



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

UAV robust attitude hybrid controller

LAUREA MAGISTRALE IN AERONAUTICAL ENGINEERING - INGEGNERIA AERONAUTICA

Author: FEDERICO CAVAGNINI

Advisor: PROF. M. LOVERA

Co-advisors: PROF. D. INVERNIZZI, ENG. G. GOZZINI

Academic year: 2020-2021

1. Introduction

The objective of this thesis is to investigate the problems related to the attitude parametrization by use of unit-quaternions and to implement a UAV attitude controller robust to measurement noise. The new regulator implements a hybrid and hysteretic logic which allows the controller to preserve its global attractivity, unlike the case of discontinuous controller which simply implements a control law relying on the sign of the quaternion error scalar component. Finally, the control architecture is implemented onboard a drone produced by ANT-X which is simulated for various flight conditions and experimentally tested in the Flying Arena for Rotorcraft Technologies (Fly-ART) of the Aerospace Systems and Control Laboratory (ASCL) of Politecnico di Milano.

2. Problem formulation

2.1. Attitude kinematics, dynamics and stabilization

The attitude of a rigid body is represented by a 3×3 rotation matrix R , describing the rotation between two reference frames. The set of 3×3 orthogonal rotation matrices with positive determinant is the special orthogonal group of

order three,

$$SO(3) = \left\{ R \in \mathbb{R}^{3 \times 3} : R^T R = I, \det R = 1 \right\}, \quad (1)$$

where $I \in \mathbb{R}^{3 \times 3}$ denotes the identity matrix. A quadrotor UAV can be described as a rigid body of constant mass m and constant inertia matrix $J \in \mathbb{R}^{3 \times 3}$. Let $R \in SO(3)$ represent the rotation of a vector from the body frame to the inertial frame, and $\omega_b \in \mathbb{R}^3$ the UAV angular velocity in the body frame. The attitude kinematics of a rigid body are expressed as:

$$\dot{R} = RS(\omega), \quad R \in SO(3), \quad (2)$$

where the skew operator $S(\omega)$ is defined as

$$S(\omega) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (3)$$

Under the rigid body assumption, the angular velocity dynamics are described as:

$$J\dot{\omega} = S(J\omega)\omega + \tau, \quad (4)$$

where τ represents the control torque vector. The control torque term τ is the result of the

angular moments along the three body axis generated by the four rotor producing different thrusts.

Unit quaternions

$$q = [\eta \ \epsilon^\top]^\top = \begin{bmatrix} \cos(\frac{\phi}{2}) \\ e_1 \sin(\frac{\phi}{2}) \\ e_2 \sin(\frac{\phi}{2}) \\ e_3 \sin(\frac{\phi}{2}) \end{bmatrix}, \quad \|q\| = 1, \quad (5)$$

are often used to parameterize rotation matrices of the special orthogonal group $\text{SO}(3)$, where $\eta \in \mathbb{R}$ denotes the scalar component of the quaternion and $\epsilon \in \mathbb{R}^{3 \times 3}$ the vector component. The unit quaternion can be transformed in a rotation matrix through the Rodrigues formula defined through the map $\mathcal{R} : \mathbb{S}^3 \rightarrow \text{SO}(3)$ as

$$\mathcal{R}(q) = I + 2\eta S(\epsilon) + 2S^2(\epsilon), \quad (6)$$

where the $S(\cdot)$ operator is defined as in equation (3) and $I \in \mathbb{R}^{3 \times 3}$ denotes the identity matrix. When dealing with quaternions, equation (2) involves time derivatives of quaternions and becomes

$$\dot{q} = \frac{1}{2}q \otimes \nu(\omega) = \frac{1}{2}q \otimes [0 \ \omega^\top]^\top, \quad (7)$$

where (\otimes) indicates the quaternion product operator. From equation (6), it results that there are exactly two unit quaternions, $\pm q$, representing the same physical attitude, namely $R = \mathcal{R}(q) = \mathcal{R}(-q)$. The ambiguity arising from the quaternion representation can cause inconsistent quaternion-based controllers to unnecessarily rotate the rigid body through a full rotation, leading to the so called *unwinding* phenomenon. This behavior can be induced by inconsistent control laws that are designed to stabilize a single point in \mathbb{S}^3 , while leaving the antipodal point unstable, despite the fact that they both correspond to the same physical orientation.

Hence, stabilizing a single attitude in $\text{SO}(3)$ requires stabilizing a disconnected two-point set in the quaternion space and selecting which quaternion to use for control. Let $R_d \in \text{SO}(3)$ denotes a constant desired reference attitude and R the attitude of the rigid body. The error coordinate can be expressed as

$$R_e = R_d^\top R. \quad (8)$$

The corresponding kinematic equation is

$$\dot{R}_e = R_e S(\omega_b). \quad (9)$$

When dealing with quaternions, the attitude error is computed as a quaternion error

$$q_e = q_{sp}^{-1} \otimes q. \quad (10)$$

The attitude control objective is to design a control torque that globally asymptotically stabilizes R_e to the identity matrix. When dealing with quaternions, the unit quaternion associated to $R_e = I$ is the identity element $q_e = \pm \mathbf{1} = [1 \ 0_{1 \times 3}^\top]^\top$. In terms of attitude dynamics, the control requirement is to robustly and globally asymptotically stabilize the set

$$\mathcal{A}_d = \{(q, \omega) \in \mathbb{S}^3 \times \mathbb{R}^3 : q_e = \pm \mathbf{1}, \omega = 0\}. \quad (11)$$

2.2. Noise effect on attitude control

As discussed in [4], robustly and globally stabilizing the disconnected set of points in quaternions space is not achievable with non-hybrid (discontinuous) state feedback controllers when measurement noise is present. To show this fact, consider the Lyapunov function

$$\bar{V}(q) = 2(1 - \eta), \quad (12)$$

which satisfies that $\bar{V}(\mathbb{S}^3 \setminus \mathbf{1}) > 0$ and $\bar{V}(\mathbf{1}) = 0$. When the feedback law $\omega = -\epsilon$ is applied, the gradient of the Lyapunov function is computed as

$$\left\langle \nabla \bar{V}(q), \frac{1}{2}q \otimes \nu(-\epsilon) \right\rangle = -\|\epsilon\|_2^2. \quad (13)$$

This particular choice of feedback law generates two equilibrium points $\pm \mathbf{1}$; in particular, $-\mathbf{1}$ is unstable and $+\mathbf{1}$ is stable. Note that, as discussed in Section 2.1, the two points $\pm \mathbf{1}$ represent the same point (same attitude) in $\text{SO}(3)$: therefore, the desired attitude can be either stable or unstable.

To remedy this issue, some authors [1] proposed stabilizing the set $\{\pm \mathbf{1}\}$ with discontinuous control defining the sign function

$$\text{sgn}(\eta) = \begin{cases} -1 & \eta < 0 \\ 1 & \eta \geq 0. \end{cases} \quad (14)$$

It results that the feedback law $\omega = -\text{sgn}(\eta)\epsilon$ is globally asymptotically stabilizing with Lyapunov function $\bar{V}_d = 2(1 - |\eta|)$. However, the global attractivity property is not robust to arbitrary small measurement noise and, as discussed in [4], there exists an arbitrarily small piecewise-constant noise signal that, for initial conditions arbitrarily close to the discontinuity, keeps the state near the discontinuity. Figure 1 illustrates the unwinding produced by continuous control, as well as the non-robust global asymptotic stability produced by discontinuous control. Note that the discontinuity lies at $\eta = 0$, which corresponds to attitudes that are a 180° rotation from the desired equilibrium. This discontinu-

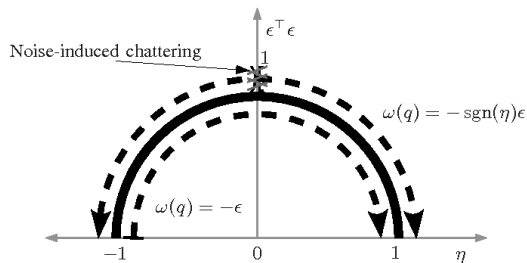


Figure 1: Quaternion-based attitude control: non-robust global asymptotic stability. Arrows indicate the direction of rotation. (from [3])

ous strategy is the logic on which relies the attitude controller already implemented onboard the drone produced by ANT-X and for this reason called "baseline" in the context of this thesis.

3. Hybrid solutions

3.1. Hybrid systems preliminaries

A hybrid system is one where both continuous and discrete evolution of the state $x \in \mathbb{R}^n$ are possible. As described in [2], a hybrid system \mathcal{H} is defined as

$$\mathcal{H} = \begin{cases} \dot{x} \in F(x) & x \in C \\ x^+ \in G(x) & x \in D, \end{cases} \quad (15)$$

where C denotes the flow set, $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ the flow map, D the jump set, $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ the jump map. Sets and maps describe the continuous- and discrete-time dynamics and the regions on which these dynamics apply.

3.2. Attitude stabilization

C. G. Mayhew et al. [4] proposed a quaternion-based hybrid feedback for the attitude stabiliza-

tion of a rigid body adopting the strategy illustrated in Figure 2. The attitude controller implements a dynamic feedback that exploits a memory state to select which pole of \mathbb{S}^3 to regulate in a hysteretic fashion. Let $\delta \in (0, 1)$ denote

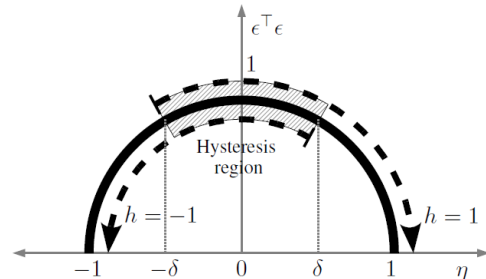


Figure 2: Hysteretic regulation of unit quaternions to the set $\{\pm 1\}$ (from [3]).

the hysteresis half-width and $h \in H := \{-1, 1\}$ a logic variable that selects the desired rotation direction to move q to either $+1$ or -1 . The feedback law $\omega = -h\epsilon$ proposed in [4] is based the following dynamics of h :

$$\begin{cases} \dot{h} = 0 & (q, h) \in \{h\eta \geq -\delta\} \\ h^+ = -h & (q, h) \in \{h\eta < -\delta\}. \end{cases} \quad (16)$$

The inequalities defining the flow set and the jump set are designed to detect a sign mismatch, which indicates whether the feedback $\omega = -h\epsilon$ is pulling in the direction of the shortest rotation to align q with ± 1 ($h\eta \geq 0$) or a longer rotation ($h\eta < 0$).

Hence, the parameter δ manages a trade-off between robustness to measurement noise and a small amount of unwinding: when $\delta \geq 1$, the value of the logic variable h cannot change and the strategy reduces to a static feedback that induces unwinding; when $\delta = 0$, the resulting control becomes discontinuous and therefore not robust to attitude noise. When the above hybrid feedback is implemented for the stabilization of *attitude dynamics*, the closed-loop system becomes

$$\left. \begin{aligned} \dot{q} &= \frac{1}{2}q \otimes \nu(\omega) \\ \dot{\omega} &= J^{-1}(S(J\omega)\omega + \tau) \\ \dot{h} &= 0 \end{aligned} \right\} (q, \omega, h) \in C, \quad (17)$$

$$\left. \begin{aligned} q^+ &= q \\ \omega^+ &= \omega \\ h^+ &= -h \end{aligned} \right\} (q, \omega, h) \in D,$$

where the flow set and the jump set are defined as

$$\begin{aligned} C &= \{(q, \omega, h) \in \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta \geq -\delta\} \\ D &= \{(q, \omega, h) \in \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta < -\delta\}, \end{aligned} \quad (18)$$

and the control objective requires to stabilize the set \mathcal{A}_d of equation (11). C. G. Mayhew [4] proposed an energy-based Lyapunov function for the stability analysis of the closed loop system described by equation (17). Let $c > 0$ and

$$V_d(x) = cV(x) + \frac{1}{2}\omega^T J\omega, \quad (19)$$

where $x = (q, h, \omega)$, such that $V_d((\mathbb{S}^3 \times \mathbb{R}^3 \times H) \setminus \mathcal{A}_d) > 0$ and $V_d(\mathcal{A}_d) = 0$. Assume a control torque τ of the form

$$\tau = -ch\epsilon - K_\omega\omega, \quad (20)$$

where $K_\omega = K_\omega^T > 0$. The change in V_d along flows is computed as

$$\langle \nabla V_d(x), f(x, \tau(x)) \rangle = -\omega^T K_\omega\omega. \quad (21)$$

The change of V_d over jumps is computed as

$$V_d(g(x)) - V_d(x) = 4ch\eta. \quad (22)$$

Under the conditions expressed by the flow set and the jump set of equation (18), it results that

$$\begin{aligned} \langle \nabla V_d(x), f(x, \tau(x)) \rangle &\leq 0 \text{ for all } x \in C \\ V_d(g(x)) - V_d(x) &< 0 \text{ for all } x \in D. \end{aligned} \quad (23)$$

Hence, the energy-based Lyapunov function V_d is strictly decreasing along all trajectories, except along those starting from the set $\{q = h\mathbf{1}\}$ and with $\omega = 0$: the set \mathcal{A}_d is globally asymptotically stable for the closed-loop system described in equation (17).

3.3. Attitude tracking

J. Su et al. [5] analyzed the problem of hybrid control for robust global *attitude tracking*. In particular, Su proposed a Proportional-Integral-Derivative-like control law that prevents the singular orientation and the unwinding phenomenon discussed in Section 2.1, and achieves robust global asymptotic tracking. Let $q_d = [\eta_d \ \epsilon_d]$ be the unit quaternion representing the

desired attitude, and $\omega_d \in \mathbb{R}^3$ be the desired angular velocity. It is a reasonable hypothesis to assume that both ω_d and $\dot{\omega}_d$ are bounded so that the reference trajectory is bounded and well defined. The desired attitude motion evolves as

$$\dot{q}_d = \frac{1}{2}q_d \otimes \nu(\omega_d). \quad (24)$$

The attitude error, defined as a quaternion error, and the angular velocity error are computed as

$$\begin{aligned} q_e &= q_d^{-1} \otimes q = [\eta_e \ \epsilon_e] \\ \omega_e &= \omega - \mathcal{R}(q_e)^T \omega_d, \end{aligned} \quad (25)$$

where the operator $\mathcal{R}(q_e)$ is defined by equation (6) and (q, ω) represents the actual state. The control objective is to design τ so that (q, ω) asymptotically tracks a desired bounded reference trajectory for any initial state satisfying $q_e(0) \in \mathbb{S}^3$ and $\omega_e(0) \in \mathbb{R}^3$, namely the set $\mathcal{A}_t = \{(q_e, \omega_e) \in \mathbb{S}^3 \times \mathbb{R}^3 : q_e = \pm\mathbf{1}, \omega_e = 0\}$ must be made asymptotically stable. The error dynamics is described as

$$\begin{aligned} \dot{q}_e &= \frac{1}{2}q_e \times \nu(\omega_e) \\ J\dot{\omega}_e &= \Xi(\omega_e, \omega_d) + \tau, \end{aligned} \quad (26)$$

where J denotes the inertia matrix and $\Xi(\omega_e, \omega_d) = -S(\omega_e + \mathcal{R}(q_e)\omega_d)J(\omega_e + \mathcal{R}(q_e)\omega_d) + J(S(\omega_e)\mathcal{R}(q_e)\omega_d - \mathcal{R}(q_e)\dot{\omega}_d)$. The tracking control law is designed as $\tau = -\Xi(\omega_e, \omega_d) - h(JC_i + c_p I^{3 \times 3})\epsilon_e - C_d\omega_e - C_d C_i I_e$, where $0 < c_p \in \mathbb{R}$, $0 < C_i \in \mathbb{R}^{3 \times 3}$, $0 < C_d \in \mathbb{R}^{3 \times 3}$ are constants, $I_e = \int_0^t h\epsilon_e dt$ represents the integral action, and h a logic variable. The resulting hybrid closed-loop system is governed by flow and jump set

$$\begin{aligned} C &= \{x \in \mathbb{R}^3 \times \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta \geq -\delta\} \\ D &= \{x \in \mathbb{R}^3 \times \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta < -\delta\}, \end{aligned} \quad (27)$$

where $x = (I_e, q_e, \omega_e, h)$. The global asymptotic stability of the set \mathcal{A}_t is demonstrated in [5] with the analysis of a Lyapunov function.

4. Simulation result

Hybrid control has been introduced for its robustness to measurement noise. In order to verify this property, a simulation involving only the identified attitude dynamics of the UAV and

	K_P	K_I	K_D
Roll	0.08	0.05	0.0015
Pitch	0.08	0.05	0.0015
Yaw	0.2	0.1	0

Table 1: PID controller parameters.

measurement noise on quaternion has been performed; in particular, the controller is subjected to a measurement noise so that the measured state satisfies $\tilde{q} = (q + e)/\|q + e\|$ where e is selected randomly from a uniform distribution with mean $\mu = 0$ and variance $\text{Var} = \sigma^2$, $\sigma = 0.0349$ representing the attitude noise standard deviation. This simulation shows the benefits of using hysteresis over discontinuous control when measurement noise is present. Consider a hybrid controller with feedback law

$$\omega_o = 2 \begin{bmatrix} K_P^{ROLL} & 0 & 0 \\ 0 & K_P^{PITCH} & 0 \\ 0 & 0 & K_P^{YAW} \end{bmatrix} h\epsilon, \quad (28)$$

and with control parameters $\delta = 0.11$, $h(0) = 1$. For both the hybrid and baseline strategy, $K_P^{ROLL} = 10$, $K_P^{PITCH} = 10$ and $K_P^{YAW} = 2.8$; an angular rate saturation limit of $\pm 10 \text{ rad/s}$ is imposed on all the three components p, q, r . Table 1 reports the parameters for each PID of the angular rate regulator; note that the PIDs for roll and pitch motion are characterized by the same parameters due to the geometry of the quadrotor. Control moments saturation limits of ± 1 are present since normalized control actions are considered. This simulation shows the response for a set-point of 180° in yaw, with null roll and pitch angles: the initial conditions are set to be $(0, 0, 0)^\circ$ and $\omega(0) = 0$, the desired attitude $(0, 0, -180)^\circ$. Figure 3 shows that the hybrid controller is impervious to the noise and outperforms the discontinuous strategy, which exhibits a lag in the response. This is explained by checking the yaw control moment presented in Figure 4, where for the baseline controller a noise-induced chattering forces the UAV to initially oscillate around the initial condition by continuously applying positive and negative control action. The hybrid controller is impervious to measurement noise thanks to the hysteresis variable δ and does not exhibit any chattering.

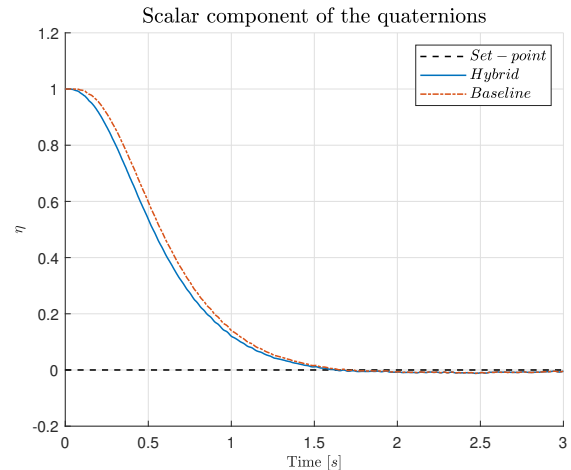


Figure 3: Identified dynamics attitude stabilization in presence of measurement noise: scalar components of the quaternions.

