c_\theta c_\varphi \end{bmatrix}$$

$s_()$ and $c_()$ are abbreviations for sin(.) and cos(.) respectively. Where g denotes the gravity acceleration, m the mass, $e_z = (0,0,1)^T$ the unit vector expressed in the earth frame, and u_T the global thrust produced by the eight rotors of speeds Ω_i. Each motor M_i (for $i = 1,\dots,8$) produces the thrust force T_i , so that

$$u_T = \sum_{i=1}^{i=8} T_i = b\sum_{i=1}^{i=8} \Omega_i^2 \quad (2)$$

Where b is the thrust factor.

Yaw motion is generated by the differential drag forces D_i. Pitch and Roll motions, are created as the difference in combined thrust for opposite sides of the vehicle,

$$D_i = d\Omega_i^2 \quad i = 1,\dots,8 \quad (3)$$

Where d is the drag factor.

The rotational dynamic equation of an Octo-Rotor can be written as follows:

$$I\dot{\varpi} = -\varpi \times I\varpi - G_a + \tau \quad (4)$$

Where $\varpi = (\varpi_x, \varpi_y, \varpi_z)^T$ is the angular velocity vector, $I = diag(I_x, I_y, I_z)$ is the diagonal inertia matrix and G_a is the gyroscopic effect, while τ is the control torque obtained by varying the rotor speeds. $\tau = (\tau_\varphi, \tau_\theta, \tau_\Psi)^T$ is defined for square shaped Octo-Copter as

$$\begin{pmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \begin{pmatrix} L(T_7 - T_3) + l\frac{\sqrt{2}}{2}(T_6 + T_8 - T_2 - T_4) \\ l\frac{\sqrt{2}}{2}(T_6 + T_4 - T_2 - T_8) - L(T_1 - T_5) \\ l(D_2 + D_4 + D_6 + D_8) - L(D_1 + D_3 + D_5 + D_7) \end{pmatrix} \quad (5)$$

Where L represents the distance from the motors placed on the long arms to the center of mass and l the distance from the motors placed on the short arms (see Fig. 2). The gyroscopic effects G_a are neglected according to assumption (2) considered above. Also, translational velocity and acceleration are defined in the earth fixed frame R_0, and angular velocity and acceleration are defined in the body fixed frame R_1.

$$\dot{\varpi} = I^{-1}(-\varpi \times I\varpi + \tau) \quad (6)$$

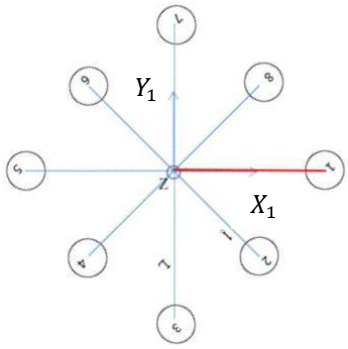

Fig. 2. Simplified diagram of the drone with 8 propellers.

Regarding the angular dynamics, the angular velocities of the drone ϖ are transformed into Euler angular speeds $\dot{\eta}$. This yields

$$\begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\Psi} \end{pmatrix} = \begin{bmatrix} 1 & s_\varphi \tan\theta & c_\varphi \tan\theta \\ 0 & c_\varphi & -s_\varphi \\ 0 & s_\varphi / c_\theta & c_\varphi / c_\theta \end{bmatrix} \begin{pmatrix} \varpi_x \\ \varpi_y \\ \varpi_z \end{pmatrix} \quad (7)$$

Then, by using equations (1-7) and accepting assumption (3), the dynamic model of the vehicle in terms of position χ and rotation η is finally written as

$$\ddot{\chi} = \begin{cases} \frac{c_\Psi s_\theta c_\varphi + s_\Psi s_\varphi}{m} u_T \\ \frac{s_\Psi s_\theta c_\varphi - c_\Psi s_\varphi}{m} u_T \\ -g + \frac{c_\theta c_\varphi}{m} u_T \end{cases} \quad (8)$$

$$\ddot{\eta} = \begin{cases} \dot{\theta}\dot{\Psi}\left(\frac{I_y - I_z}{I_x}\right) + \frac{\tau_\varphi}{I_x} \\ \dot{\varphi}\dot{\Psi}\left(\frac{I_z - I_x}{I_y}\right) + \frac{\tau_\theta}{I_y} \\ \dot{\varphi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) + \frac{\tau_\Psi}{I_z} \end{cases} \quad (9)$$