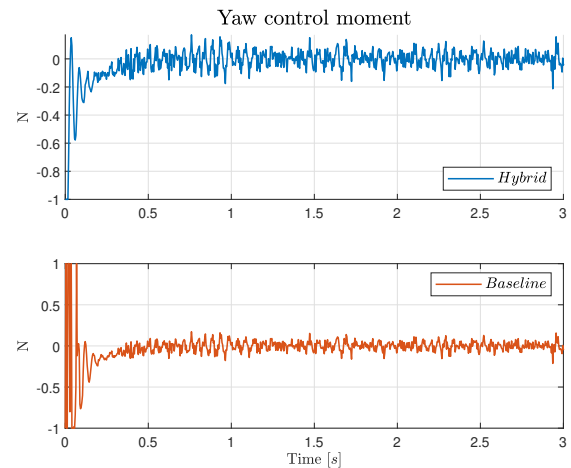


Figure 4: Identified dynamics attitude stabilization in presence of measurement noise: yaw control moments.

5. Experimental result

The implemented control laws have been validated experimentally inside the Flying Arena for Rotorcraft Technologies (Fly-ART) of the Aerospace Systems and Control Laboratory. In the following, a direct comparison of the three controller is presented with respect to a circular trajectory characterized by a radius $R = 1 \text{ m}$ and a frequency $\omega = 1 \text{ rad/s}$, so that the North and East position components are computed as $R \cos(\omega t)$ and $R \sin(\omega t)$, respectively. Consider as initial conditions the position $(1, 0, -1.5) \text{ m}$ expressed in the NED reference frame, null attitude, and null angular rates. Af-

ter one circle and a half, the drone has to reach the final conditions set to be the center of the circle, namely the position $(0, 0, -1.5) m$, and null attitude. For the sake of simplicity, the related figures focus only on the pure circular motion without considering the final position set-point for the center of the circle. Figures 5 and 6 show the comparison in terms of North and East position and position errors, respectively: baseline and hybrid kinematic controllers exhibit the same behavior, whereas the hybrid PID controller introduces delay in the response. Figure 7 illustrates the attitude comparison in terms of roll and pitch angles: once again, baseline and hybrid kinematic controllers are comparable, whereas the hybrid PID controller exhibits a more aggressive behavior that prevents the correct attitude tracking causing the error in position. Note that, unlike the case of the position for which the set-point is the same for all the three controllers, the desired attitude is different for each controller and therefore a single reference cannot be considered. Figure 8 shows the attitude errors.

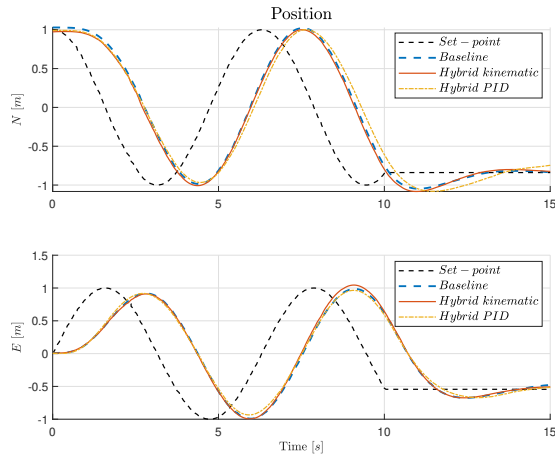


Figure 5: Experimental circular trajectory. Comparison of the three controllers. NE position.

6. Conclusions

The aim of this thesis was to investigate the problems related to the attitude parametrization by use of unit-quaternions and to implement an attitude controller robust to measurement noise for a UAV. The identified dynamics of the drone have been simulated in the context of a set-point of 180° in yaw in presence of measurement noise

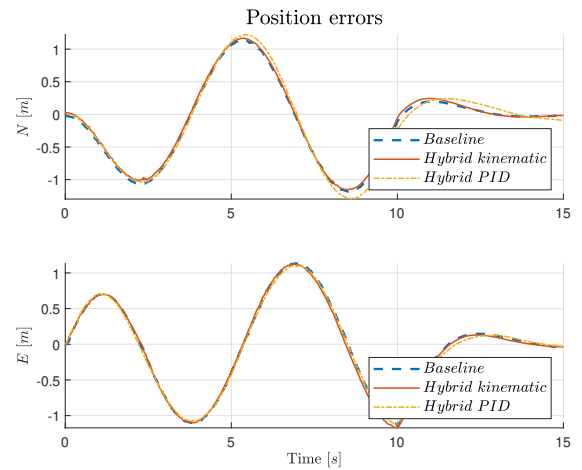


Figure 6: Experimental circular trajectory. Comparison of the three controllers. NE position errors.

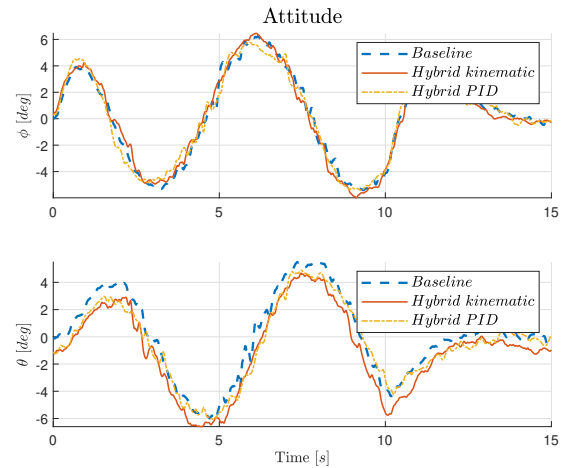


Figure 7: Experimental circular trajectory. Comparison of the three controllers. Attitude.

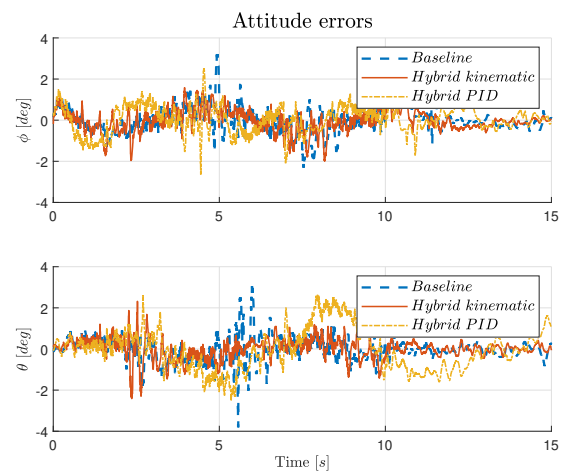


Figure 8: Experimental circular trajectory. Comparison of the three controllers. Attitude errors.

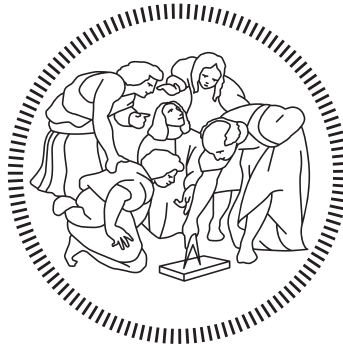
considering two different control strategies. The first control strategy implements discontinuous control law relying on the sign of the quaternion error scalar component; in the context of this thesis, this controller is called "baseline" since it is the control law already present on the ANT-X quadrotor. The second control strategy implements a hybrid and hysteretic logic which allows the controller to preserve the global attractivity. The results of the simulation showed that the hybrid kinematic controller is impervious to the noise and outperforms the discontinuous strategy, which exhibited a lag in the response due to a noise-induced chattering that forced the UAV to initially oscillate around the initial condition by continuously applying positive and negative control actions. In the context of attitude tracking, a second hybrid regulator has been introduced with a more complex control architecture, as described by J. Su [5]: unlike the hybrid kinematic controller, this regulator implements a feed-forward term proportional to the desired angular velocity to improve tracking performance.

Finally, the three controllers have been experimentally tested and validated. The results showed that the hybrid kinematic controller resembled the performance of the baseline regulator, however the hybrid logic improves its performances for its robustness to measurement noise. The hybrid PID controller resulted less performing than the other two due to a non-optimal tuning which was not possible to achieve in the context of this thesis.

References

- [1] D. Fragopoulos and M. Innocenti. Stability considerations in quaternion attitude control using discontinuous Lyapunov functions. *IEE Proceedings - Control Theory and Applications*, 151(3):253–258, 2004.
- [2] R. Goebel, R. G. Sanfelice, and A. R. Teel. Hybrid dynamical systems: Robust stability and control for systems that combine continuous-time and discrete-time dynamics. *IEEE Control Systems Magazine*, 29(2):28–93, 2009.
- [3] C. G. Mayhew. *Hybrid Control for Topologically Constrained Systems*. PhD thesis, University of California, December 2010.
- [4] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Robust global asymptotic attitude stabilization of a rigid body by quaternion-based hybrid feedback. In *Proceedings of the 2009 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 2522–2527, 2009.
- [5] J. Su and K. Cai. Globally stabilizing proportional-integral-derivative control laws for rigid-body attitude tracking. *Journal of Guidance, Control, and Dynamics*, 34(4):1260–1264, July-August 2011.

POLITECNICO DI MILANO
School of Industrial and Information Engineering
Master of Science in Aeronautical Engineering



UAV robust attitude hybrid controller

Advisor: Prof. Marco LOVERA
Co-Advisor: Prof. Davide INVERNIZZI
Eng. Giovanni GOZZINI

Thesis by:
Federico CAVAGNINI Matr. 944873

Academic Year 2020–2021

A mia Madre e a mio Padre

Acknowledgments

Set your intentions

Dream with care

Tomorrow's a new day for everyone

Brand new Moon, brand new Sun

Xavier Rudd, *Follow The Sun*

Finalmente giunto al termine di questo percorso, penso sia doveroso fare alcuni ringraziamenti.

Il primo ringraziamento è per il Professor Marco Lovera che mi ha accompagnato con i suoi corsi per tutto il percorso della Laurea Magistrale e che mi ha permesso di entrare a far parte del suo team per questo progetto di tesi.

Un grosso grazie va anche al Professor Davide Invernizzi per aver messo a disposizione le sue abilità e capacità con consigli e suggerimenti.

Un ringraziamento speciale a Giovanni per avermi pazientemente ed instancabilmente seguito durante tutta l'attività di tesi mostrandosi sempre pronto e disponibile a rispondere alle mie chiamate per aiutarmi e rassicurarmi.

Un ringraziamento per gli amici che, in un modo o nell'altro, mi sono stati vicino. A special thank to my friend Mateusz: we lived this adventure together through every lecture and project; I'll never forget our endless work sessions and the talks about football on our Monday meetings.

Un grazie a Simona (*Simo-Simo*) per la complicità e per il supporto che mi ha sempre mostrato.

Un grazie di cuore a Barbara, *la simpatica della coppia*, per avermi sempre supportato e soprattutto sopportato, per avermi regalato gioia e serenità standomi accanto anche nei momenti più difficili e di bisogno: è stato più facile superare questi ostacoli con te al mio fianco. Desidero ringraziare anche Antonella e Giorgio per il supporto e per avermi fatto sempre sentire parte della loro famiglia.

Infine un ringraziamento enorme alla mia famiglia per l'affetto e il sostegno dimostratomi. Alle nonne Vincenza e Ida e al nonno Emanuele che hanno sempre creduto in me; al nonno Angelo che, anche se non più tra noi, sono sicuro mi abbia sempre seguito e sarebbe stato fiero e felice di festeggiare questo traguardo.

A mia sorella Valentina con la sua dolcezza, ad Emanuele, Dafne e Aurora che mi

hanno sempre sostenuto ed incoraggiato nonostante tutto.

A Mamma Ines e a Papà Emilio che mi hanno permesso di studiare con tranquillità e serenità e che mi sono sempre stati vicini in ogni momento, anche nei periodi in cui ero "malmostoso" (cit. Mamma); mi avete sempre incoraggiato ed esortato a non mollare mai, a non accontentarmi: questi i principi cardine su cui avete basato la vostra vita e che hanno permesso, con grande sacrificio, di raggiungere ogni obiettivo.

Federico Cavagnini
21 Dicembre 2021

Abstract

Nowadays, unmanned aerial vehicles (UAVs) represent a continuously expanding research field thanks to their numerous applications. Not only for military and professional, but also for recreational purposes, UAVs are winning the worldwide market with a variety of projects and different control strategies, as well as avionics architectures and sensors onboard the drone. In particular, the interaction between measurement noise and attitude control strategies may lead to undesired behaviors of the drone whenever the attitude regulator is not correctly designed. The purpose of this thesis is to investigate the effect of the quaternions double coverage, that is there are exactly two unit quaternions representing each element in $SO(3)$, hence the need to stabilize a disconnected, two-point set in order to stabilize the desired attitude. Measurement noise is the natural consequence of using sensors: when combined with a discontinuous attitude control law, that is a quaternion-based law relying on the sign of the quaternion error scalar component, measurement noise can destroy global attractivity of the controller. Hybrid control laws are proposed as solutions and implemented and simulated for various flight conditions. Eventually, the same control strategies are successfully tested and validated through the experimental activity in the Flying Arena for Rotorcraft Technologies (Fly-ART) arena of Aerospace Systems and Control Laboratory (ASCL) of Politecnico di Milano.

Sommario

Oggigiorno, i velivoli a pilotaggio remoto sono oggetto di ricerca in continuo sviluppo grazie alle loro numerose applicazioni. Non solo militari e professionali, ma anche ricreativi, gli UAV stanno diventando sempre più diffusi sul mercato, dando vita ad una varietà che si riflette inevitabilmente su diverse strategie di controllo, nonché di avionica e sensori di bordo. In particolare, l'interazione tra misure con rumore e strategia di controllo di assetto può portare ad un comportamento indesiderato da parte del drone quando il controllore risulti essere non opportunamente progettato.

Lo scopo di questa tesi è di investigare l'effetto secondo cui i quaternioni forniscono una doppia copertura del gruppo di rotazione $SO(3)$, cioè lo stesso assetto è descritto da due quaternioni diversi, e a causa del quale, per stabilizzare un determinato assetto, è necessario stabilizzare un insieme disconnesso di due punti (quaternioni). Come anticipato, naturale conseguenza dell'utilizzo di sensori è il rumore di misura che, combinato con una legge di controllo di tipo discontinuo, perché basata sul segno della parte scalare del quaternionione errore, può ledere l'efficacia del controllore e della proprietà di attrattività globale.

Le leggi di controllo ibride vengono proposte come soluzione a tale problematica e sono implementate e simulate in diverse condizioni di volo. Infine, le stesse strategie di controllo sono state testate e validate con successo attraverso l'attività sperimentale condotta nell'arena FlyArt del Laboratorio di Controllo e Sistemi Aerospaziali (ASCL) del Politecnico di Milano.

Contents

Acknowledgments	I
Abstract	III
Sommario	V
List of figures	IX
List of tables	XIII
Introduction	1
1 Problem formulation	5
1.1 Definition of reference frames	5
1.1.1 NED frame	5
1.1.2 Body frame	5
1.2 Rotation of reference frames	6
1.2.1 Direction cosine matrix	6
1.2.2 Euler Angles	6
1.2.3 Quaternions	8
1.3 Attitude dynamics of a quadrotor	9
1.4 Quaternions and attitude stabilization	11
1.4.1 Pitfalls of attitude stabilization	11
1.4.2 Noise effect on attitude control	12
1.4.3 ANT-X baseline attitude controller	12
2 Attitude hybrid controller	15
2.1 Modeling of hybrid systems	15
2.1.1 Hybrid systems preliminaries	15
2.1.2 Hybrid controllers background	16
2.1.3 Hybrid control problem	17
2.2 Attitude stabilization	18
2.3 Attitude tracking	20
2.4 ANT-X attitude hybrid controller	22

2.4.1	ANT-X attitude stabilization	22
2.4.2	ANT-X attitude tracking	23
3	Simulation results	27
3.1	Attitude stabilization simulation	27
3.1.1	Attitude stabilization case from the literature	27
3.1.2	Identified dynamics attitude stabilization	29
3.2	Attitude tracking simulation	33
3.2.1	Attitude tracking case from the literature	33
3.2.2	Identified dynamics attitude tracking	33
3.3	ANT-X simulator	39
3.3.1	Hybrid kinematic controller	39
3.3.2	Hybrid PID controller	47
4	Experimental Results	55
4.1	System architecture	55
4.1.1	Flight Control Unit	55
4.1.2	Companion computer	55
4.1.3	Drone	56
4.1.4	Motion Capture system (Mo-Cap)	56
4.1.5	Ground Control Station	58
4.2	Flight test results	58
4.2.1	Discontinuous controller	58
4.2.2	Hybrid kinematic controller	65
4.2.3	Hybrid PID controller	71
4.3	Comparison of circular trajectories	77
	Conclusions	81

List of Figures

1.1	Euler angles rotations (from [1])	7
1.2	Quadcopter scheme ([2])	10
1.3	Quaternion-based attitude control: non-robust global asymptotic stability. Arrows indicate the direction of rotation. (from [3]) . . .	13
1.4	ANT-X attitude baseline controller	13
2.1	Solution to a hybrid system (from [4]).	16
2.2	Hysteretic regulation of unit quaternions to the set $\{\pm\mathbf{1}\}$ (from [3]).	18
2.3	Quaternions kinematics hybrid controller.	22
2.4	ANT-X identified attitude dynamics hybrid controller.	23
2.5	ANT-X identified attitude dynamics and hybrid controller for attitude tracking.	23
2.6	ANT-X trajectory filter.	25
3.1	Simulation from the paper of C. G. Mayhew et al. [5]: quaternions components.	28
3.2	Simulation from the paper of C. G. Mayhew et al. [5]: logic variable h	28
3.3	Simulation from the paper of C. G. Mayhew et al. [5]: second power of the norm of ω	29
3.4	Identified dynamics attitude stabilization in presence of measurement noise: scalar components of the quaternions	30
3.5	Identified dynamics attitude stabilization in presence of measurement noise: detail of the delay of the quaternions scalar components.	31
3.6	Identified dynamics attitude stabilization in presence of measurement noise: scalar component of the quaternion errors.	31
3.7	Identified dynamics attitude stabilization in presence of measurement noise: yaw rate.	32
3.8	Identified dynamics attitude stabilization in presence of measurement noise: yaw control moments.	32
3.9	Simulation from the paper of J. Su [6]: quaternions components.	34
3.10	Simulation from the paper of J. Su [6]: angular rates.	34
3.11	Identified dynamics attitude tracking: Euler angles.	35