3. NON LINEAR CONTROLLER DESIGN

3.1 Nonlinear IMC design procedure

Let us recall, for a linear system, quite briefly the IMC basic principle: if the control system, contains partial or complete representation of the process to be controlled, then accurate control can be achieved. In Fig. 3, $U(s)$ is the input of both process $G(s)$ and its model $\tilde{G}(s)$. $D(s)$ is an unknown disturbance acting on the system. The output $Y(s)$ is compared with the output of the model $Y_m(s)$, resulting in a signal $\tilde{Y}(s)$, that is,

$$\tilde{Y}(s) = \left(G(s) - \tilde{G}(s)\right)U(s) + D(s) \quad (10)$$

Let $C(s)$ be the inverse of the model $C(s) = \tilde{G}(s)^{-1}$, if this one is invertible, then the output will perfectly track the reference input $Y_r(s)$. Nevertheless, this controller needs to be multiplied by the so-called IMC-filter. This latter makes the resulting controller proper and realizable (for details see Morari et al. 1989).

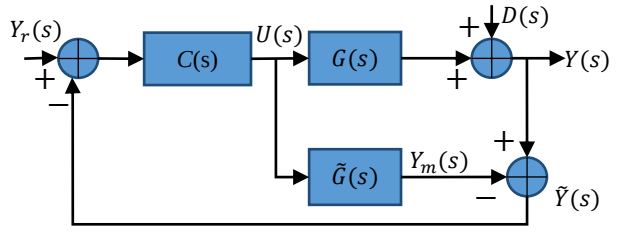

Fig. 3. IMC basic principle.

The difference between the process and the model gives rise to the poor performance of the closed-loop. Therefore, the linear IMC scheme has been extended to a non-linear version by different approaches (see for example Economou & Morari. 1986; Dong et al. 1998). Moreover, to accomplish tracking, the IMC requires the inverse of the process. The inversion of nonlinear models is more involved, so that problems can be solved using numerical techniques, such as the Newton-Raphson method (Economou al. 1986), which is too much complex to obtain. So, hereafter, we propose a more simple and effective analytical solution.

Consider a class of nonlinear single-input, single-output system for $t \in [0, \infty)$ given by:

$$(\sum_x)\begin{cases}\dot{x} = \mathcal{F}(x) + G(x)u \\ y = h(x)\end{cases} \quad (11)$$

Where $x \in R^n$ is an n-dimentional state vector, $u \in R$ is a scalar input, $y \in R$ is a scalar output, $\mathcal{F}: D_x \dashrightarrow R^n$ and $G: D_x \dashrightarrow R^n$ are n-dimentional vector functions sufficiently smooth on a domain $D_x \subset R^n$ and $h(x)$ the output scalar function. In the following, the controller design u, is described step by step. Let us define the tracking error, $e(t) = y_r(t) - y(t)$, as the difference between the reference trajectory $y_r(t)$ and the output $y(t)$. The reference trajectories allowed for this study are considered piecewise constant. Assuming that the tracking error satisfies a first order differential equation

$$\dot{e}(t) + \frac{1}{\mu}e(t) = 0 \quad (12)$$

Because the reference trajectory is constant, equation (12) is equivalent to

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

So,

$$y^{(i)}(t) = \frac{1}{\mu}e^{(i-1)}(t) \quad i = 1, \dots, n \quad (13)$$

And

$$y(t) = \frac{1}{\mu}\int_0^t e(\tau)d\tau + y_0$$

Where μ is a positive time tuning parameter providing a compromise between performance and robustness.

Remark 1: When the input is constant, $y^{(i)}(t)$ may be written, in different forms, according to available signals. This is explained thereafter.