3.12	Identified dynamics attitude tracking: quaternions components.	36
3.13	Identified dynamics attitude tracking: quaternion error scalar components.	36
3.14	Identified dynamics attitude tracking: angular rates.	37
3.15	Identified dynamics attitude tracking: angular rates detail of the first two seconds of simulation.	37
3.16	Identified dynamics attitude tracking: angular rates errors.	38
3.17	Identified dynamics attitude tracking: control moments.	38
3.18	Identified dynamics attitude tracking: control moments detail of the first two seconds of simulation.	39
3.19	ANT-X simulator with hybrid kinematic controller. Position set-point: East position.	40
3.20	ANT-X simulator with hybrid kinematic controller. Position set-point: East position errors.	41
3.21	ANT-X simulator with hybrid kinematic controller. Position: roll attitude.	41
3.22	ANT-X simulator with hybrid kinematic controller. Position set-point: roll attitude error.	42
3.23	ANT-X simulator with hybrid kinematic controller. Attitude set-point: yaw attitude.	43
3.24	ANT-X simulator with hybrid kinematic controller. Attitude set-point: yaw attitude error.	43
3.25	ANT-X simulator with hybrid kinematic controller. Circular trajectory: NE position.	44
3.26	ANT-X simulator with hybrid kinematic controller. Circular trajectory: NE position errors.	45
3.27	ANT-X simulator with hybrid kinematic controller. Circular trajectory.	45
3.28	ANT-X simulator with hybrid kinematic controller. Circular trajectory: attitude.	46
3.29	ANT-X simulator with hybrid kinematic controller. Circular trajectory: attitude errors.	46
3.30	ANT-X simulator with Hybrid PID controller. Position set-point: East position.	47
3.31	ANT-X simulator with hybrid PID controller. Position set-point: East position error.	48
3.32	ANT-X simulator with hybrid PID controller. Position set-point: roll attitude.	48
3.33	ANT-X simulator with hybrid PID controller. Position set-point: roll attitude error.	49
3.34	ANT-X simulator with hybrid PID controller. Attitude set-point: yaw attitude.	50

3.35	ANT-X simulator with hybrid PID controller. Attitude set-point: yaw attitude error.	50
3.36	ANT-X simulator with hybrid PID controller. Circular trajectory: NE position.	51
3.37	ANT-X simulator with hybrid PID controller. Circular trajectory: NE position errors.	52
3.38	ANT-X simulator with hybrid PID controller. Circular trajectory.	52
3.39	ANT-X simulator with hybrid PID controller. Circular trajectory: attitude.	53
3.40	ANT-X simulator with hybrid PID controller. Circular trajectory: attitude errors.	53
4.1	ANT-X.	57
4.2	Optitrack IR camera and IR markers.	57
4.3	Experimental position set-point. Discontinuous controller. East position.	59
4.4	Experimental position set-point. Discontinuous controller. East position errors.	59
4.5	Experimental position set-point. Discontinuous controller. Attitude.	60
4.6	Experimental position set-point. Discontinuous controller. Attitude errors.	60
4.7	Experimental circular trajectory. Discontinuous controller. NED position.	62
4.8	Experimental circular trajectory. Discontinuous controller. NED position errors.	62
4.9	Experimental circular trajectory. Discontinuous controller. Trajectory.	63
4.10	Experimental circular trajectory. Discontinuous controller. 3D trajectory.	63
4.11	Experimental circular trajectory. Discontinuous controller. Attitude.	64
4.12	Experimental circular trajectory. Discontinuous controller. Attitude errors.	64
4.13	Experimental position set-point. Hybrid kinematic controller. East position.	65
4.14	Experimental position set-point. Hybrid kinematic controller. East position error.	66
4.15	Experimental position set-point. Hybrid kinematic controller. Attitude.	66
4.16	Experimental position set-point. Hybrid kinematic controller. Attitude errors.	67
4.17	Experimental circular trajectory. Hybrid kinematic controller. NED position.	68

4.18	Experimental circular trajectory. Hybrid kinematic controller. NED position errors.	68
4.19	Experimental circular trajectory. Hybrid kinematic controller. Trajectory.	69
4.20	Experimental circular trajectory. Hybrid kinematic controller. 3D trajectory.	69
4.21	Experimental circular trajectory. Hybrid kinematic controller. Attitude.	70
4.22	Experimental circular trajectory. Hybrid kinematic controller. Attitude errors.	70
4.23	Experimental position set-point. Hybrid PID controller. East position.	71
4.24	Experimental position set-point. Hybrid PID controller. East position error.	72
4.25	Experimental position set-point. Hybrid PID controller. Attitude.	72
4.26	Experimental position set-point. Hybrid PID controller. Attitude errors.	73
4.27	Experimental circular trajectory. Hybrid PID controller. NED position.	74
4.28	Experimental circular trajectory. Hybrid PID controller. NED position errors.	74
4.29	Experimental circular trajectory. Hybrid PID controller. Trajectory.	75
4.30	Experimental circular trajectory. Hybrid PID controller. 3D trajectory.	75
4.31	Experimental circular trajectory. Hybrid PID controller. Attitude.	76
4.32	Experimental circular trajectory. Hybrid PID controller. Attitude errors.	76
4.33	Experimental circular trajectory. Comparison of the three controllers. NE position.	77
4.34	Experimental circular trajectory. Comparison of the three controllers. NE position errors.	78
4.35	Experimental circular trajectory. Comparison of the three controllers. Attitude.	78
4.36	Experimental circular trajectory. Comparison of the three controllers. Attitude errors.	79

List of Tables

3.1	Controller parameters.	27
3.2	PID controller parameters.	30
3.3	Controller parameters	33
4.1	ANT-X characteristics.	56

Introduction

Unmanned Aerial Vehicles (UAVs) are aircraft characterized by the absence of a pilot onboard; usually called drones, they can be piloted from ground or even fly autonomously. The interest for UAV has continuously risen in recent years thanks to their wide range of applications both in military and civil fields, *e.g.*, aerial recon and surveillance, search-and-rescue operations, photography, product delivery and entertainment. This variety of applications naturally led to various categories of products with different control laws, as well as avionics architecture and sensors onboard the drone. The quality of the sensors reflects on the attitude controller which is fed with measurements affected by noise; when the attitude regulator is not correctly designed, this interaction may cause undesired behaviors of the drone.

In this thesis the objective is to investigate the problems related to the attitude parametrization by use of unit-quaternions and to implement an attitude controller robust to measurement noise. The new regulator implements a hybrid and hysteretic logic which allows the controller to preserve its global attractivity, unlike the case of discontinuous controller which simply implements a control law relying on the sign of the quaternion error scalar component. The effect of this strategies is studied with respect to a Vertical Take-Off and Landing (VTOL) quadrotor UAV.

Finally, the system is simulated for various flight conditions and experimentally tested in the Flying Arena for Rotorcraft Technologies (Fly-ART) of the Aerospace Systems and Control Laboratory (ASCL) [7] of Politecnico di Milano.

State of the art

Hybrid systems are dynamical systems which incorporate both continuous and discrete dynamics, so that the states of the system can "flow" when governed by the continuous dynamics and "jump" when governed by the discrete dynamics. The recent development of a comprehensive theory for hybrid dynamic systems [4], [8] that allows to integrate continuous-time dynamical systems and discrete-time dynamical systems in a unified manner, provides a unifying modelling language for different applications to complex control systems (see Mayhew et al. [9], [10] and [11], Casau et al. [12], [13] and [14], Malladi et al. [15], Gozzini et al. [16]

and Poveda et al. [17] for a recent survey). In the literature there are examples of hybrid controller for different applications.

R. Goebel et al. [4] gave an insight into hybrid systems behavior and studied the stability properties of hybrid controller.

Hybrid control techniques are exploited, for example, to create supervisor controllers that can handle different operating modes switching robustly between different controllers, *e.g.*, uniting local and global controllers [18] and [19], or to manage different spacecraft operations [20], [15] and [21].

B. P. Malladi et al. [15] discussed the problem of rendezvous, proximity operations and docking of an autonomous spacecraft characterizing a family of individual controllers. Due to the different constraints, available measurements, and tasks to perform on each phase, a hybrid systems approach is proposed: the system exhibited different modes of operation for which a suitable hybrid supervisor was designed.

R. G. Sanfelice et al. [19] proposed a well-posed construction of general hybrid supervisors for robust, global asymptotic stabilization in nonlinear systems. The hybrid supervisors scheduled an appropriate hybrid controller for every point in the region of operation to accomplish the desired task.

G. Gozzini [22] et al. [16] discussed the problem of automatic air-to-air landing (AAAL) of UAVs and proposed a hybrid logic combined with a quasi time-optimal tracking controller as solution. In particular, the landing of a UAV on a carrier drone is achieved in a safe and quasi time-optimal way.

Hybrid control has been used in recent years to solve the limitation of continuous feedback controller in achieving global asymptotic stabilization of the attitude of rigid bodies; in fact, lack of robustness and the so-called unwinding effect are typical of non-hybrid stabilizer. In the literature there are several examples of hybrid designs for attitude control.

D. Fragopoulos et al. [23] conducted a stability analysis of discontinuous controller considering Lyapunov functions and proving that this type of regulator is not robust to measurement noise.

C. G. Mayhew [3] et al. [5], [24] discussed the global asymptotic stabilization of the attitude of a rigid body. Supposing an attitude parametrization by use of unit-quaternions, a stability analysis by use of Lyapunov functions is conducted. A hybrid control strategy which guarantees global asymptotic stability that is robust to measurement noise is presented as solution: the control scheme employs a

logic variable that defines the desired direction of rotation and a hysteretic trend that can mitigate the unwanted effect of chattering due to measurement noise, whose presence can delay control response and waste energy.

C. G. Mayhew et al. [24] developed a hybrid-dynamic path-lifting algorithm, which, when paired with a quaternion-based feedback, allows to translate stability results obtained in the unit-quaternions space directly to the rigid-body-attitude space.

J. Su et al. [6] proposed a hybrid Proportional-Integral-Derivative (PID) attitude control law for rigid body. It is proved that for the attitude tracking problem, it achieves globally asymptotically stable attitude trajectory that is robust to measurement noise.

Thesis structure

The thesis is organized as follows:

- In Chapter 1 an overview of reference frames and rotations formalism adopted in flight dynamics and throughout the thesis are presented, as well as the attitude dynamics of a quadrotor UAV and its parametrizations. Finally, the topological problems associated with global attitude control are discussed.
- In Chapter 2 modeling of hybrid systems and their application as controllers are described. Attitude control strategies and stability analysis from the literature are reported and, at the end, the control strategies implemented onboard the UAV are presented.
- In Chapter 3 the simulations run before the in-laboratory experiments are described and their results discussed. In particular, it is shown the advantage of using a hybrid attitude controller over a discontinuous controller when measurement noise is present. Finally, a position and attitude set-point and a circular trajectory are simulated within the ANT-X complete simulator for both the kinematic hybrid controller and the hybrid PID controller.
- In Chapter 4, after a brief description of the experimental setup (hardware and software), the experimental results are discussed. In particular, flight tests of a position and attitude set-point and a circular trajectory are presented for the baseline controller (discontinuous), the kinematic hybrid controller, and the hybrid PID controller.

Chapter 1

Problem formulation

This chapter is divided in three main parts: in the first one, conventions, formalism and reference frame of flight dynamics are presented; the second part provides an insight into the flight dynamics of an UAV. To take an overall view of these parts, see [25]. The last part deals with the problem of control strategies for attitude stabilization [3].

1.1 Definition of reference frames

In order to describe the motion and the attitude of an aircraft, more reference frames are needed.

1.1.1 NED frame

The North-East-Down (NED) frame, also known as Local Tangent Plane (LTP), represents the usual choice for the reference system on Earth. Assuming the hypotheses of flat and non-rotating Earth, and that the gravity is constant and normal to the tangent plane, Earth fixed frame is defined as $\mathcal{F}_E = \{O_E, e_{1E}, e_{2E}, e_{3E}\}$, where the first element represents the origin and the others are three unit vectors. The origin is arbitrary, it could be the intersection of the equator, the prime meridian and mean sea level. Conventionally, the unit vector e_{1E} points toward North, e_{3E} is aligned with the direction of gravity and e_{2E} points toward East conventionally to a right-handed coordinate system.

1.1.2 Body frame

When dealing with flight dynamics, it is easier to consider a reference frame attached to the body of the aircraft: a moving right-handed reference frame $\mathcal{F}_B = \{O_B, e_{1B}, e_{2B}, e_{3B}\}$ is centered in the center of gravity of the UAV. The unit vector e_{1B} lies in the plane of symmetry of the aircraft and points forward, e_{2B} points

rightward perpendicularly to the plane of symmetry and e_{3B} points downward (*frd* convention).

1.2 Rotation of reference frames

1.2.1 Direction cosine matrix

Let v be a generic vector, $\mathcal{F}_a = \{O, I, J, K\}$ and $\mathcal{F}_b = \{o, i, j, k\}$ two distinct reference frames with their own origin and unit axes. In general, the vector v can be expressed with respect to both reference frames \mathcal{F}_a and \mathcal{F}_b as $v^a = [x^a, y^a, z^a]$ and $v^b = [x^b, y^b, z^b]$, respectively. By using the reference frame unit vectors, v^b can be written as

$$v^b = x^b i + y^b j + z^b k. \quad (1.1)$$

It is now possible to express the vector v^b of equation (1.1) with respect to the reference frame \mathcal{F}_a as

$$v^a = x^b i^a + y^b j^a + z^b k^a = [i^a \ j^a \ k^a] v^b = R_b^a v^b. \quad (1.2)$$

The direction cosine matrix $R_b^a = [i^a \ j^a \ k^a]$ represents the rotation from frame \mathcal{F}_b to frame \mathcal{F}_a .

Hence, the attitude of a rigid body is represented by a 3×3 rotation matrix, describing the rotation between two reference frames. The set of 3×3 orthogonal rotation matrices with positive determinant is the special orthogonal group of order three,

$$\text{SO}(3) = \{R \in \mathbb{R}^{3 \times 3} : R^T R = I, \det R = 1\}, \quad (1.3)$$

where $I \in \mathbb{R}^{3 \times 3}$ denotes the identity matrix.

1.2.2 Euler Angles

The Euler angles (ϕ, θ, ψ) , commonly known as roll, pitch and yaw, result from a rotations sequence required to align a reference frame on Earth's surface with an aircraft body-fixed frame. The rotation matrix R_B^E is obtained combining three elementary rotations (rotations occurring with respect to one of the three unit vectors of the frame). Consider, for example, a rotation about the x -axis of an angle ϕ : it does not affect the component of the vector directed along the rotation axis, however it does change the y and z components. The x -axis rotation matrix is given by

$$R_X(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}, \quad (1.4)$$

Similarly, the rotation matrices about y -axis and z -axis are given by

$$R_Y(\theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}, \quad (1.5)$$

$$R_Z(\psi) = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (1.6)$$

In flight dynamics, it is widespread adopted the ZYX rotations sequence (321): the rotation matrix is computed as $R_B^E = R_Z R_Y R_X$, so

$$R_B^E = \begin{bmatrix} C_\theta C_\psi & C_\theta S_\psi & -S_\theta \\ S_\phi S_\theta C_\psi - C_\phi S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & S_\phi C_\theta \\ C_\phi S_\theta C_\psi - S_\phi S_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi & C_\phi C_\theta \end{bmatrix}, \quad (1.7)$$

where $C_\alpha = \cos(\alpha)$ and $S_\alpha = \sin(\alpha)$, for compactness. The time derivatives of

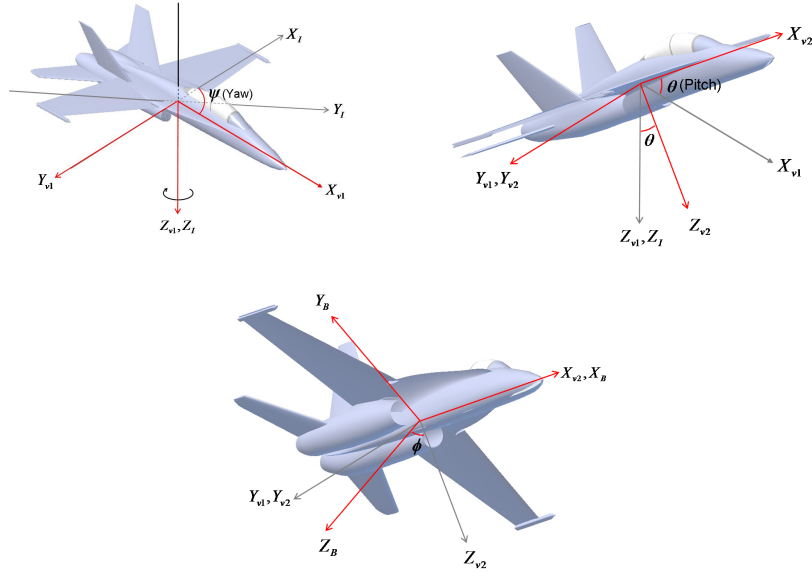


Figure 1.1: Euler angles rotations (from [1])

Euler angles are the Euler rates $(\dot{\phi}, \dot{\theta}, \dot{\psi})$. They are related to the angular rates in body frame $\omega_b = [p, q, r]$ by

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}. \quad (1.8)$$

1.2.3 Quaternions

W. R. Hamilton (1805 - 1865) introduced a quaternion as a four-dimensional representation of a sphere to be used to determine the orientation of a rigid body in three-dimensional space. A unit quaternion q is defined as:

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}, \quad \|q\| = 1. \quad (1.9)$$

By referring to the Euler axis/angle parametrization in which any attitude rotation can always be interpreted through an angle of rotation ϕ and a vector \mathbf{e} representing the axis of rotation, equation (1.9) can be rewritten as:

$$q = \begin{bmatrix} \cos(\frac{\phi}{2}) \\ e_1 \sin(\frac{\phi}{2}) \\ e_2 \sin(\frac{\phi}{2}) \\ e_3 \sin(\frac{\phi}{2}) \end{bmatrix}. \quad (1.10)$$

Quaternions are particularly well-suited for numerical simulation and control since the related coordinate transformation involves quadratic forms only:

$$A(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}. \quad (1.11)$$

The Euler angles can be computed from equation (1.11) as:

$$\begin{aligned} \phi &= \arctan\left(\frac{A(2,3)}{A(3,3)}\right), \\ \theta &= \arctan(-A(1,3)), \\ \psi &= \arctan\left(\frac{A(1,2)}{A(1,1)}\right). \end{aligned} \quad (1.12)$$

Once again, let v be a generic vector, \mathcal{F}_a and \mathcal{F}_b two different reference frames and q_{AB} the quaternion representing the orientation of frame \mathcal{F}_b with respect to frame \mathcal{F}_a . Vector v can be expressed relative to frame \mathcal{F}_b exploiting the following relationship:

$$v_B = q_{AB} \otimes v_a \otimes q_{AB}^*, \quad (1.13)$$

where q_{AB}^* denotes the quaternion conjugate of q_{AB} . It represents the orientation of frame \mathcal{F}_a with respect to frame \mathcal{F}_b . The quaternion conjugate, namely q_{BA} , can be written as:

$$q_{BA} = q_{AB}^* = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix}. \quad (1.14)$$

Equation (1.13) also introduces the quaternion product operator (\otimes). Let \mathcal{F}_c be a third reference frame, it results:

$$q_{AC} = q_{BC} \otimes q_{AB} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \otimes \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 \\ a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2 \\ a_0b_2 + a_2b_0 + a_3b_1 - a_1b_3 \\ a_0b_3 + a_3b_0 + a_1b_2 - a_2b_1 \end{bmatrix}. \quad (1.15)$$

1.3 Attitude dynamics of a quadrotor

A quadrotor UAV can be described as a rigid body of constant mass m and constant inertia matrix $J \in \mathbb{R}^{3 \times 3}$. Let $R \in \text{SO}(3)$ represent the rotation of a vector from the body frame to the inertial frame, and $\omega_b \in \mathbb{R}^3$ the UAV angular velocity in the body frame. The attitude kinematics of a rigid body are expressed as:

$$\dot{R} = RS(\omega), \quad R \in \text{SO}(3), \quad (1.16)$$

where the skew operator $S(\omega)$ is defined as

$$S(\omega) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (1.17)$$

When dealing with quaternions, equation (1.16) involves time derivatives of quaternions and becomes

$$\dot{q} = \frac{1}{2}q \otimes \nu(\omega) = \frac{1}{2}q \otimes \begin{bmatrix} 0 & \omega^\top \end{bmatrix}^\top. \quad (1.18)$$