Introducing two parameters (α, β) to equation (13), we get

$$y^{(i)}(t) = \frac{\alpha}{\mu}\, e^{(i-1)}(\mathrm{t}) + \frac{\beta}{\mu}\left(y_r(t) - y(t)\right)^{(i-1)}(\mathrm{t}), i = 1, \dots, n$$

Then

$$y^{(i)}(t) = \frac{\alpha}{\mu}\, e^{(i-1)}(\mathrm{t}) - \frac{\beta}{\mu}\, y(t)^{(i-1)}(\mathrm{t}), i = 1, \dots, n$$

By recurrence, the general form may be written as

$$y^{(i)}(t) = \sum_{k=1}^{k=i}(-1)^{k+1}\frac{\alpha_{(k)}\prod_{j=1}^{k}\beta_{(j-1)}}{\mu^k}e^{(i-k)} \quad i = 1, \dots, n \quad (14)$$

$(\alpha_{(k)}, \beta_{(k)}) \in R \times R$ verifying $\alpha_{(k)} + \beta_{(k)} = 1$, with $\alpha_{(i)} = 1$ and $\beta_{(0)} = 1$ are chosen to select the available signals ensuring asymptotic stability. However, the inversion of a nonlinear model requires an input-output relationship. For this purpose, we use the flatness property of systems.

A. *Flatness properties*

Nonlinear system (11) is called flat if there is a flat output variable $\varsigma(t)$, such that, the following statements are satisfied (see for instance Nitsche et al. 2007):

- The flat variable $\varsigma(t)$ can be expressed in terms of the state vector $x(t)$.

$$\varsigma(t) = \Theta\big(x(t)\big) \quad (15)$$

- $x(t), u(t)$ can be expressed in terms of $\varsigma(t)$ and a finite number of its time derivatives. In other words,

$$\begin{cases} x(t) = \Upsilon_1\left(\varsigma(\mathrm{t}), \dot{\varsigma}(\mathrm{t}), \dots, \varsigma^{(n-1)}(\mathrm{t})\right) \\ u(t) = \Upsilon_2\left(\varsigma(\mathrm{t}), \dot{\varsigma}(\mathrm{t}), \dots, \varsigma^{(n)}(\mathrm{t})\right) \end{cases} \quad (16)$$

The flat output $\varsigma(\mathrm{t})$ and its time derivatives describe the system dynamics, since their knowledge is sufficient to compute all the variables $x(\mathrm{t}), u(t), y(t)$ and their time derivatives. This allows computing the following differential equation,

$$y^{(n)}(t) + \rho\big(y, y^{(1)}, \dots, y^{(n-1)}, u, u^{(1)}, \dots, u^{(m-1)}\big) + \vartheta\big(y, y^{(1)}, \dots, y^{(n-1)}, u, u^{(1)}, \dots, u^{(m-1)}\big)u^{(m)}(\mathrm{t}) = 0 \quad (17)$$

which summarizes state-space model (11) where ρ is a nonlinear function and ϑ is a nonlinear and invertible function, $m \leq n$ and $r = n - m$ denotes the relative degree of the system.

Substituting equations (13) into equation (17) gives

$$u^{(m)}(t) = -\frac{\frac{1}{\mu}e^{(n-1)}(\mathrm{t}) + \rho\left(\int e, e, e^{(1)}, \dots, e^{(n-2)}, \mu, u, u^{(1)}, \dots, u^{(m-1)}\right)}{\vartheta\left(\int e, e, e^{(1)}, \dots, e^{(n-2)}, \mu, u, u^{(1)}, \dots, u^{(m-1)}\right)} \quad (18)$$

By integration, the control effort $u(t)$ is computed. In the case where $n = r$, we obtain $u(t)$ as

$$u(t) = -\vartheta^{-1}\left(\int e, e, e^{(1)}, \dots, e^{(n-2)}, \mu\right)\left(\frac{1}{\mu}e^{(n-1)}(\mathrm{t}) + \rho\left(\int e, e, e^{(1)}, \dots, e^{(n-2)}, \mu\right)\right) \quad (19)$$

Remark 2: control law (18) represents the model inverse dynamic. It depends on the tracking error and the tuning parameter.

3.2 Feedback linearization design procedure

Now, recall that $(\sum_x)$ is called input-state linearizable, if it exists: a diffeomorphism $\Gamma: D_x \dashrightarrow R^n$ such that $D_\xi = \Gamma(D_x)$ contains the origin, and a change of variables $\xi = \Gamma(x)$ that transforms the system into an equivalent linear one:

$$(\tilde{\Sigma}_{\xi})\begin{cases} \dot{\xi}_1 = \xi_2 \\ \dots \\ \dot{\xi}_i = \xi_{i+1} \\ \dots \\ \dot{\xi}_n = f(\xi) + g(\xi)u \\ y = \xi_1 \end{cases} \quad (20)$$

The controller is obtained by two types of nonlinearities cancellation . From system (20) we get:

$$u(t) = g(\xi)^{-1}(-f(\xi) + v(t)) \quad (21)$$

Where $g(\xi)$ is invertible in a domain D_{ξ}. Generally, $v(t)$, is given by: $v(t) = \sum_{i=1}^{i=n} k_i e^{(i-1)}(t) + y_r{}^{(n)}$

The coefficients k_i are chosen to get the asymptotic stability.

Remark 3*:* The difference between Nonlinear IMC control (19) and feedback linearization one (21) is that

- The first one uses the inversion of the system dynamics while the second uses the cancellation of the nonlinearities implying inversion.
- As consequence, the feedback linearization can be concluded through the NLIMC approach.

Combining these approaches (Fig. 4), we can write

$$u(t) = g(\xi)^{-1}(-f(\xi) + \frac{1}{\mu}e^{(n-1)}(t)\ ) \quad (22)$$

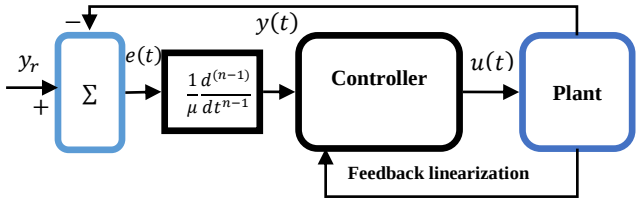

Fig .4. Nonlinear IMC-Feedback control.

Notice that this approach is convenient for a large class of systems and may be extended to Multi input-Multi output (MIMO) systems. Often we don't have an accurate definition about the model (neglected dynamics, uncertainties and external disturbances) which make the use of the proposed approach particular important.

4. OCTO-COPTER APPLICATION

This section shows how the proposed method that we described above can be applied to an octo-copter in order to determine the controllers. The control structure (Fig. 5) is herein based on the decomposition into four sub-systems. The first one concerns the altitude control while the second one is for the yaw orientation; the others two sub-systems are related to the lateral and longitudinal controls. Through the control efforts τ_{θ} and τ_{φ}, the state x and the output y are controlled to allow the system to reach their references x_r and y_r respectively while the pitch and the roll angles (φ, θ) are stabilised about the origin. However, the altitude is controlled by u_T and the yaw angle is controlled by τ_{Ψ}.

4.1 Altitude control

We first calculate u_T for the altitude motion

$$\ddot{z} = -g + \frac{\mathrm{c}_{\theta}\mathrm{c}_{\varphi}}{m}u_T(\mathrm{t}) \quad (23)$$

The linearizing control is then written as

$$u_T(t) = \frac{\mathrm{m}}{\mathrm{c}_{\theta}\mathrm{c}_{\varphi}}\left(g + v_z(t)\right) \quad (24)$$

First of all, for comparison, use individually the different approaches:

- Feedback linearization leads to

$$u_T = \frac{\mathrm{m}}{\mathrm{c}_{\theta}\mathrm{c}_{\varphi}}(g + k_2 e_z(t) + k_1 \dot{e_z}(t) + z_r{}^{(2)})$$

- Nonlinear Feedback- IMC approach, gives

$$u_T = \frac{\mathrm{m}}{\mathrm{c}_{\theta}\mathrm{c}_{\varphi}}(g + \sum_{k=1}^{k=2}(-1)^{k+1}\frac{\alpha_{z(k)}\prod_{j=1}^{k}\beta_{z(j-1)}}{\mu_z^k}e_z^{(2-k)}(t)\ )$$

- Nonlinear IMC approach gives

$$u_T(t) = \frac{m\left(\sum_{k=1}^{k=2}(-1)^{k+1}\frac{\alpha_{z(k)}\prod_{j=1}^{k}\beta_{z(j-1)}}{\mu_z^k}e_z^{(2-k)}(t) + g\right)}{\cos(\frac{1}{\mu_{\varphi}}\int_0^t e_{\varphi}(\tau)d\tau + \varphi_0)\cos(\frac{1}{\mu_{\theta}}\int_0^t e_{\theta}(\tau)d\tau + \theta_0)}$$

4.2 Lateral and longitudinal control

Once u_T is done, substituting equation (24) into system (8) we obtain

$$\begin{cases} \ddot{x}(t) = (v_z(t) + g)\left(\mathrm{c}_{\Psi}\tan(\theta) + \frac{\mathrm{s}_{\Psi}\mathrm{s}_{\varphi}}{c_{\theta}c_{\varphi}}\right) \\ \ddot{y}(t) = (v_z(t) + g)\left(\mathrm{s}_{\Psi}\tan(\theta) - \frac{\mathrm{c}_{\Psi}\mathrm{s}_{\varphi}}{c_{\theta}c_{\varphi}}\right) \end{cases} \quad (25)$$

Assumption 4: For a large enough time T, z is close to $z_r(v_z \to 0)$.