Under the rigid body assumption, the angular velocity dynamics are described as:

$$J\dot{\omega} = S(J\omega)\omega + \tau, \quad (1.19)$$

where τ represents the control torque vector. The control torque term τ is the result of the angular moments along the three body axis generated by the four rotor producing different thrusts. In order to define the moments generated, the geometry of the UAV must be described. Figure 1.2 shows the configuration scheme of the quadcopter: each propeller produces a thrust force along the z -axis and a torque that depends on the corresponding angular velocity Ω_i of the propellers by

$$\begin{aligned} T_i &= K_T \Omega_i^2, \quad K_T = C_T \rho A R^2, \\ Q_i &= K_Q \Omega_i^2, \quad K_Q = C_Q \rho A R^3, \end{aligned} \quad (1.20)$$

where C_T and C_Q are the thrust and torque coefficients, ρ the air density, A and R the area of the propeller disk and its radius, respectively. Let b be the distance between the center of gravity and the i -th propeller; the equations of the forces and

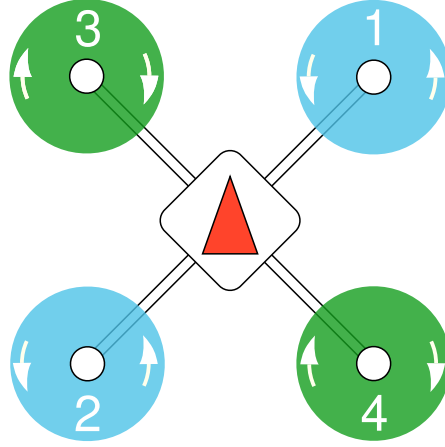


Figure 1.2: Quadcopter scheme ([2])

the moments produced by the propeller expressed in body frame (x -axis pointing forward in the plane of symmetry, y -axis pointing rightward and z -axis pointing downward) are:

$$F_{props} = - \begin{bmatrix} 0 \\ 0 \\ K_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix}, \quad (1.21)$$

$$M_{props} = \begin{bmatrix} K_T \frac{b}{\sqrt{2}}(-\Omega_1^2 + \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \\ K_T \frac{b}{\sqrt{2}}(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \\ K_Q(\Omega_1^2 + \Omega_2^2 - \Omega_3^2 - \Omega_4^2) \end{bmatrix}. \quad (1.22)$$

The control moments can be then related to the rotational speed of the propellers (the control inputs of the UAV) as

$$\begin{bmatrix} T \\ L \\ M \\ N \end{bmatrix} = \begin{bmatrix} K_T & K_T & K_T & K_T \\ -K_T \frac{b}{\sqrt{2}} & K_T \frac{b}{\sqrt{2}} & K_T \frac{b}{\sqrt{2}} & -K_T \frac{b}{\sqrt{2}} \\ K_T \frac{b}{\sqrt{2}} & -K_T \frac{b}{\sqrt{2}} & K_T \frac{b}{\sqrt{2}} & -K_T \frac{b}{\sqrt{2}} \\ K_Q & K_Q & -K_Q & -K_Q \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} = \chi \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix}, \quad (1.23)$$

where the matrix χ is defined as mixer matrix. Hence, the required rotor angular velocities can be computed as:

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} = \chi^{-1} \begin{bmatrix} T \\ L \\ M \\ N \end{bmatrix}. \quad (1.24)$$

1.4 Quaternions and attitude stabilization

1.4.1 Pitfalls of attitude stabilization

As seen in Section 1.2, different parametrizations of the attitude of an aircraft are available. Unit quaternions $q = [\eta \ \epsilon^\top]^\top$ are often used to parameterize rotation matrix of the special orthogonal group $\text{SO}(3)$, where $\eta \in \mathbb{R}$ denotes the scalar component of the quaternion and $\epsilon \in \mathbb{R}^{3 \times 3}$ the vector component. The unit quaternion can be transformed in a rotation matrix through the Rodrigues formula defined through the map $\mathcal{R} : \mathbb{S}^3 \rightarrow \text{SO}(3)$ as

$$\mathcal{R}(q) = I + 2\eta S(\epsilon) + 2S^2(\epsilon), \quad (1.25)$$

where the $S(\cdot)$ operator is defined as in equation (1.17) and $I \in \mathbb{R}^{3 \times 3}$ is the identity matrix. From equation (1.25), it results that there are exactly two unit quaternions, $\pm q$, representing the same physical attitude, namely $R = \mathcal{R}(q) = \mathcal{R}(-q)$. The ambiguity arising from the quaternion representation can cause inconsistent quaternion-based controllers to unnecessarily rotate the rigid body through a full rotation, leading to the so called *unwinding* phenomenon. This behavior can be induced by inconsistent control laws that are designed to stabilize a single point in \mathbb{S}^3 , while leaving the antipodal point unstable, despite the fact that they both correspond to the same physical orientation.

Hence, stabilizing a single attitude in $\text{SO}(3)$, requires stabilizing a disconnected two-point set in the quaternion space and selecting which quaternion to use for control. Let $R_d \in \text{SO}(3)$ denotes a constant desired reference attitude and R the attitude of the rigid body. The error coordinate can be expressed as

$$R_e = R_d^\top R. \quad (1.26)$$

The corresponding kinematic equation is

$$\dot{R}_e = R_e S(\omega_b). \quad (1.27)$$

When dealing with quaternions, the attitude error is computed as a quaternion error

$$q_e = q_{sp}^{-1} \otimes q. \quad (1.28)$$

The attitude control objective is to design a control torque that globally asymptotically stabilizes R_e to the identity matrix. When dealing with quaternions, the unit quaternion associated to $R_e = I$ is the identity element $q_e = \pm \mathbf{1} = [1 \ 0_{1 \times 3}^\top]^\top$. In terms of attitude dynamics, the control requirement is to robustly and globally asymptotically stabilize the set

$$\mathcal{A}_d = \{(q, \omega) \in \mathbb{S}^3 \times \mathbb{R}^3 : q_e = \pm \mathbf{1}, \omega = 0\}. \quad (1.29)$$

Note that, when only the attitude kinematics are considered, the set in equation (1.29) reduces to

$$\mathcal{A}_k = \{q_e \in \mathbb{S}^3 : q_e = \pm \mathbf{1}\}. \quad (1.30)$$

1.4.2 Noise effect on attitude control

As discussed in [3] and [5], robustly and globally stabilizing the disconnected set of points in quaternions space is not achievable with non-hybrid (discontinuous) state feedback controllers when measurement noise is present. To show this fact, consider the Lyapunov function

$$\bar{V}(q) = 2(1 - \eta), \quad (1.31)$$

which satisfies $\bar{V}(\mathbb{S}^3 \setminus \mathbf{1}) > 0$ and $\bar{V}(\mathbf{1}) = 0$. When the feedback law $\omega = -\epsilon$ is applied, the gradient of the Lyapunov function is computed as

$$\left\langle \nabla \bar{V}(q), \frac{1}{2}q \otimes \nu(-\epsilon) \right\rangle = -\|\epsilon\|_2^2. \quad (1.32)$$

This particular choice of feedback law generates two equilibrium points $\pm \mathbf{1}$; in particular, $-\mathbf{1}$ is unstable and $+\mathbf{1}$ is stable. Note that, as discussed in Section 1.4.1, the two points $\pm \mathbf{1}$ represent the same point (same attitude) in $\text{SO}(3)$: therefore, the desired attitude can be either stable or unstable.

To remedy this issue, some authors ([26] and [23]) proposed stabilizing the set $\{\pm \mathbf{1}\}$ with discontinuous control defining the sign function

$$\text{sgn}(\eta) = \begin{cases} -1 & \eta < 0 \\ 1 & \eta \geq 0. \end{cases} \quad (1.33)$$

It results that the feedback law $\omega = -\text{sgn}(\eta)\epsilon$ is globally asymptotically stabilizing with Lyapunov function $\hat{V} = 2(1 - |\eta|)$. However, the global attractivity property is not robust to arbitrary small measurement noise and, as discussed in [5], there exists an arbitrarily small piecewise-constant noise signal that, for initial conditions arbitrarily close to the discontinuity, keeps the state near the discontinuity. Figure 1.3 illustrates the unwinding produced by continuous control, as well as the non-robust global asymptotic stability produced by discontinuous control. Note that the discontinuity lies at $\eta = 0$, which corresponds to attitudes that are a 180° rotation from the desired equilibrium.

1.4.3 ANT-X baseline attitude controller

The objective of this thesis is to compare two different strategies used for attitude stabilization and tracking of UAVs. The attitude controller generates the required moments set-point for the drone to achieve position and attitude tracking. The first controller, in the following referred as "baseline" attitude controller, is a cascade regulator of a PID controller for the angular velocity error, and a proportional controller for the attitude error. The latter, in particular, results in a

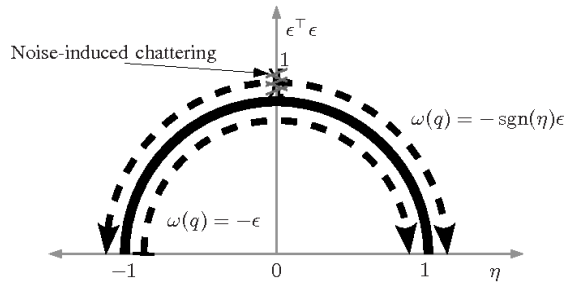


Figure 1.3: Quaternion-based attitude control: non-robust global asymptotic stability. Arrows indicate the direction of rotation. (from [3])

discontinuous controller because the proportional gain multiplies the quaternion error components as:

$$q_e = \text{sgn}(\eta)\epsilon . \tag{1.34}$$

Figure 1.4 shows the control law architecture of the baseline controller. The second regulator is a hybrid controller with hysteresis and has been implemented to solve all the limitations shown by discontinuous (*e.g.*, the baseline attitude controller) and continuous controllers.

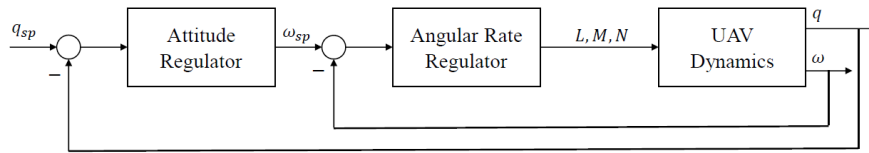


Figure 1.4: ANT-X attitude baseline controller

Chapter 2

Attitude hybrid controller

This chapter opens with a brief introduction to hybrid systems which are dynamical systems possessing both a continuous and discrete behavior ([4]). In the second part, an overall review of control strategies for attitude stabilization and attitude tracking already present in the literature ([5],[27],[6]) is presented. The chapter closes with a discussion on the controller implemented onboard the UAV used for simulations and experiments.

2.1 Modeling of hybrid systems

2.1.1 Hybrid systems preliminaries

Many dynamical systems combine behaviors that are typical of continuous-time dynamical systems with behaviors typical of discrete-time dynamical systems: the interaction between continuous- and discrete-time dynamical systems leads to the so called *hybrid systems*. A widely used representation of continuous-time dynamical systems is the first-order differential equation

$$\dot{x} = f(x), \quad (2.1)$$

where $x \in C \subset \mathbb{R}^n$. The subset C might denote a set of state constraints or a set of initial conditions. When dealing with feedback control systems, error measurements or perturbations must be considered: the right-hand side of the differential equation can be replaced by a set-valued mapping $F(x)$, a multi-valued function. Equation (2.1) becomes

$$\dot{x} \in F(x). \quad (2.2)$$

A typical model of discrete-time dynamical systems is the first order difference equation

$$x^+ = g(x), \quad (2.3)$$

where $g(x)$ is the function relating the next value of the state x^+ and the current state $x \in \mathbb{R}^n$. Similarly to the continuous-time case, it is straightforward to consider constrained difference equations and difference inclusions by introducing the set-valued mapping $G(x)$ and a subset $D \subset \mathbb{R}^n$. A hybrid system \mathcal{H} is defined as

$$\mathcal{H} = \begin{cases} \dot{x} \in F(x) & x \in C \\ x^+ \in G(x) & x \in D, \end{cases} \quad (2.4)$$

where C denotes the flow set, F the flow map, D the jump set, G the jump map. Sets and maps describe the continuous- and discrete-time dynamics and the regions on which these dynamics apply. Solutions to continuous-time dynamical systems are parametrized by $t \in \mathbb{R}_{\geq 0} := [0, +\infty)$, which indicates the time elapsed during flows, whereas solutions to discrete-time dynamical systems are parametrized by $j \in \mathbb{N} := \{0, 1, \dots\}$, which indicates the number of jumps that have occurred. Therefore, the solution to a hybrid system \mathcal{H} is a function parametrized by a *hybrid time domain* (t, j) ; with reference to the example of Figure 2.1 the solution $x(t, j)$ is defined on

$$([0, t_1] \times 0) \cup ([t_1, t_2] \times 1) \cup \dots \cup ([t_j, t_{j+1}] \times j). \quad (2.5)$$

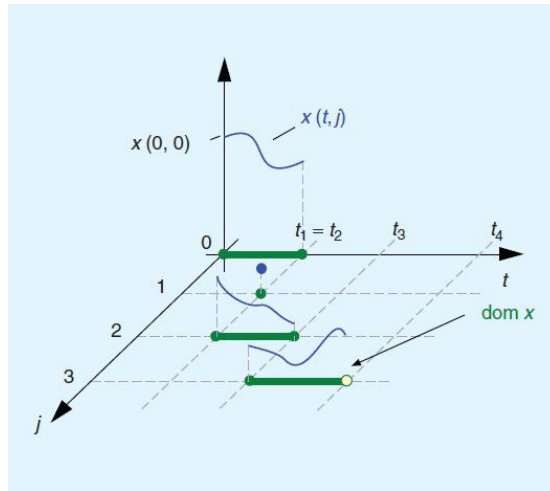


Figure 2.1: Solution to a hybrid system (from [4]).

2.1.2 Hybrid controllers background

Hybrid dynamical systems can model a variety of closed loop feedback control systems; the plant can be either a hybrid system itself or a continuous-time system that is controlled by an algorithm employing discrete-valued states. Consider a plant p described by the differential equation

$$\dot{x}_p = f_p(x_p, u), \quad (2.6)$$

where $x_p \in \mathbb{R}^n$ denotes the state, $u \in \mathbb{R}^r$ the control signal, and f_p a continuous function. A hybrid controller with state $x_c \in \mathbb{R}^m$ is defined by a flow set $C_c \subset \mathbb{R}^{n+m}$, flow map $f_c : C_c \rightarrow \mathbb{R}^n$, jump set $D_c \subset \mathbb{R}^{n+m}$, and a set-valued jump map $G_c : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^n$; the control signal u is specified by a feedback law $k_c : C_c \rightarrow \mathbb{R}^r$. The resulting closed-loop system is a hybrid system with state $x = (x_p, x_c)$, flow set $C = C_c$, and jump set $D = D_c$. During continuous-time evolution ($x = (x_p, x_c) \in C$), the controller state satisfies $\dot{x}_c = f_c(x)$ and the control signal is generated as $u = k_c(x)$. The flow map is defined as

$$F(x) = \begin{bmatrix} f_p(x_p, k_c(x)) \\ f_c(x) \end{bmatrix}, \quad x \in C. \quad (2.7)$$

At jumps ($x = (x_p, x_c) \in D$), the state of the controller is changed based on the rule $x_c^+ \in G_c(x)$. The jump map is defined as

$$G(x) = \begin{bmatrix} x_p \\ G_c(x) \end{bmatrix}, \quad x \in D. \quad (2.8)$$

2.1.3 Hybrid control problem

The solution of a dynamical system may converge to a set rather than to an equilibrium point; for this reason, asymptotic stability of sets must be considered. Given a set \mathcal{A} and a hybrid system \mathcal{H} to be controlled, \mathcal{A} is pre-asymptotically stable if it is stable and pre-attractive. Stability condition requires that

$$\forall \epsilon > 0, \exists \delta > 0 : |x(0, 0)|_{\mathcal{A}} \implies |x(t, j)|_{\mathcal{A}} \leq \epsilon, \quad \forall x(t, j) \in \mathcal{H}, \quad \forall (t, j) \in \text{dom}(x). \quad (2.9)$$

Pre-attractivity condition requires that if there exist a neighborhood of \mathcal{A} from which each solution is bounded, the complete solutions converge to \mathcal{A} , that is,

$$|x(t, j)|_{\mathcal{A}} \rightarrow 0 \text{ as } t + j \rightarrow \infty, \quad (t, j) \in \text{dom}(x). \quad (2.10)$$

In principle, asymptotic stability requires to check the above properties $\forall(\epsilon, \delta)$. However, Lyapunov functions can help when dealing with asymptotic stability problems. Consider the set \mathcal{A} and a hybrid system \mathcal{H} that describes a close-loop system, that is the input is already applied and considered within \mathcal{H} so that no extra input is defined, as required by stability in the sense of Lyapunov.

The set \mathcal{A} is asymptotically stable if there exists a continuously differentiable Lyapunov function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ such that $V(x) \geq 0$, $V(x) = 0 \iff x \in \mathcal{A}$, and

$$\underbrace{\frac{\partial V}{\partial x} f(x)}_{\dot{V}(x)} \leq 0, \quad \forall x \in C \text{ and } \frac{\partial V}{\partial x} f(x) = 0 \iff x \in \mathcal{A}, \quad (2.11)$$

and

$$\underbrace{V(g(x)) - V(x)}_{\Delta V(x)} \leq 0, \quad \forall x \in D \text{ and } V(g(x)) - V(x) = 0 \iff x \in \mathcal{A}, \quad (2.12)$$

that is the variation in energy must be negative in the domain of the solution.

2.2 Attitude stabilization

C. G. Mayhew ([3] and [5]) proposed a quaternion-based hybrid feedback for the attitude stabilization of a rigid body adopting the strategy illustrated in Figure 2.2. The attitude controller implements a dynamic feedback that exploits a memory state to select which pole of \mathbb{S}^3 to regulate in a hysteretic fashion. Let

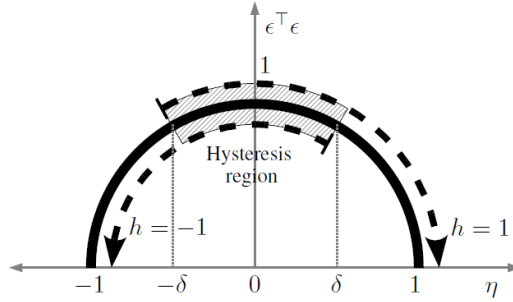


Figure 2.2: Hysteretic regulation of unit quaternions to the set $\{\pm\mathbf{1}\}$ (from [3]).

$\delta \in (0, 1)$ denote the hysteresis half-width and $h \in H := \{-1, 1\}$ a logic variable that selects the desired rotation direction to move q to either $+\mathbf{1}$ or $-\mathbf{1}$. The feedback law $\omega = -h\epsilon$ proposed in [5] is based on the following dynamics of h :

$$\begin{cases} \dot{h} = 0 & (q, h) \in \{h\eta \geq -\delta\} \\ h^+ = -h & (q, h) \in \{h\eta < -\delta\}. \end{cases} \quad (2.13)$$

The inequalities defining the flow set and the jump set are designed to detect a sign mismatch, which indicates whether the feedback $\omega = -h\epsilon$ is pulling in the direction of the shortest rotation to align q with $\pm\mathbf{1}$ ($h\eta \geq 0$) or a longer rotation ($h\eta < 0$).