Using assumptions 3- 4, system (25) is reduced to

$$\begin{cases} \ddot{x}(t) = g(\theta\mathrm{c}_{\Psi} + \varphi\mathrm{s}_{\Psi}) \\ \ddot{y}(t) = g(\theta\mathrm{s}_{\Psi} - \varphi\mathrm{c}_{\Psi}) \end{cases} \quad (26)$$

This system is equivalent to

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} = \mathcal{R}\begin{pmatrix} \theta \\ \varphi \end{pmatrix} \quad (27)$$

Where $\mathcal{R} = g\begin{pmatrix} \mathrm{c}_{\Psi} & \mathrm{s}_{\Psi} \\ \mathrm{s}_{\Psi} & -\mathrm{c}_{\Psi} \end{pmatrix} = g\hat{\mathcal{R}}$ with $\hat{\mathcal{R}}$ an element of the special orthogonal group $SO(2)$.

Accepting that $\begin{pmatrix} \ddot{\bar{x}} \\ \ddot{\bar{y}} \end{pmatrix} = \mathcal{R}^{-1}\begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix}$, a simplified system is obtained

$$\begin{pmatrix} \ddot{\bar{x}} \\ \ddot{\bar{y}} \end{pmatrix} = \begin{pmatrix} \theta \\ \varphi \end{pmatrix} \quad (28)$$

Using system (9), we note

$$\bar{\tau}(t) = \begin{pmatrix} \overline{u_2} \\ \overline{u_3} \end{pmatrix} = \begin{cases} \dot{\theta}\dot{\Psi}\left(\frac{I_y - I_z}{I_x}\right) + \frac{\tau_{\varphi}}{I_x} \\ \dot{\varphi}\dot{\Psi}\left(\frac{I_z - I_x}{I_y}\right) + \frac{\tau_{\theta}}{I_y} \end{cases}$$

Finally

$$\begin{cases} \dddot{\bar{x}} = \dot{\theta} \\ \ddot{\theta} = \overline{u_3} \\ \dddot{\bar{y}} = \dot{\varphi} \\ \ddot{\varphi} = \overline{u_2} \end{cases} \quad (29)$$

Choosing the state vector as $(x_1, x_2, x_3, x_4)^T = \left(\bar{x}, \dot{\bar{x}}, \theta, \dot{\theta}\right)^T$, $\bar{x}$ as an output and $e_{\bar{x}} = \bar{x}_r - \bar{x}$ as the tracking error between the state $\bar{x}$ and its reference trajectory $\bar{x}_r$, we obtain for the subsystem, formed by the two first equations of system (29)

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \bar{u}_3 \end{cases} \quad (30)$$

As performed for the altitude motion, the Feedback-IMC approach gives

$$\bar{u}_3(t) = \bar{x}^{(4)}(t)$$

Using equation (14)

$$\bar{u}_3(t) = \sum_{k=1}^{k=4} (-1)^{k+1} \frac{\alpha_{\bar{x}(k)} \prod_{j=1}^{k} \beta_{\bar{x}(j-1)}}{\mu_{\bar{x}}^{k}} e_{\bar{x}}^{(4-k)}(t)$$

We proceed in a similar way, so that

$$\bar{u}_2(t) = \sum_{k=1}^{k=4} (-1)^{k+1} \frac{\alpha_{\bar{y}(k)} \prod_{j=1}^{k} \beta_{\bar{y}(j-1)}}{\mu_{\bar{y}}^{k}} e_{\bar{y}}^{(4-k)}(t)$$

Finally,

$$\tau_{\varphi}(t) = I_x \left(-\dot{\theta}\dot{\Psi}\left(\frac{I_y - I_z}{I_x}\right) + \sum_{k=1}^{k=4} (-1)^{k+1} \frac{\alpha_{\bar{y}(k)} \prod_{j=1}^{k} \beta_{\bar{y}(j-1)}}{\mu_{\bar{y}}^{k}} e_{\bar{y}}^{(4-k)}(t) \right)$$

$$\tau_{\theta}(t) = I_y \left(-\dot{\varphi}\dot{\Psi}\left(\frac{I_z - I_x}{I_y}\right) + \sum_{k=1}^{k=4} (-1)^{k+1} \frac{\alpha_{\bar{x}(k)} \prod_{j=1}^{k} \beta_{\bar{x}(j-1)}}{\mu_{\bar{x}}^{k}} e_{\bar{x}}^{(4-k)}(t) \right)$$

4.3 Yaw control

Feedback-IMC approach gives

$$\tau_{\Psi}(t) = I_z \left(\sum_{k=1}^{k=2} (-1)^{k+1} \frac{\alpha_{\Psi(k)} \prod_{j=1}^{k} \beta_{\Psi(j-1)}}{\mu_{\Psi}^{k}} e_{\Psi}^{(2-k)}(t) - \dot{\varphi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) \right)$$

$\mu_{\bar{x}}, \mu_{\bar{y}}, \mu_z, \mu_{\Psi}$ denote tuning positive parameters. The asymptotic stability of the closed loop system is trivially proved provided that $\beta_{z1} = -\mu_z^2$, $\alpha_{z1} = 1 + \mu_z^2$, $\beta_{\Psi 1} = -\mu_{\Psi}^2$ and $\alpha_{\Psi 1} = 1 + \mu_{\Psi}^2$ for the altitude and the yaw controls while the lateral and the longitudinal motions asymptotic stability is ensured by taking care that the characteristic equation $P(X) = X^4 + \sum_{k=1}^{k=4} (-1)^{k+1} \frac{\alpha_{\bar{x}(k)} \prod_{j=1}^{k} \beta_{\bar{x}(j-1)}}{\mu_{\bar{x}}^{k}} X^{(4-k)}$ has the roots in the left-half complex plane.

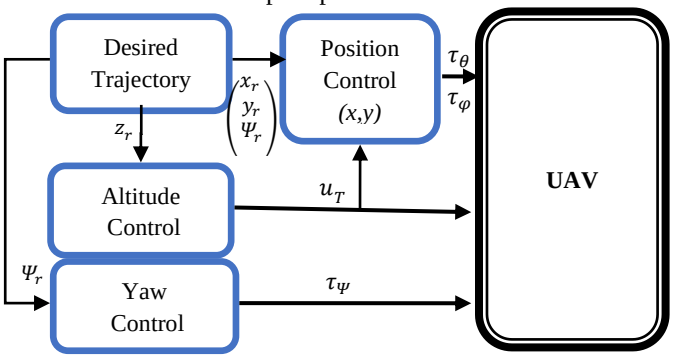

Fig. 5. Octo-Rotor control architecture.

5. EXPERIMENTAL RESULTS

A. Numerical simulation

We created a simulation environment in order to test and view the comparison of the results for the different controllers. Solving differential equations was made by Euler's method with sampling time of 0.01 seconds using the UAV parameters. The available system signals are χ, $\dot{\chi}$ and η with $(\mu_{\bar{x}}, \mu_{\bar{y}}, \mu_z, \mu_{\Psi}) = (0.05, 0.05, 0.1, 0.4)$.

As a first attempt, for the stationary flight, we take the altitude motion for a set point of 30 m. In Fig. 6, one may observe that, the system time response using the new proposed controller is faster than that using the feedback linearization controller alone.

Using a small sampling time, the main approach designed initially for constant trajectories, is extended to other trajectories types of class C^{∞}. Therefore, in this second test, we consider helix reference trajectories. The Octo-Rotor starts from the origin. The controllers have to stabilize its attitude and to reach the circular helix reference trajectory, which has radius of 10 meters (see Fig. 7).

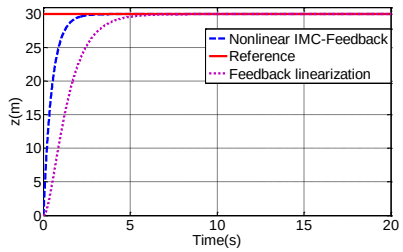

Fig. 6. Response for altitude motion in stationary flight.