Hence, the parameter δ manages a trade-off between robustness to measurement noise and a small amount of unwinding: when $\delta \geq 1$, the value of the logic variable h cannot change and the strategy reduces to a static feedback that induces unwinding; when $\delta = 0$, the resulting control becomes discontinuous and therefore not robust to attitude noise. When the above hybrid feedback is implemented for the stabilization of the *attitude kinematics*, namely the set in equation (1.30), the closed-loop system becomes

$$\begin{cases} \dot{q} = \frac{1}{2}q \otimes \nu(-hK_\epsilon\epsilon) \\ \dot{h} = 0 \end{cases} \left. \vphantom{\begin{cases} \dot{q} = \frac{1}{2}q \otimes \nu(-hK_\epsilon\epsilon) \\ \dot{h} = 0 \end{cases}} \right\} (q, h) \in C \quad (2.14)$$

$$\left. \vphantom{\begin{cases} q^+ = q \\ h^+ = -h \end{cases}} \right\} (q, h) \in D,$$

where the feedback law $\omega = -hK_\epsilon\epsilon$ includes a tunable gain $K_\epsilon = K_\epsilon^T > 0$ and the flow set and the jump set are defined as

$$\begin{aligned} C &= \{(q, h) \in \mathbb{S}^3 \times H : h\eta \geq -\delta\} \\ D &= \{(q, h) \in \mathbb{S}^3 \times H : h\eta < -\delta\}. \end{aligned} \quad (2.15)$$

For the stability analysis of the set

$$\mathcal{A} = \{(q, h) \in \mathbb{S}^3 \times H : q = h\mathbf{1}\}, \quad (2.16)$$

consider a Lyapunov function of the form

$$V(q, h) = 2(1 - h\eta), \quad (2.17)$$

which satisfies $V(q, h) = 0$ if and only if $q = h\mathbf{1}$, while it is positive otherwise. The change in V along flows is computed as

$$\langle \nabla V(q, h), f(q, h) \rangle = -\epsilon^T K_\epsilon \epsilon. \quad (2.18)$$

Note that $-1 < -\delta \leq h\eta$ when $(q, h) \in C$: it follows that $\langle \nabla V(q, h), f(q, h) \rangle < 0$ for all $x \in C$, that is whenever $h\eta \geq -\delta$ and $q \neq h\mathbf{1}$. The change of V over jumps is computed as

$$V(g(q, h)) - V(q, h) = 2(1 - (-h)\eta) - 2(1 - h\eta) = 4h\eta. \quad (2.19)$$

Note that $h\eta \leq -\delta$ when $(q, h) \in D$: it follows that $V(g(q, h)) - V(q, h) \leq -4\delta$ that is, V decreases over jumps of the closed-loop system. This stability analysis implies that the set of equation (2.16) is globally asymptotically stable for the closed-loop system since V is strictly decreasing along all trajectories, except of those starting from the set $\{q = h\mathbf{1}\}$.

When the *attitude dynamics* is introduced, the control objectives requires to stabilize the set \mathcal{A}_d of equation (1.29). The closed-loop system becomes

$$\left. \begin{aligned} \dot{q} &= \frac{1}{2}q \otimes \nu(\omega) \\ \dot{\omega} &= J^{-1}(S(J\omega)\omega + \tau) \\ \dot{h} &= 0 \end{aligned} \right\} (q, \omega, h) \in C, \quad (2.20)$$

$$\left. \begin{aligned} q^+ &= q \\ \omega^+ &= \omega \\ h^+ &= -h \end{aligned} \right\} (q, \omega, h) \in D,$$

where the flow set and the jump set are defined as

$$\begin{aligned} C &= \{(q, \omega, h) \in \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta \geq -\delta\} \\ D &= \{(q, \omega, h) \in \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta < -\delta\}. \end{aligned} \quad (2.21)$$

C. G. Mayhew [5] proposed the energy-based Lyapunov function of equation (2.22) for the stability analysis of the closed loop system described by equation (2.20). Let $c > 0$ and

$$V_d(x) = cV(x) + \frac{1}{2}\omega^T J\omega, \quad (2.22)$$

where $x = (q, h, \omega)$, such that $V_d((\mathbb{S}^3 \times \mathbb{R}^3 \times H) \setminus \mathcal{A}_d) > 0$ and $V_d(\mathcal{A}_d) = 0$. Assume a control torque τ of the form

$$\tau = -ch\epsilon - K_\omega\omega, \quad (2.23)$$

where $K_\omega = K_\omega^T > 0$. The change in V_d along flows is computed as

$$\langle \nabla V_d(x), f(x, \tau(x)) \rangle = -\omega^T K_\omega \omega. \quad (2.24)$$

The change of V_d over jumps is computed as

$$V_d(g(x)) - V_d(x) = 4ch\eta. \quad (2.25)$$

Under the conditions expressed by the flow set and the jump set of equation (2.21), it results that

$$\begin{aligned} \langle \nabla V_d(x), f(x, \tau(x)) \rangle &\leq 0 \text{ for all } x \in C \\ V_d(g(x)) - V_d(x) &< 0 \text{ for all } x \in D. \end{aligned} \quad (2.26)$$

Hence, the energy-based Lyapunov function V_d is strictly decreasing along all trajectories, except along those starting from the set $\{q = h\mathbf{1}\}$ and with $\omega = 0$: the set \mathcal{A}_d is globally asymptotically stable for the closed-loop system described in equation (2.20).

2.3 Attitude tracking

C. G. Mayhew ([3] and [27]) and J. Su [6] analyzed the problem of hybrid control for robust global *attitude tracking*. In particular, Su proposed a Proportional-Integral-Derivative-like control law that prevents the singular orientation and the unwinding phenomenon discussed in Section 1.4, and achieves robust global asymptotic tracking in presence of constant disturbances. Let $q_d = [\eta_d \ \epsilon_d]$ be the unit quaternion representing the desired attitude, and $\omega_d \in \mathbb{R}^3$ be the desired angular velocity. It is a reasonable hypothesis to assume that both ω_d and $\dot{\omega}_d$ are bounded so that the reference trajectory is bounded and well defined. The desired attitude motion evolves as

$$\dot{q}_d = \frac{1}{2}q_d \otimes \nu(\omega_d). \quad (2.27)$$

The attitude error, defined as a quaternion error, and the angular velocity error are computed as

$$\begin{aligned} q_e &= q_d^{-1} \otimes q = [\eta_e \ \epsilon_e] \\ \omega_e &= \omega - \mathcal{R}(q_e)^T \omega_d, \end{aligned} \quad (2.28)$$

where the operator $\mathcal{R}(q_e)$ is defined by equation (1.25) and (q, ω) represents the actual state. The control objective is to design τ so that (q, ω) asymptotically tracks a desired bounded reference trajectory for any initial state satisfying $q_e(0) \in \mathbb{S}^3$ and $\omega_e(0) \in \mathbb{R}^{\neq}$, namely the set $\mathcal{A}_t = \{(q_e, \omega_e) \in \mathbb{S}^3 \times \mathbb{R}^3 : q_e = \pm \mathbf{1}, \omega_e = 0\}$ must be made asymptotically stable. The error dynamics is described as

$$\begin{aligned} \dot{q}_e &= \frac{1}{2} q_e \times \nu(\omega_e) \\ J\dot{\omega}_e &= \Xi(\omega_e, \omega_d) + \tau, \end{aligned} \quad (2.29)$$

where J denotes the inertia matrix and $\Xi(\omega_e, \omega_d) = -S(\omega_e + \mathcal{R}(q_e)\omega_d)J(\omega_e + \mathcal{R}(q_e)\omega_d) + J(S(\omega_e)\mathcal{R}(q_e)\omega_d - \mathcal{R}(q_e)\dot{\omega}_d)$. The tracking control law is designed as

$$\tau = -\Xi(\omega_e, \omega_d) - h(JC_i + c_p I^{3 \times 3})\epsilon_e - C_d \omega_e - C_d C_i I_e \quad (2.30)$$

where $0 < c_p \in \mathbb{R}$, $0 < C_i \in \mathbb{R}^{3 \times 3}$, $0 < C_d \in \mathbb{R}^{3 \times 3}$ are constants, $I_e = \int_0^t h \epsilon_e dt$ represents the integral action, and h a logic variable. The resulting hybrid closed-loop system is defined as

$$\left. \begin{aligned} \dot{I}_e &= h \epsilon_e \\ \dot{q}_e &= \frac{1}{2} q_e \otimes \nu(\omega_e) \\ J\dot{\omega}_e &= -h(JC_i + c_p I^{3 \times 3})\epsilon_e - C_d \omega_e - C_d C_i I_e \\ \dot{h} &= 0 \end{aligned} \right\} (I_e, q_e, \omega_e, h) \in C \quad (2.31)$$

$$\left. \begin{aligned} I_e^+ &= I_e \\ q_e^+ &= q_e \\ \omega_e^+ &= \omega_e \\ h^+ &= -h \end{aligned} \right\} (I_e, q_e, \omega_e, h) \in D,$$

where

$$\begin{aligned} C &= \{(I_e, q_e, \omega_e, h) \in \mathbb{R}^3 \times \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta \geq -\delta\} \\ D &= \{(I_e, q_e, \omega_e, h) \in \mathbb{R}^3 \times \mathbb{S}^3 \times \mathbb{R}^3 \times H : h\eta < -\delta\} \end{aligned} \quad (2.32)$$

represent the flow and jump set, respectively. The global asymptotic stability of the set \mathcal{A}_t is demonstrated in [6] by leveraging a suitable Lyapunov function.

2.4 ANT-X attitude hybrid controller

This section presents the strategies implemented in the attitude controller installed onboard a quadrotor UAV, developed by ANT-X [28], used for simulations and experimental results.

2.4.1 ANT-X attitude stabilization

Based on the strategies reported in Section 2.2, a hybrid controller with hysteretic memory state has been implemented onboard ANT-X for the attitude stabilization. At first, a simple kinematic controller has been studied to familiarize with the hybrid logic and to assess its behavior when coupled with the simple quaternion kinematics. Figure 2.3 illustrates the control law architecture. The quaternion error q_e is computed as in equation (1.28): its scalar component η_e enters the "Hybrid Logic" block and defines the value of h to be used in the feedback law

$$\omega_o = 2 \begin{bmatrix} K_P^{ROLL} & 0 & 0 \\ 0 & K_P^{PITCH} & 0 \\ 0 & 0 & K_P^{YAW} \end{bmatrix} h \epsilon, \quad (2.33)$$

based on the constraints expressed by the flow and the jump set of equation (2.16). Note that the control law expressed in equation (2.33) resembles the strategy adopted by the baseline controller, but substituting the logic variable h to $\text{sgn}(\eta)$ to remedy the discontinuous control issues discussed in Section 1.4. As described in [5], the next natural step is to introduce the *attitude dynamics*;

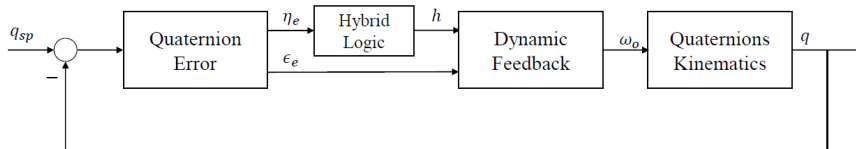


Figure 2.3: Quaternions kinematics hybrid controller.

however, instead of the simple quaternions dynamics, the UAV identified dynamics has been considered. The control system architecture of Figure 2.3 has been modified as illustrated in Figure 2.4, where the "Angular Rate Regulator" block is composed by the same three PID controllers of the baseline strategy (see Section 1.4.3). The UAV identified dynamics comes directly from the ANT-X simulator implemented in MATLAB and Simulink by *Fly-ART* of the *Aerospace System and Control Laboratory* of *Politecnico di Milano*: in particular, black-box models of the attitude dynamics have been identified with the Predictor Based Subspace Identification (PBSID) algorithm [29] using closed loop experimental data. This control system architecture allows to test only the attitude controller without the

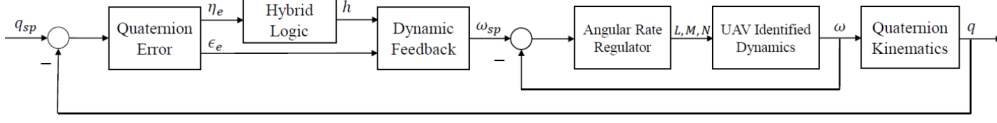


Figure 2.4: ANT-X identified attitude dynamics hybrid controller.

influence of position and velocity control loops, which are instead present in the complete ANT-X simulator.

2.4.2 ANT-X attitude tracking

In line with the strategy reported in Section 2.3 (see [6]), a hybrid controller for attitude tracking has been implemented. As for the case of the attitude dynamics stabilization described in section 2.4.1, only the ANT-X identified attitude dynamics has been at first considered to test the pure attitude tracking controller, without the influence of position and velocity control loops of the complete ANT-X simulator. The control architecture is illustrated in Figure 2.5. Unlike equation

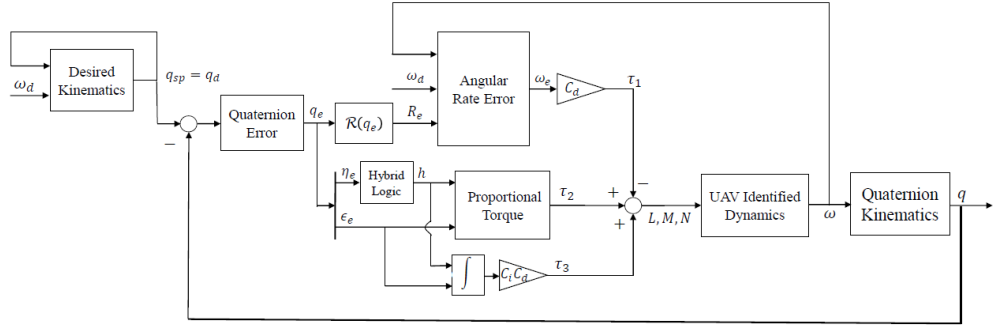


Figure 2.5: ANT-X identified attitude dynamics and hybrid controller for attitude tracking.

(2.30), the control moments L, M, N are computed as the sum of three terms only

$$\begin{aligned}
 \tau_1 &= C_d \omega_e \\
 \tau_2 &= (J C_i + C_p) h \epsilon_e \\
 \tau_3 &= C_i C_d \int_0^t h \epsilon_e dt,
 \end{aligned} \tag{2.34}$$

where $0 < C_p \in \mathbb{R}$ is a scalar gain, $0 < C_i, C_d \in \mathbb{R}^3$ represent two gains matrices, and J denotes the diagonal inertia matrix estimated from the physical model of ANT-X (extra-diagonal terms are negligible). As justified in [30], the control term proportional to $\dot{\omega}_d$ has been neglected since its computation requires continuously

differentiable position trajectories up to the fourth order (see [31]) and, in case the vehicle is manually piloted, the pilot sends commands in terms of desired angles to the onboard controller and the corresponding angular velocity and acceleration must be somehow computed online. Nonetheless, this control term makes the controller structure more complex and has potentially negative effects when only a roughly estimated inertia matrix is available.

Filtering of the reference-signal

The hybrid controller proposed in Section 2.4.2 requires the desired attitude expressed in quaternions space, namely q_d , and the corresponding desired angular velocity ω_d . Following up on the comments presented above, a smooth trajectory generator is needed to provide the controllers a continuously differentiable signal. D. Invernizzi [30] proposed a suitable filter developed on $SO(3)$. However, this thesis implements a quaternion-based hybrid controller, hence the filter has been rearranged to be developed directly on the quaternions space \mathbb{S}^3 . The filter depicted in Figure 4.2 computes the quaternion error q_e between the quaternion setpoint q_{sp} coming directly from the position loop, and the filtered desired quaternion q_d^f , output of the filter itself, as

$$q_e = q_{sp}^{-1} \otimes q_d^f. \quad (2.35)$$

A reworked attitude error matrix e_m is then computed as

$$e_m = \frac{1}{2}(K_R \mathcal{R}(q_e) - \mathcal{R}(q_e)^T K_R), \quad (2.36)$$

where $K_R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, as described in [30]. The matrix e_m is symmetric by construction and the related three-dimensional attitude error vector is generated as

$$e_v = \begin{bmatrix} e_m(3, 2) \\ e_m(1, 3) \\ e_m(2, 1) \end{bmatrix}, \quad (2.37)$$

where $e_m(i, j)$ indicates the component of the matrix e_m of the i -th row, j -th column. The filter dynamics is expressed as

$$\begin{aligned} \dot{q}_d^f &= \frac{1}{2} q_d^f \otimes \omega_f \\ \dot{\omega}_f &= -\omega_n^2 e_v - 2\xi \omega_n \omega_f, \end{aligned} \quad (2.38)$$

where $\omega_n = 20$ and $\xi = 1$ are two constants (see [30]). The output q_d^f and ω_d^f of the filter are then obtained integrating the dynamics equations 2.38. In the simulation environment Simulink, the integration is performed by using two

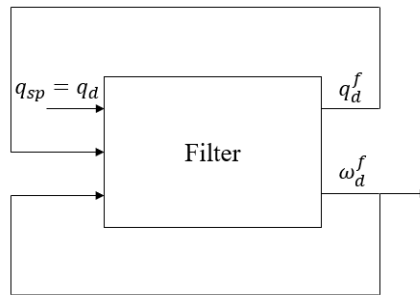


Figure 2.6: ANT-X trajectory filter.

discrete-time integrators of the same form $\frac{T_s}{z-1}$, where T_s denotes the sample time of the model. The resulting ω_d^f is given as input ω_d to the "Angular Rate Error" block of the model depicted in Figure 2.5.

Chapter 3

Simulation results

In this chapter the results of the main simulations are presented.

3.1 Attitude stabilization simulation

3.1.1 Attitude stabilization case from the literature

As pointed out in Section 2.2, the first simulations aimed to get familiar with the hybrid control logic and to assess its behavior. For these reasons, the simulation proposed by C. G. Mayhew [5] has been reproduced. For this simulation, consider an inertia matrix

$$J = \begin{bmatrix} 4.35 & 0 & 0 \\ 0 & 4.33 & 0 \\ 0 & 0 & 3.664 \end{bmatrix} \text{Kg} \cdot \text{m}^2, \quad (3.1)$$

the controller parameters reported in Table 3.1, and initial conditions $h(0) = 1$, $v(0) = \hat{v}/\|\hat{v}\|_2 \text{ m/s}$, $\omega(0) = 2v(0) \text{ rad/s}$, $q(0) = \mathbf{1}$, where $\hat{v} = [3, -4, 5]^\top$. Simulation results correctly reproduce those achieved by C. G. Mayhew in [5] for a control torque $\tau = -ch\epsilon - K_\omega\omega$. Figure 3.1 shows the time history of the quaternions: the controller opposes to the initial velocity and brings back η to +1 stabilizing the desired attitude and stopping the rotation of the body (see Figure 3.3). Figure 3.2 illustrates that no switches of the logic variable h occur since the controller is already stabilizing the correct attitude. Note that, if the initial h was set to -1 , the logic variable would switch to prevent a longer rotation.

$$\begin{array}{ccc} c & K_\omega & \delta \\ 1 & I^{3 \times 3} & 0.11 \end{array}$$

Table 3.1: Controller parameters.

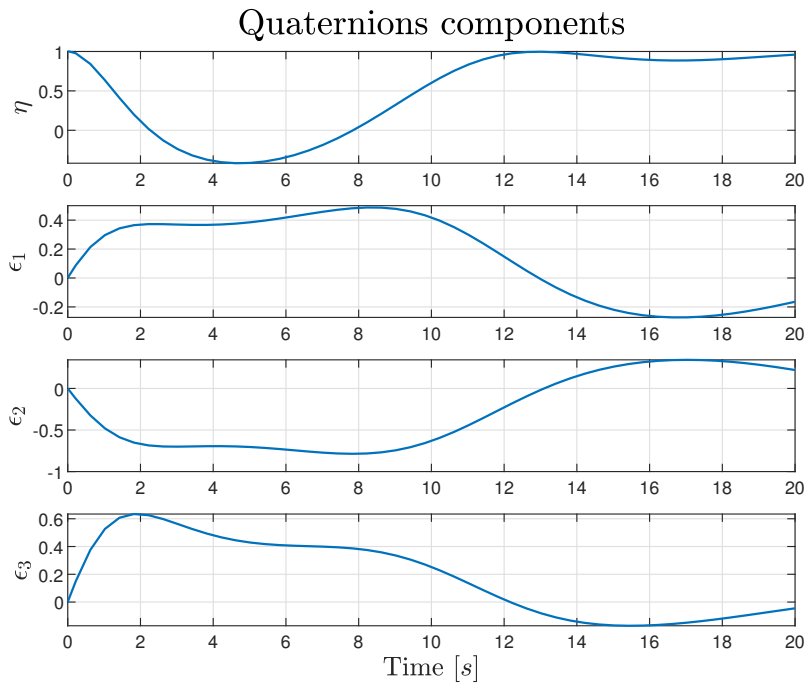


Figure 3.1: Simulation from the paper of C. G. Mayhew et al. [5]: quaternions components.

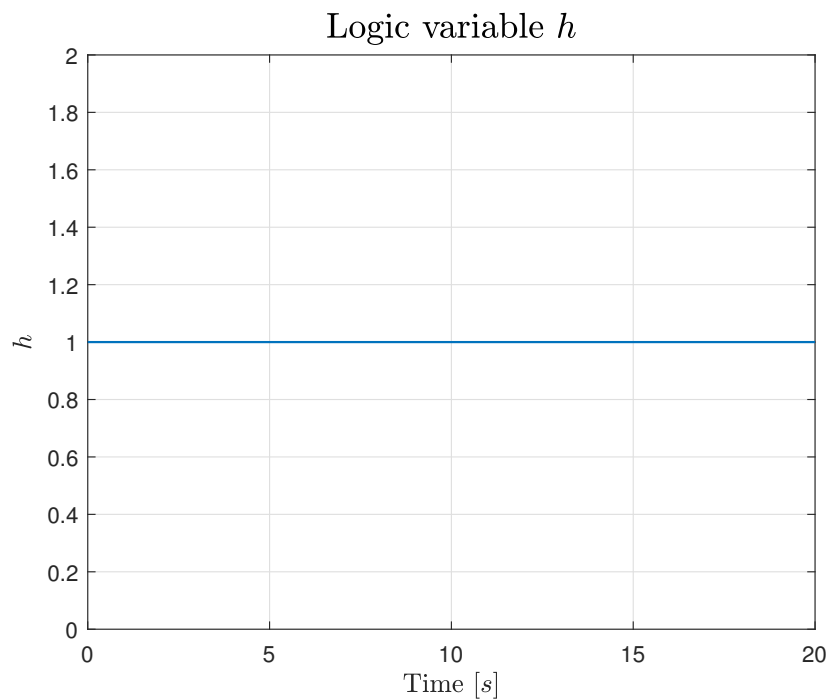


Figure 3.2: Simulation from the paper of C. G. Mayhew et al. [5]: logic variable h .

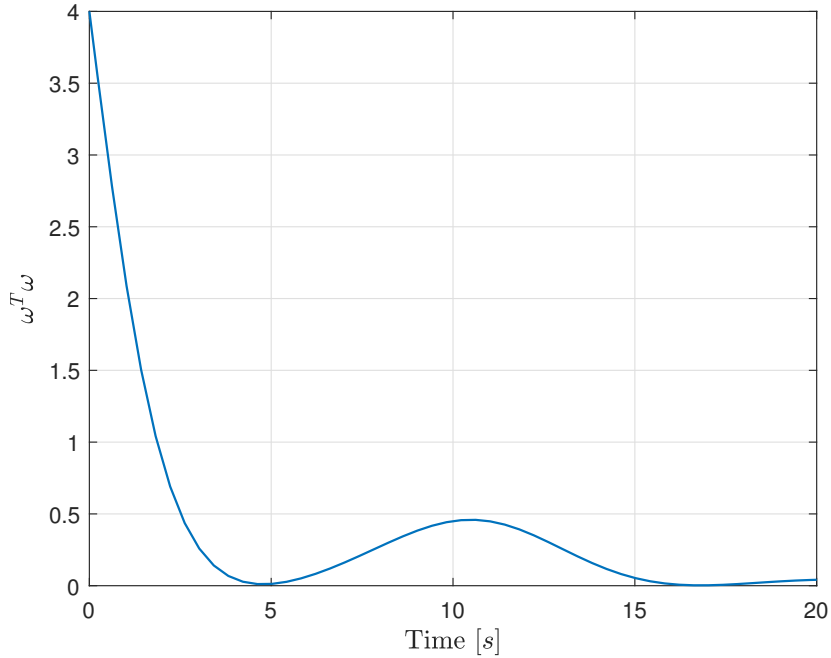


Figure 3.3: Simulation from the paper of C. G. Mayhew et al. [5]: second power of the norm of ω .

3.1.2 Identified dynamics attitude stabilization

Hybrid control has been introduced for its robustness to measurement noise. In order to verify this property, a simulation involving only the identified attitude dynamics of the UAV, as depicted in Figure 2.4, and measurement noise on quaternion has been performed; in particular, the controller is subjected to a measurement noise so that the measured state satisfies $\tilde{q} = (q + e)/\|q + e\|$ where e is selected randomly from a uniform distribution with mean $\mu = 0$ and variance $\text{Var} = \sigma^2$, $\sigma = 0.0349$ representing the attitude noise standard deviation. This simulation shows the benefits of using hysteresis over discontinuous control when measurement noise is present. Consider a hybrid controller such that ω_o has the same form of equation 2.33, and with control parameters $\delta = 0.11$, $h(0) = 1$. For both the hybrid and baseline strategy, $K_P^{ROLL} = 10$, $K_P^{PITCH} = 10$ and $K_P^{YAW} = 2.8$; an angular rate saturation limit of $\pm 10 \text{ rad/s}$ is imposed on all the three components p, q, r . Table 3.2 reports the parameters for each PID of the angular rate regulator; note that the PIDs for roll and pitch motion are characterized by the same parameters due to the geometry of the quadrotor. Control moments saturation limits of ± 1 are present since normalized control actions are considered. This simulation shows the response for a set-point of 180° in yaw, with null roll and pitch angles: the initial conditions are set to be $(0, 0, 0)^\circ$ and $\omega(0) = 0$, the desired attitude $(0, 0, -180)^\circ$. Figures 3.4 and 3.6 show that the hybrid controller

	K_P	K_I	K_D
Roll	0.08	0.05	0.0015
Pitch	0.08	0.05	0.0015
Yaw	0.2	0.1	0

Table 3.2: PID controller parameters.

is impervious to the noise and outperforms the discontinuous strategy, which exhibits a lag in the response (see the detail in Figure 3.5). This is explained by checking the yaw control moment presented in Figure 3.8, and consequently the yaw rate in Figure 3.7, where for the baseline controller a noise-induced chattering forces the UAV to initially oscillate around the initial condition by continuously applying positive and negative control action. Note that the hybrid controller is impervious to measurement noise thanks to the hysteresis variable δ and does not exhibit any chattering. When a particular kind of noise is considered, the baseline controller causes the stabilization of the undesired attitude: this proves that when discontinuous control is used in attempt to break the topological constraints for global stabilization in $SO(3)$, the introduction of arbitrary small measurement noise can destroy any global attractivity.

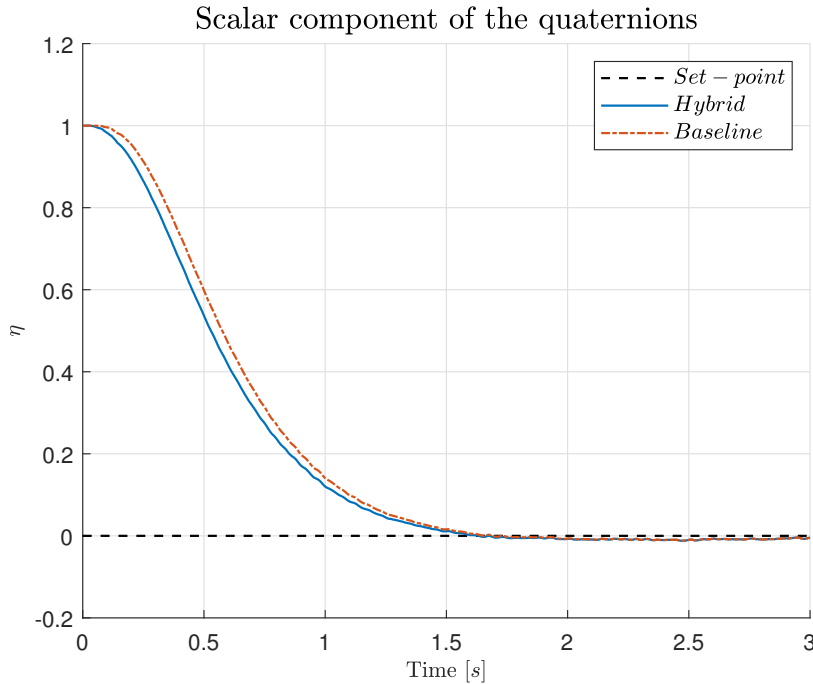


Figure 3.4: Identified dynamics attitude stabilization in presence of measurement noise: scalar components of the quaternions .

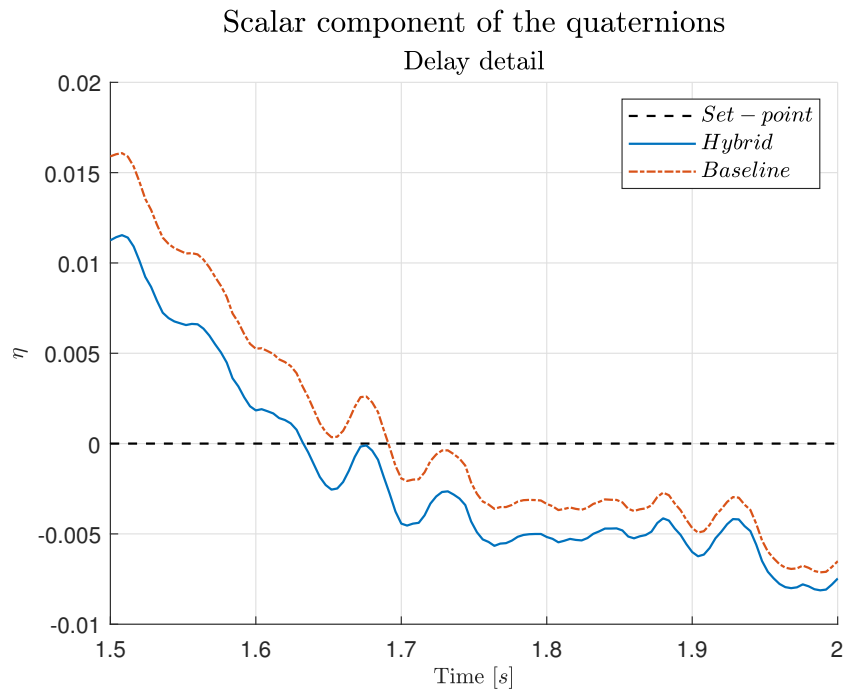


Figure 3.5: Identified dynamics attitude stabilization in presence of measurement noise: detail of the delay of the quaternions scalar components.

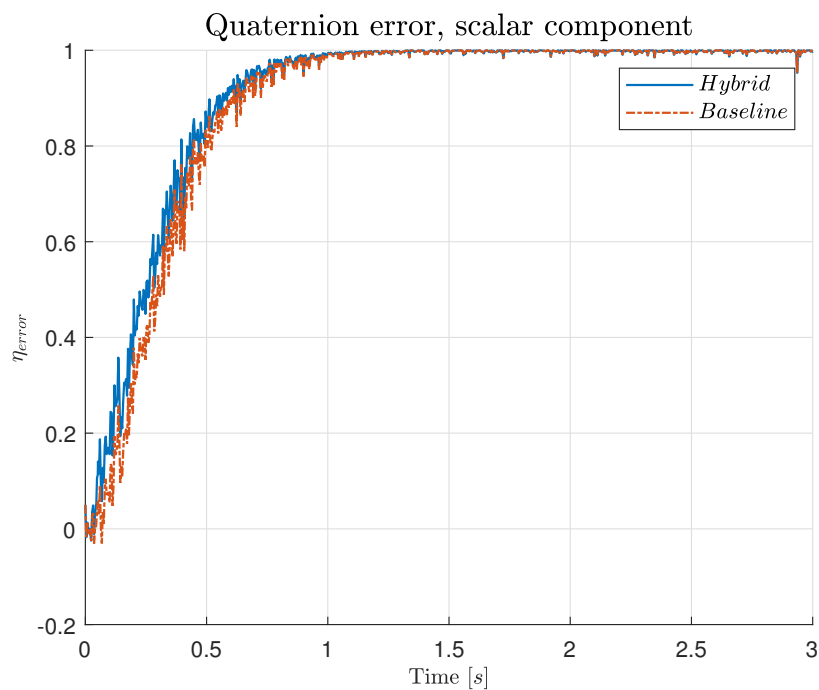


Figure 3.6: Identified dynamics attitude stabilization in presence of measurement noise: scalar component of the quaternion errors.

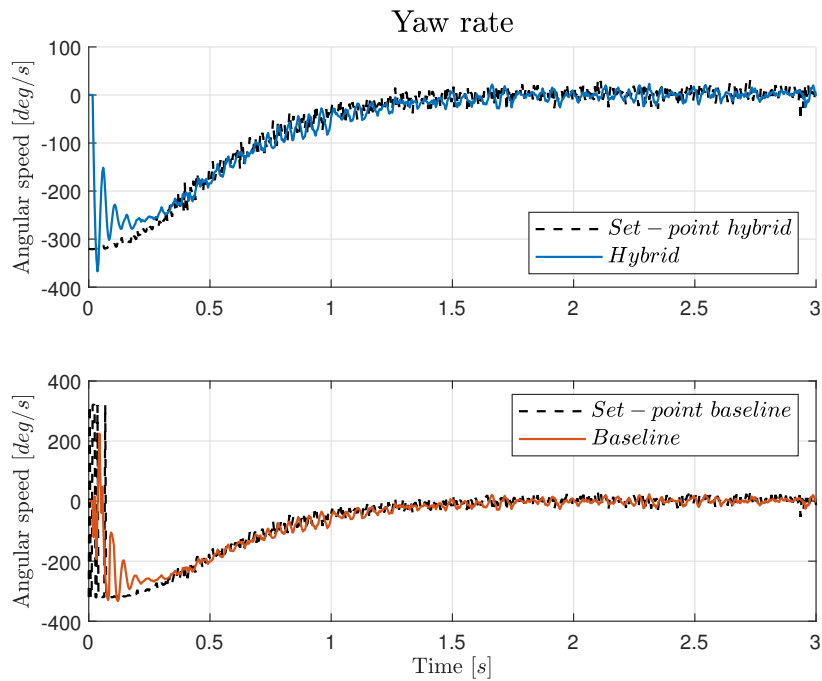


Figure 3.7: Identified dynamics attitude stabilization in presence of measurement noise: yaw rate.

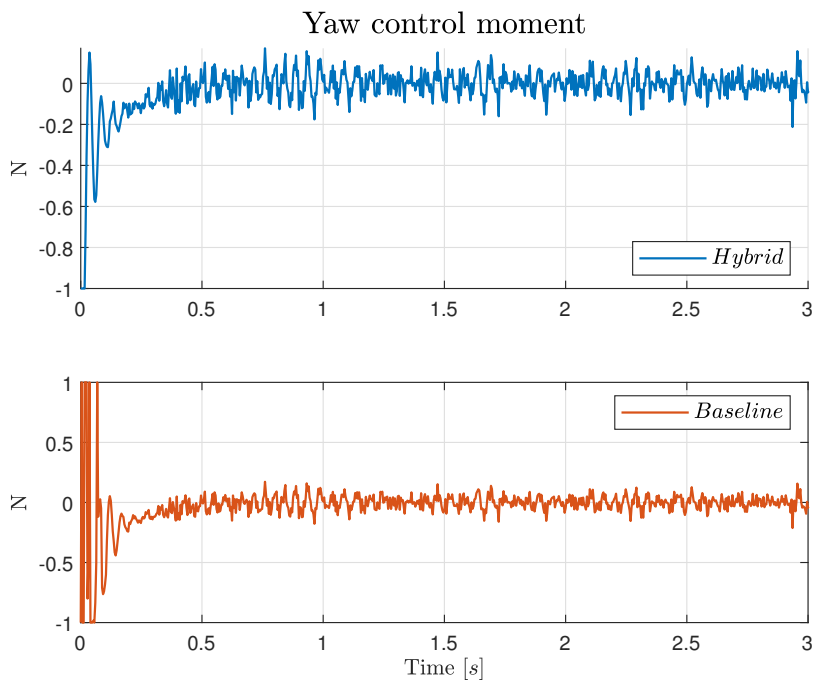


Figure 3.8: Identified dynamics attitude stabilization in presence of measurement noise: yaw control moments.

$$\begin{array}{cccc} C_i & C_d & c_p & \delta \\ \hline 0.1I^{3 \times 3} & 15I^{3 \times 3} & 25 & 0.11 \end{array}$$

Table 3.3: Controller parameters

3.2 Attitude tracking simulation

3.2.1 Attitude tracking case from the literature

As for the case of attitude stabilization, the first tracking simulation assesses the control architecture proposed by J. Su [6]. For this reason, the simulation reported in the paper has been reproduced. Consider a spacecraft approximated as a rigid body whose inertia matrix is

$$J = \begin{bmatrix} 30 & 10 & 5 \\ 10 & 20 & 3 \\ 5 & 3 & 15 \end{bmatrix} \text{ Kg} \cdot \text{m}^2. \quad (3.2)$$

For the trajectory tracking, the initial parameters are $q(0) = [-0.5 \ 0 \ 0 \ 0.866]^\top$, $\omega(0) = [0 \ 0 \ 0]^\top$, $q_d(0) = [1 \ 0 \ 0]^\top$ and $I_e(0) = 0$. The desired angular velocity is

$$\omega_d = [\sin(0.1t) \ \sin(0.1t) \ \sin(0.1t)]^\top. \quad (3.3)$$

The chosen control law parameters are reported in Table 3.3. Figures 3.9 and 3.10 show that q and ω successfully track $-q_d = q_d$ and ω_d , respectively.