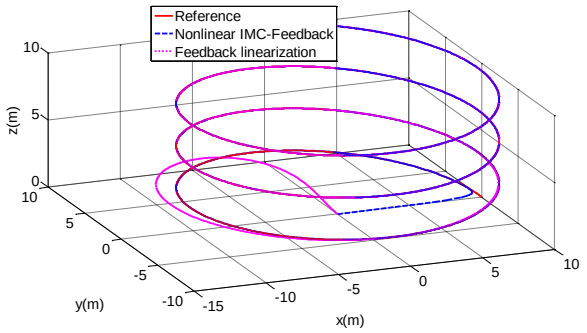

Fig. 7. Circular 3D trajectory.

In Fig. 7, the trajectory tracking results, using the Nonlinear IMC-Feedback, show that, the output rapidly converges to the desired trajectory. The results are obviously improved, compared with those obtained with the feedback linearization alone.

B. Gazebo flight simulator results

The control laws have been implemented in the Gazebo simulator. The simulator takes into consideration all the vehicle components as the payload, the dynamics of actuators and the response of sensors. Fig. 8 presents a helix trajectory which has radius of 5 meters until 1.8 meters of altitude with $(\mu_{\bar{x}}, \mu_{\bar{y}}, \mu_z, \mu_{\Psi}) = (1.7, 1.7, 0.9, 0.4)$.

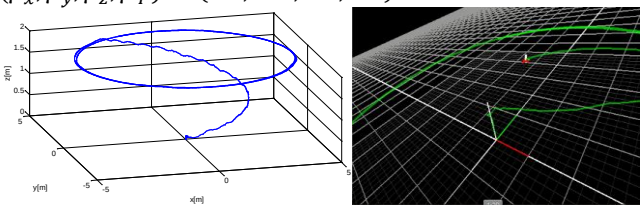

Fig. 8. Circular 3D trajectory on gazebo simulator.

In general, using numerical simulations tools, the obtained results are quite different when testing them on a more realistic simulators. However, in the present work, the obtained results are satisfactory, which demonstrate the robustness of our approach (see Fig. 8).

C. Real test implementation

To reveal the feasibility of our study, we have tested our control algorithms on a commercially AR. Drone Quadrotor, available in our laboratory. In this real test, a square trajectory of 1 meter is desired when the altitude is of 0.7 meters (see Fig. 9-10).

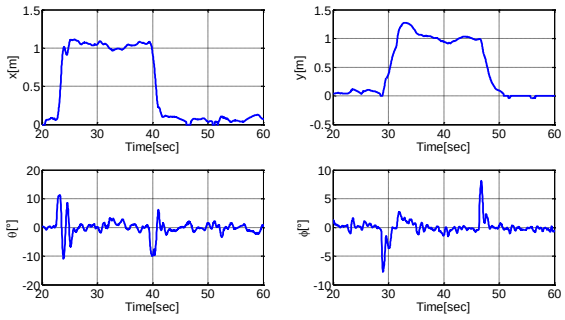

Fig. 9. Real test results of lateral and longitudinal motions.

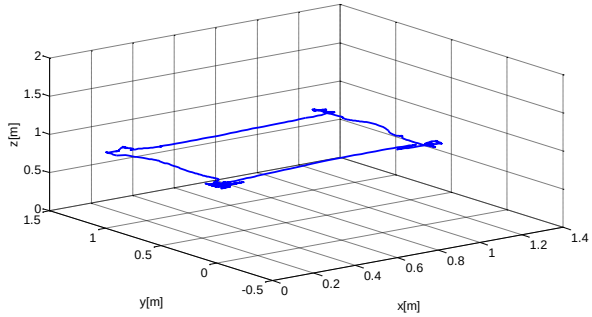

Fig. 10. 3D square trajectory real results.

The flight vehicle operates in an environment where the execution of the trajectory can be affected by atmospheric turbulence. Futheremore, some parameters of the vehicle are not well defined, or stained with uncertainties (such as aerodynamic coefficients). In this case, now, let us check the effectiveness of the control laws and their level of robustness when the system is subject to external disturbances. By introducing a disturbance term, system (11) may be written as

$$(\widetilde{\Sigma_x})\begin{cases}\dot{x} = \mathcal{F}(x) + G(x)(u + \delta(x,t)) \\ y = h(x)\end{cases} \quad (31)$$

Where $\delta(x,t): D \times [0,\infty) \to R$ is piece wise continuous, locally Lipchitz and considered as bounded by $\overline{\delta(x,t)}$. The disturbance term $G(x)\delta(x,t)$ may represent for instance: modelling errors, neglected dynamics or external disturbances. In the following example, we have applied instantaneous disturbances (3 N during the time interval [5, 8] seconds) on the altitude (see Fig. 11).