3.2.2 Identified dynamics attitude tracking

In order to verify the control architecture depicted in Figure 2.5, a first simulation involving only the identified attitude dynamics has been performed. The desired angular velocity is

$$\omega_d = [0.1 \sin(t) \ 0.1 \sin(t) \ 0.1 \sin(t)]^\top. \quad (3.4)$$

Consider the following initial conditions: attitude $(10, 10, 10)^\circ$, null initial desired attitude, namely $(0, 0, 0)^\circ$, and initial angular rates $\omega(0) = (0.0101, 0.0101, 0.0101)$ rad/s. The controller parameters are:

$$C_i = \begin{bmatrix} 18 & 0 & 0 \\ 0 & 18 & 0 \\ 0 & 0 & 0.003 \end{bmatrix}, \quad (3.5)$$

$$C_d = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}, \quad (3.6)$$

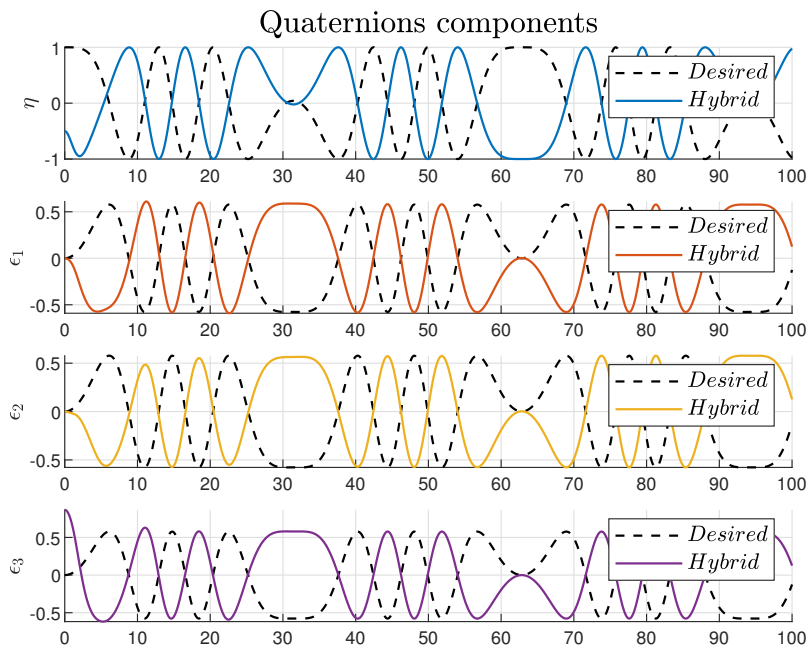


Figure 3.9: Simulation from the paper of J. Su [6]: quaternions components.

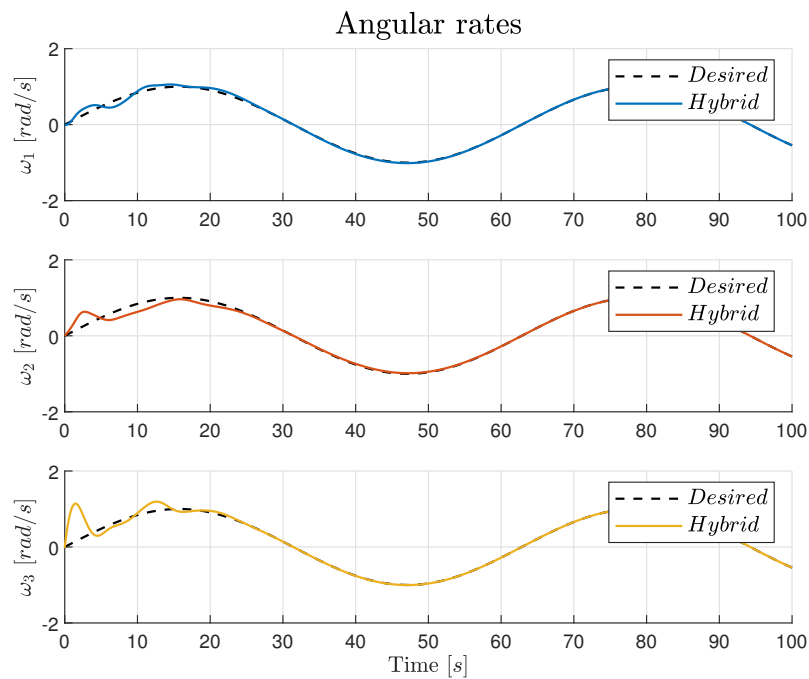


Figure 3.10: Simulation from the paper of J. Su [6]: angular rates.

$c_p = 0.008$ and $\delta = 0.11$. Moreover, the following estimated inertia matrix has been considered for the UAV:

$$J = \begin{bmatrix} 0.00307 & 0 & 0 \\ 0 & 0.00307 & 0 \\ 0 & 0 & 0.00239 \end{bmatrix} \text{Kg} \cdot \text{m}^2. \quad (3.7)$$

Figure 3.12 and 3.14 show the comparison between the hybrid and the baseline strategy in terms of desired quaternions and desired angular velocity, respectively. Both the strategies track the desired trajectory, although the hybrid controller takes more time to stabilize the plant on the desired references; in particular (see Figure 3.11), this trend is valid for roll and pitch angles, whereas in the case of the yaw angle, the hybrid controller stabilizes faster than the baseline, which exhibits delay with respect the desired yaw angle. As a consequence, minor oscillations of the scalar component of the quaternion error for the baseline controller are illustrated in Figure 3.13. Figures 3.17 and 3.18 show that the baseline controller uses more control energy than the hybrid regulator to track the desired attitude faster in the first seconds of the simulation; as a consequence, the UAV exhibits higher angular rates (see Figures 3.14 and 3.15) and therefore rates errors that takes less time to go to zero (see Figure 3.16) .

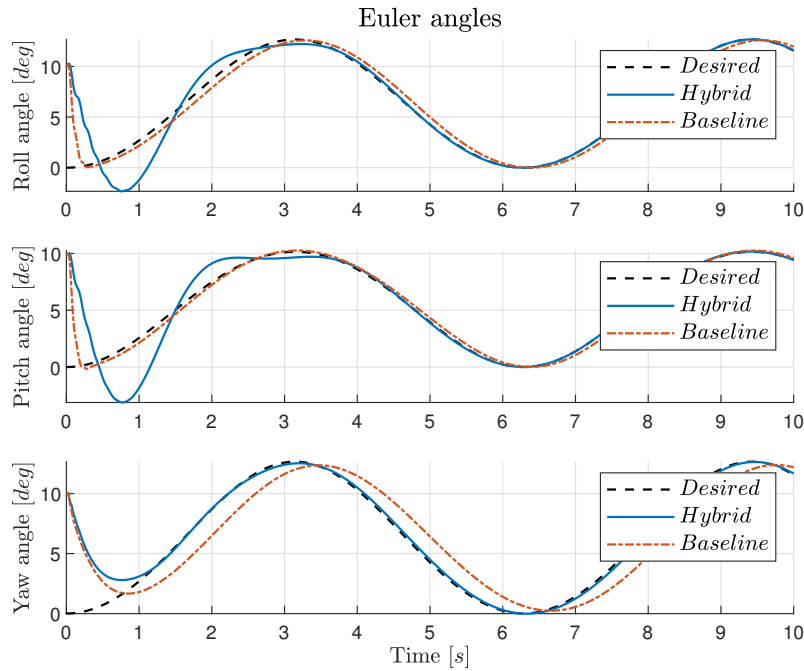


Figure 3.11: Identified dynamics attitude tracking: Euler angles.

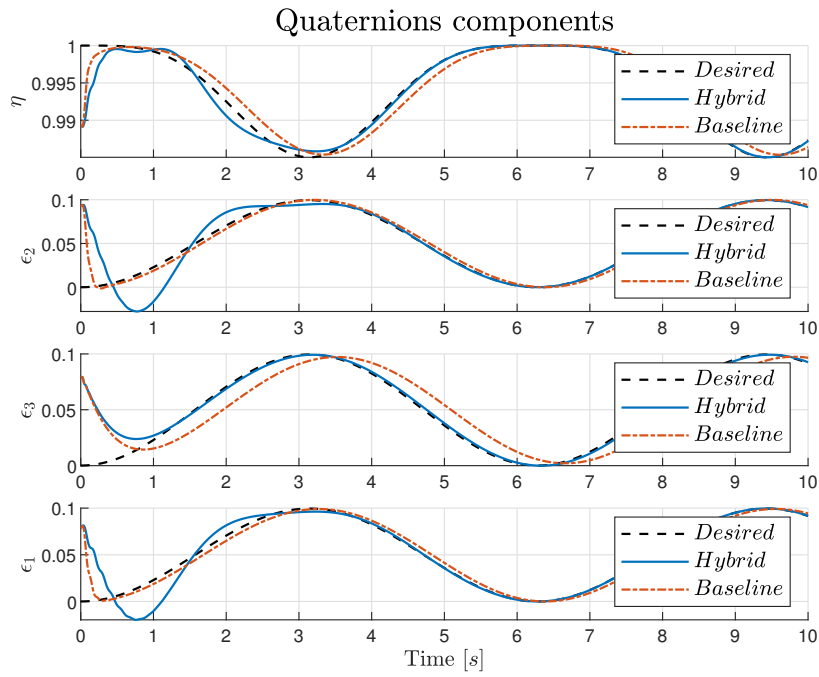


Figure 3.12: Identified dynamics attitude tracking: quaternions components.

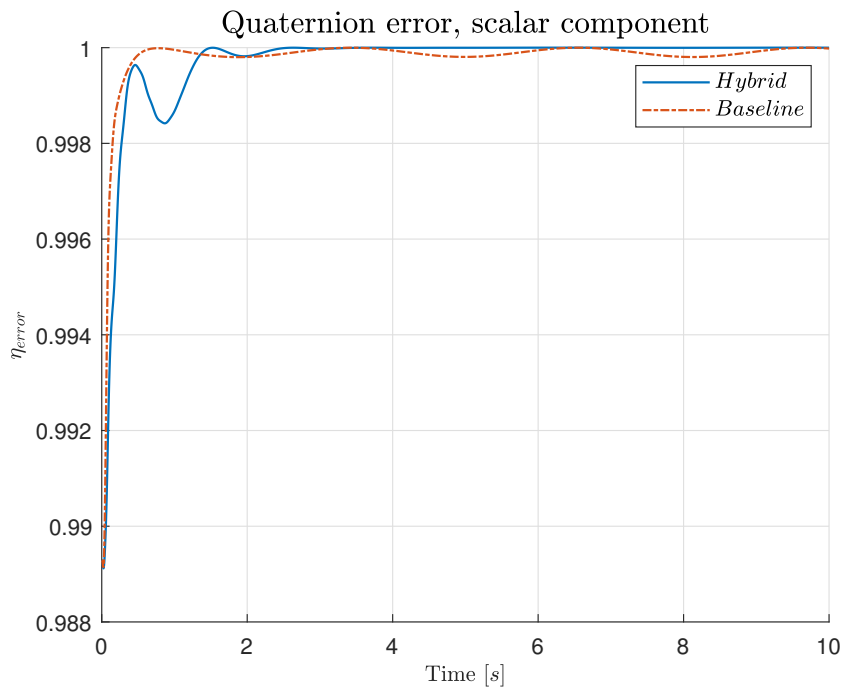


Figure 3.13: Identified dynamics attitude tracking: quaternion error scalar components.

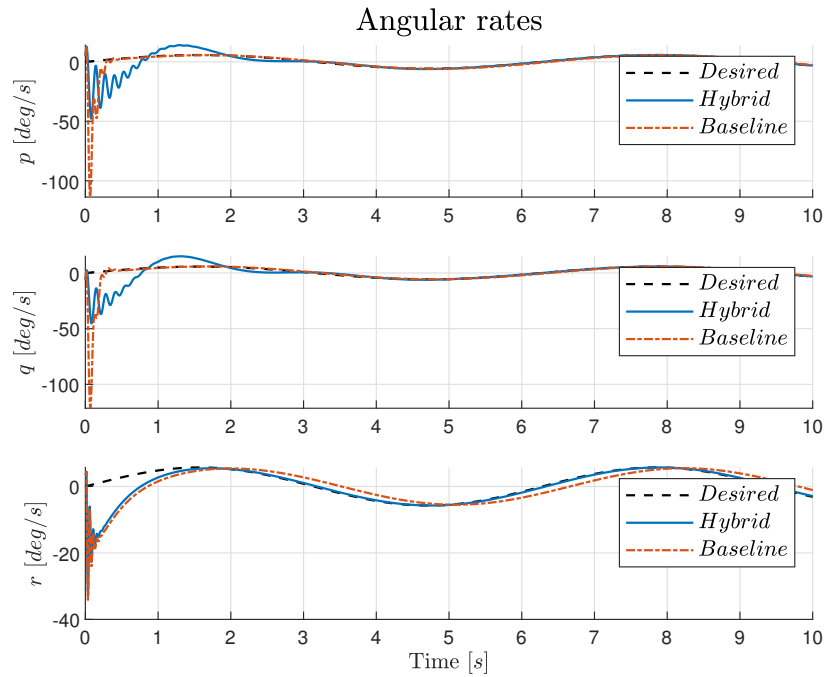


Figure 3.14: Identified dynamics attitude tracking: angular rates.

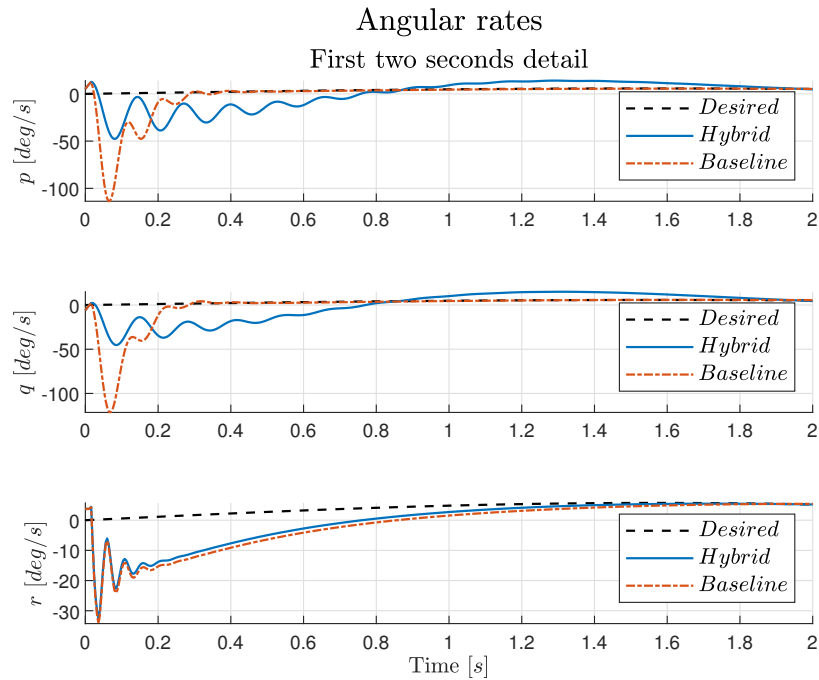


Figure 3.15: Identified dynamics attitude tracking: angular rates detail of the first two seconds of simulation.

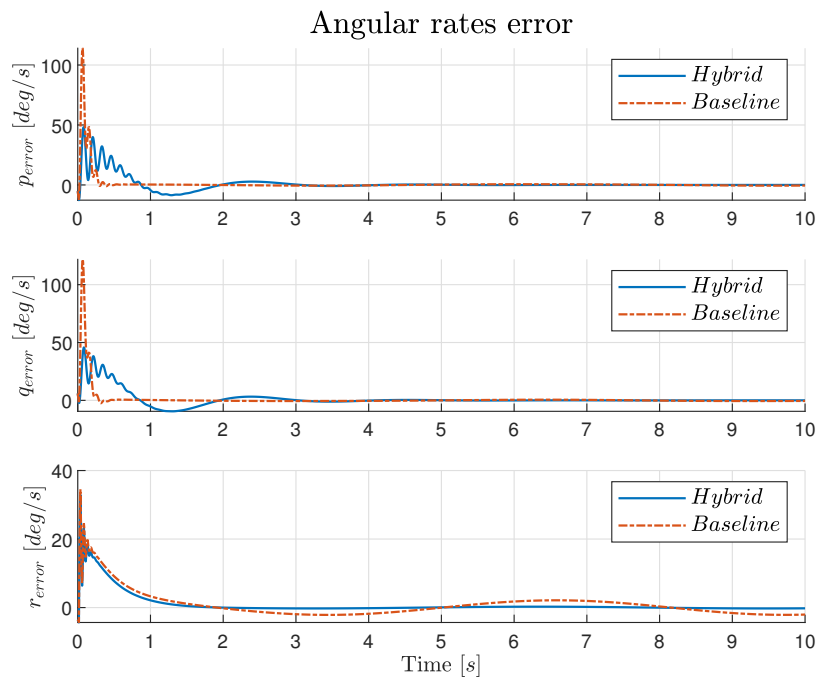


Figure 3.16: Identified dynamics attitude tracking: angular rates errors.

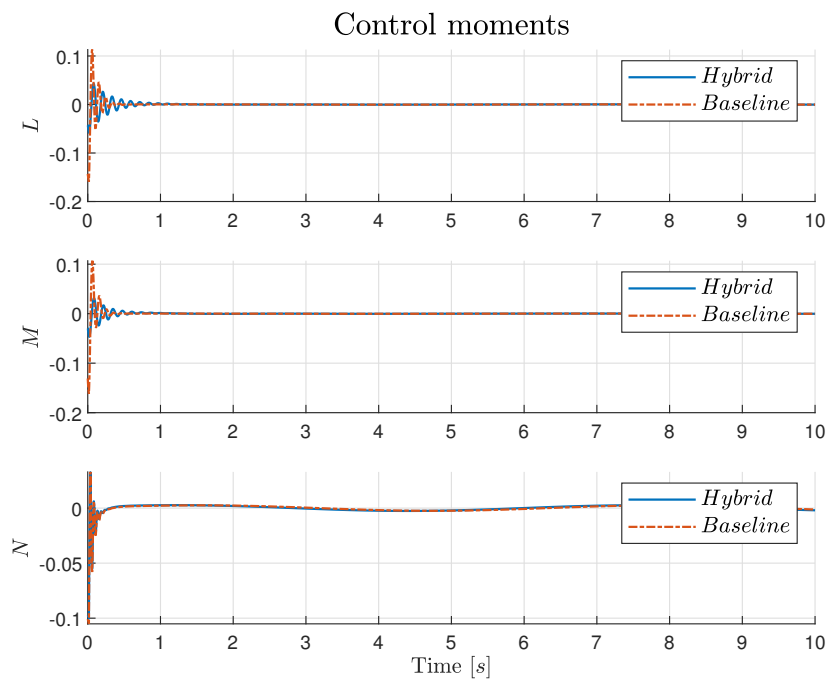


Figure 3.17: Identified dynamics attitude tracking: control moments.