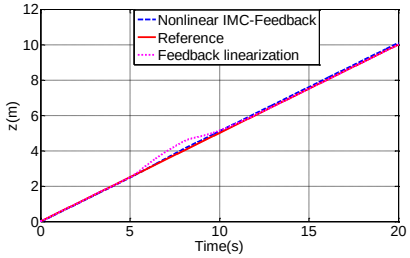

Fig. 11. System time response (instantaneous disturbances).

One may observe in Fig. 11 that the system reaches its reference trajectory when the disturbance is vanishing (after the eighth second). This explains the disturbance rejection capability of the proposed controller. However, using the feedback linearization, the system becomes unstable when the disturbance amplitude increases and goes over 22 N. Exactly, in the same condition, the system is still stable using the IMC-Feedback based one (Fig. 12).

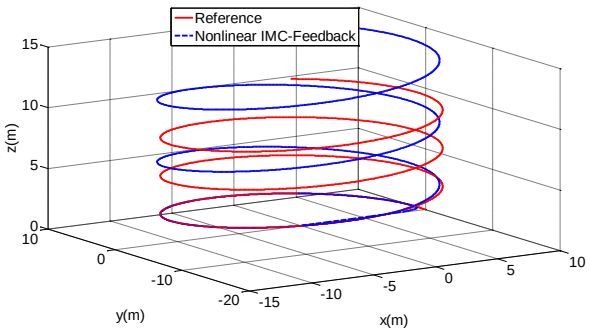

Fig. 12. 3D tracking using Nonlinear IMC-Feedback (Persistent disturbances of 23N).

6. CONCLUSION

A Square-Shaped Octo-Rotor model was described. Its modelling was simplified in order to elaborate simple control laws for a purpose of implementation. To this end, a control approach, based on a combined IMC and Feedback linearization, was proposed. Numerical simulations were performed in order to test the effectiveness of the designed control system. The performance of the designed approach was demonstrated in multiple test scenarios through a software simulator and confirmed with real tests. As matter of fact, one may guess that the proposed methodology turns out to simultaneously use the advantages of the Feedback linearization and the IMC controllers, i. e., as expected from the first one, the Octo-Rotor follows the desired trajectory with precise manner by choosing appropriate control parameters while from the second one, the disturbances are rejected and a good level of robustness is ensured.

REFERENCES

Adir, V.G., Stoica, A.M. and Whidborne, J.F. (2012). Sliding mode control of a 4y octorotor. *ISSN*, University Politehnica of Bucharest, Romania, Vol. 74, Issue. 4.

Alwi, H. (2013). LPV sliding mode fault tolerant control of an octorotor using fixed control allocation. *Control and Fault-Tolerant Systems (SysTol) Conference*, Nice, France, pp. 772-777.

Dong, J. and Brosilow, C.B. (1998). Nonlinear IMC and PID controller designs. *Prooccedings of the Amercan Control Conference*, Philadelphia, , pp. 323-327.

Economou, C.G. and Morari, M. (1986). Internal model control: extension to nonlinear systems. *Ind. Eng. Chem. Process Des. Dev.*, pp. 403-411,Philadelphia, Pennsylvania.

Isidori, A. (1985). *Nonlinear control systems*. Springer.

Khalil, H. (2002). *Nonlinear Systems.* Prentice Hall.

Morari, M. Zafriou, E. (1989). *Robust process control*. Prentice Hall.

Muhammad, D., Ahmad, Z. and Aziz, N. (2010). Implementation of internal model control (IMC) in continuous distillation column. *Proc. of the 5th International Symposium on Design, Operation and Control of Chemical Processes*, pp. 978-986.

Nitsche, R., Schwarzmann, D . and Hanschke, J. (2007). Nonlinear internal model control of Diesel air systems. *Oil & Gas Science and Technology – Rev. IFP,* Vol. 62, No. 4, pp. 501-512.

Slotine, J.-J.E. and Li, W. (1991). *Applied nonlinear control*. Prentice Hall.

Stocia, A.M. and Adir, V.G. (2012). Integral LQR control of a star-shaped octorotor. *ISSN*, Volume 4, Issue 2, pp. 3-18.