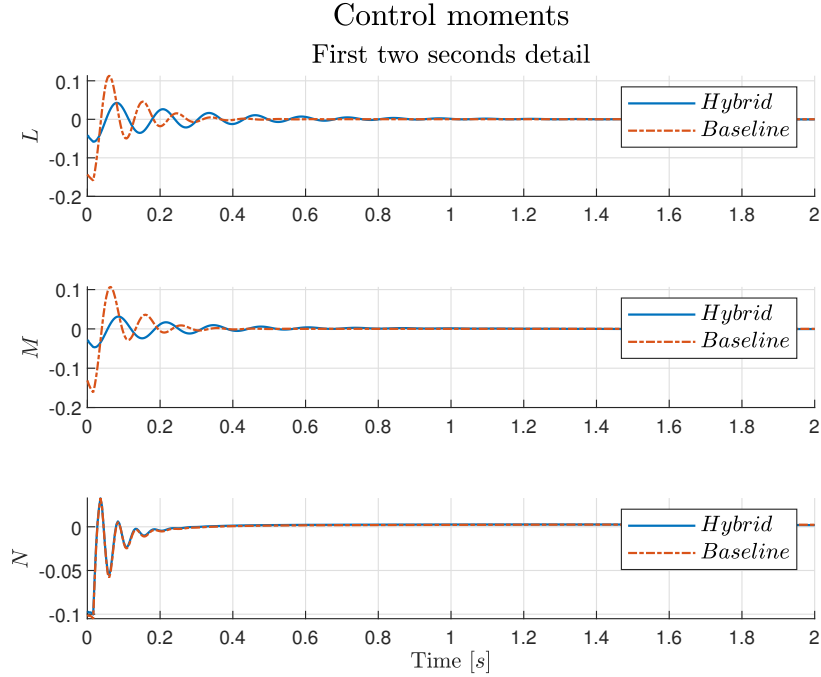


Figure 3.18: Identified dynamics attitude tracking: control moments detail of the first two seconds of simulation.

3.3 ANT-X simulator

In this section the simulation results obtained within the ANT-X simulator are presented. Working in this simulator allows to assess the behavior of the complete control architecture and of the entire system. In particular, a set-point in position, a set-point in attitude, and a circular trajectory are considered in the following.

3.3.1 Hybrid kinematic controller

At first, the controller described in Section 3.1.2 has been implemented within the ANT-X complete simulator. In principle, it is characterized by a hybrid logic; however, since no measurement noise and no high-performance maneuver which can induce a change in the logic variable h has been considered in the following simulations, the regulator behaves like the original controller (baseline, discontinuous): note that in case noise is added, the hybrid solves all the problems of the discontinuous controller.

Position set-point

In this simulation, a position set-point is commanded to the drone, in particular, a unitary step in the East component.

Consider as initial conditions the position $(0, 0, -1.5) m$, expressed in the NED

frame, and the attitude $(0, 0, 0)^\circ$, expressed in the body frame. The desired final conditions are set to be the position $(0, 1, -1.5) m$ and null attitude $(0, 0, 90)^\circ$. Figure 3.19 show that the controller correctly stabilized the desired position with an error that goes to zero after the brief initial transient (see Figure 3.20). Figure 3.21 illustrates that the actual attitude correctly tracks the desired attitude, output of the outer position and velocity control-loop, needed to move the UAV from the initial to the final position. The attitude error is reported in Figure 3.22.

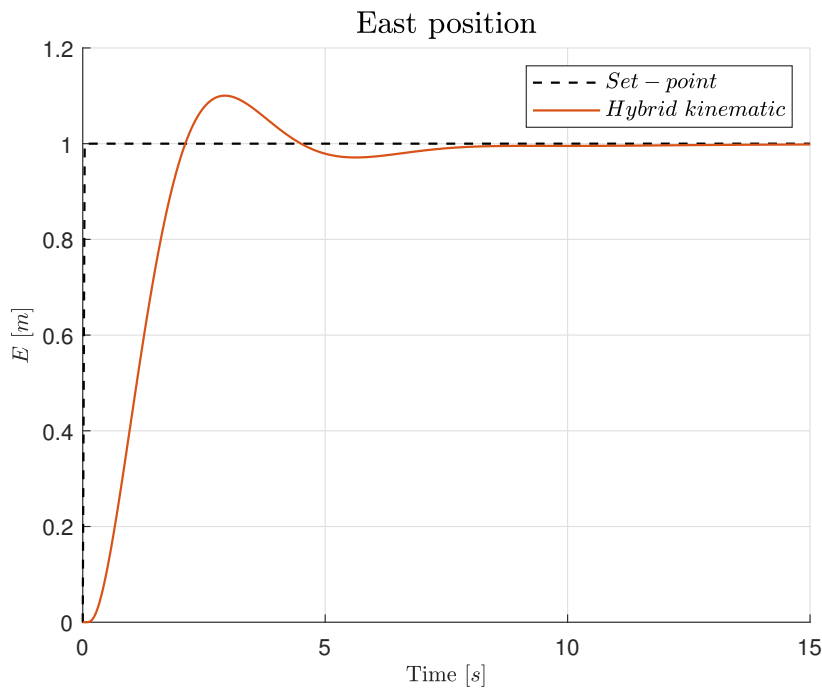


Figure 3.19: ANT-X simulator with hybrid kinematic controller. Position set-point: East position.

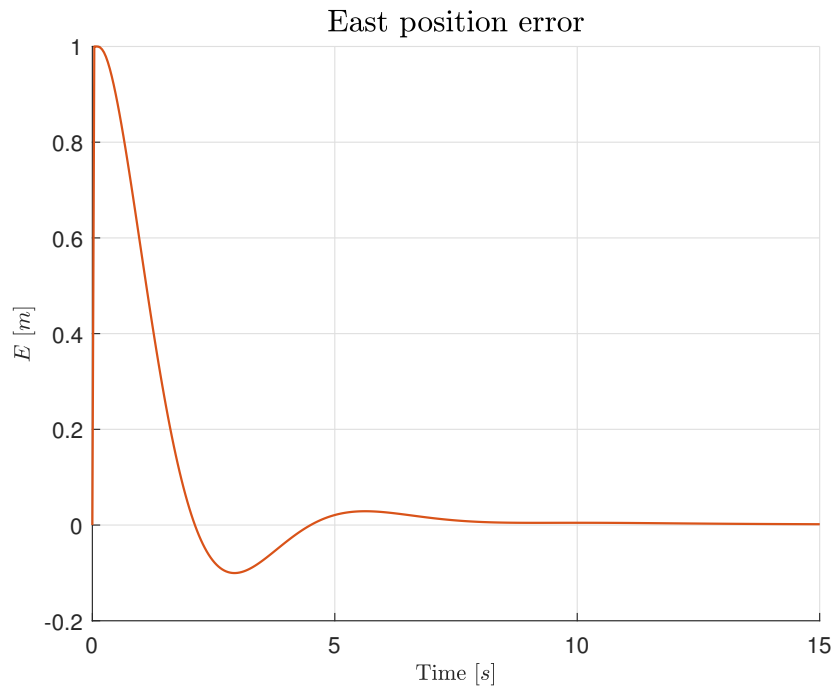


Figure 3.20: ANT-X simulator with hybrid kinematic controller. Position set-point: East position errors.

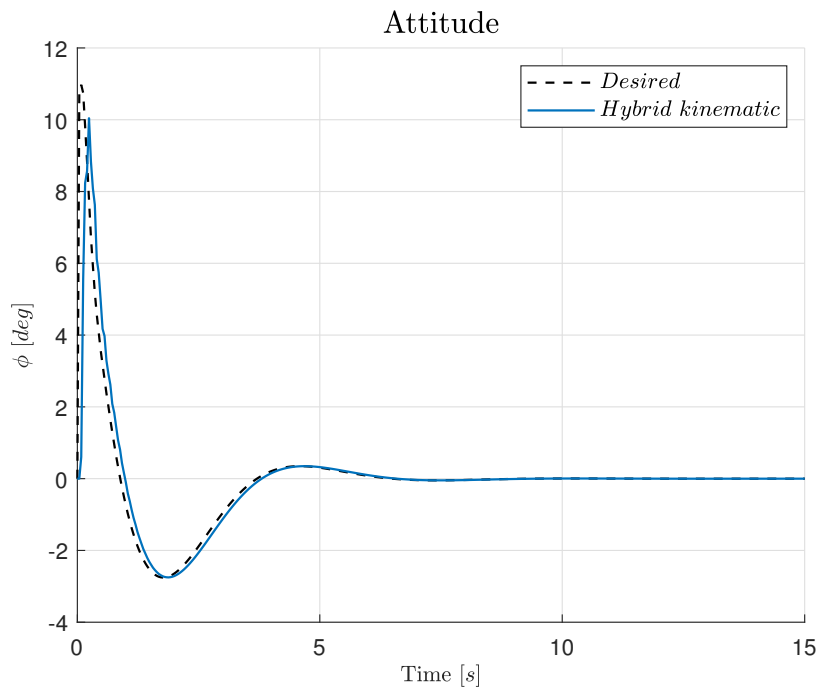


Figure 3.21: ANT-X simulator with hybrid kinematic controller. Position: roll attitude.

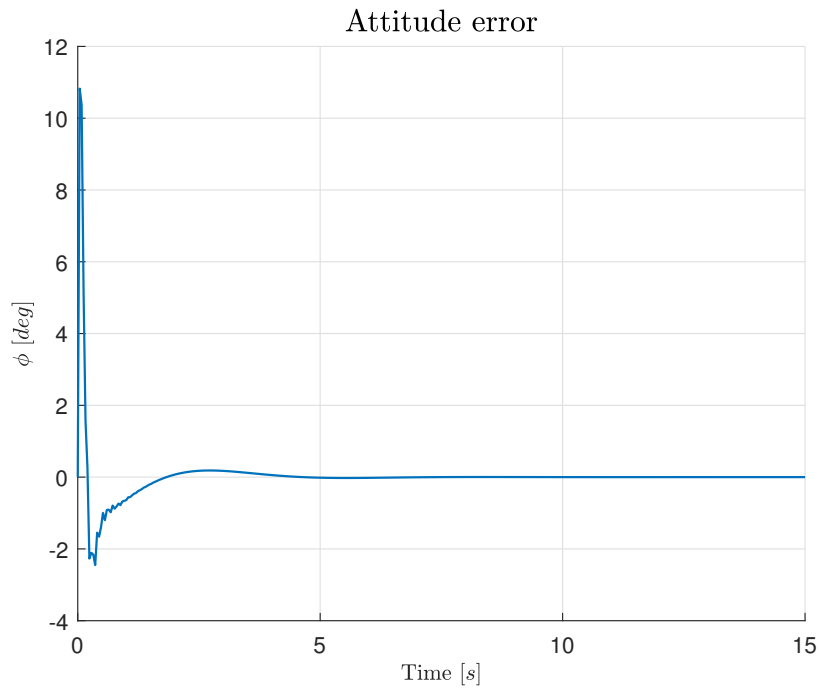


Figure 3.22: ANT-X simulator with hybrid kinematic controller. Position set-point: roll attitude error.

Attitude set-point

In this simulation, an attitude set-point is commanded to the drone, in particular, a rotation of 90° with respect to the yaw axis. Consider as initial condition the attitude $(0, 0, 0)^\circ$; the final attitude to be reached is $(0, 0, 0)^\circ$.

Figure 3.23 shows that the desired attitude is correctly stabilized with an error that goes to zero after the initial transient (see Figure 3.24).

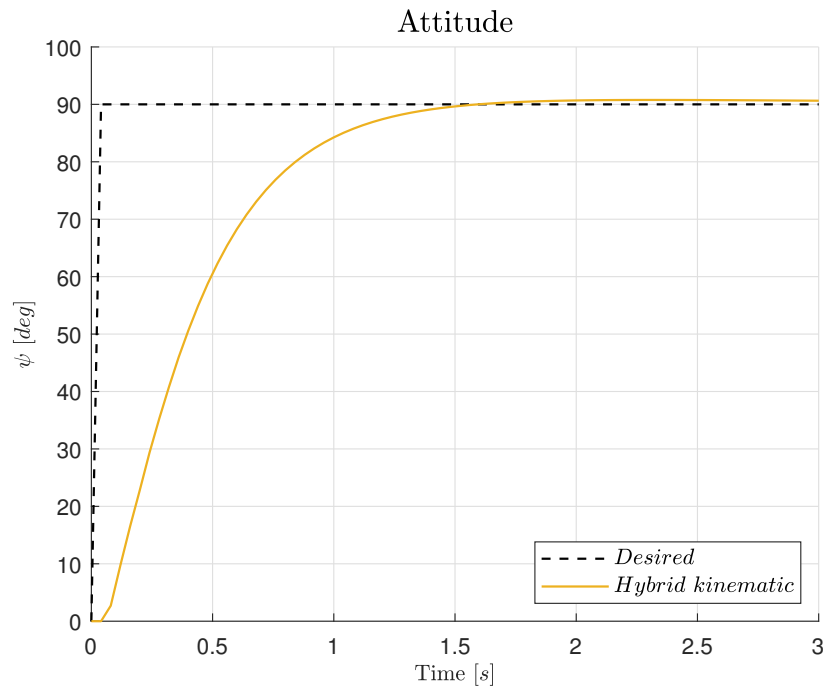


Figure 3.23: ANT-X simulator with hybrid kinematic controller. Attitude set-point: yaw attitude.

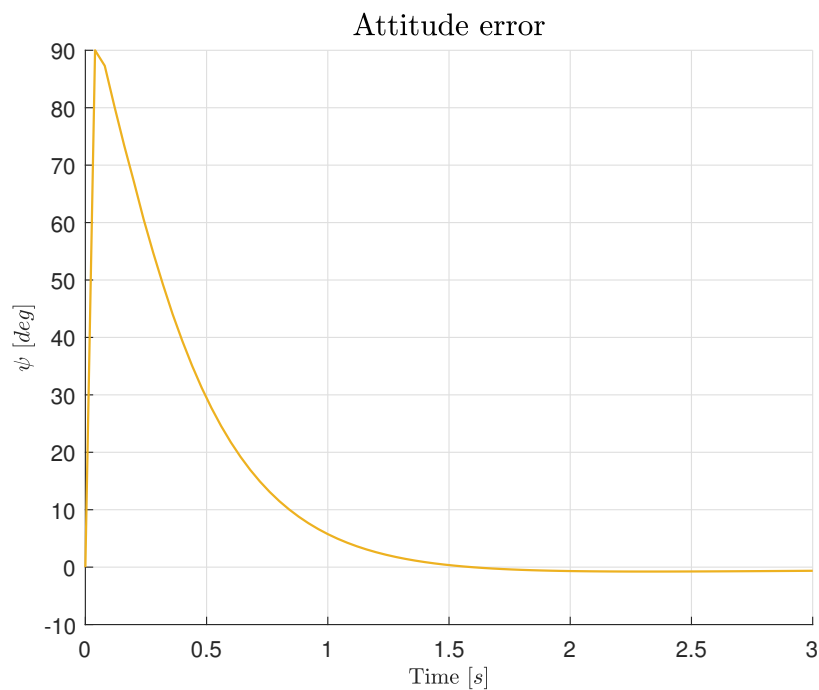


Figure 3.24: ANT-X simulator with hybrid kinematic controller. Attitude set-point: yaw attitude error.

Circular trajectory

In this simulation, a circular trajectory is commanded to the drone. In particular, the desired trajectory has radius $R = 1 \text{ m}$ and a frequency $\omega = 1 \text{ rad/s}$, so that the North and East component of the position are computed as $R \cos(\omega t)$ and $R \sin(\omega t)$, respectively. Consider as initial conditions the center of the circle, namely the position $(0, 0, -1.5) \text{ m}$, expressed in the NED frame, and the attitude $(0, 0, 0)^\circ$. The desired final conditions are set to be the center of the circle at $(0, 0, -1.5) \text{ m}$ and the attitude $(0, 0, 0)^\circ$.

Figure 3.25 shows that the controller is able to correctly track the desired trajectory, but with minor overshoots: as a consequence, the actual circular trajectory is larger than the desired, as illustrated in Figure 3.27 where the N-E in-plane trajectory is reported. The position errors are reported in Figure 3.26. Figure 3.28 show that the controller is able to correctly track the desired attitude, computed by the outer position and velocity loop and needed to run the circular trajectory, with an error that keeps oscillating around zero (see Figure 3.29).

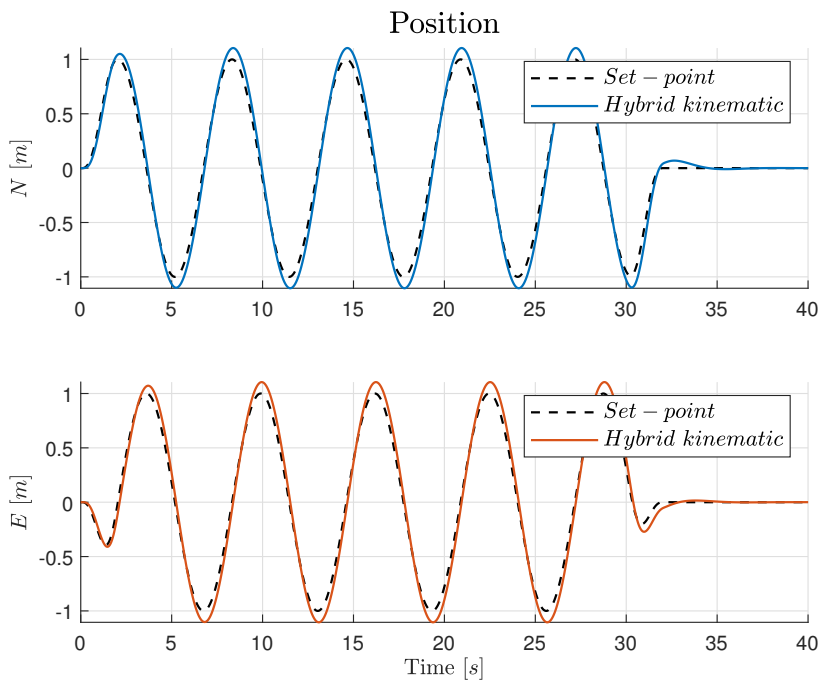


Figure 3.25: ANT-X simulator with hybrid kinematic controller. Circular trajectory: NE position.

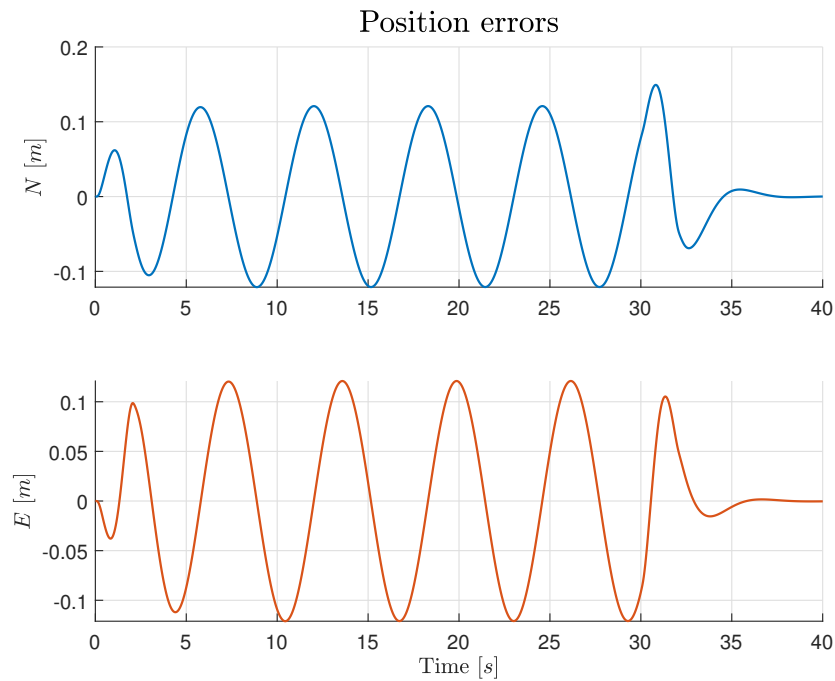


Figure 3.26: ANT-X simulator with hybrid kinematic controller. Circular trajectory: NE position errors.

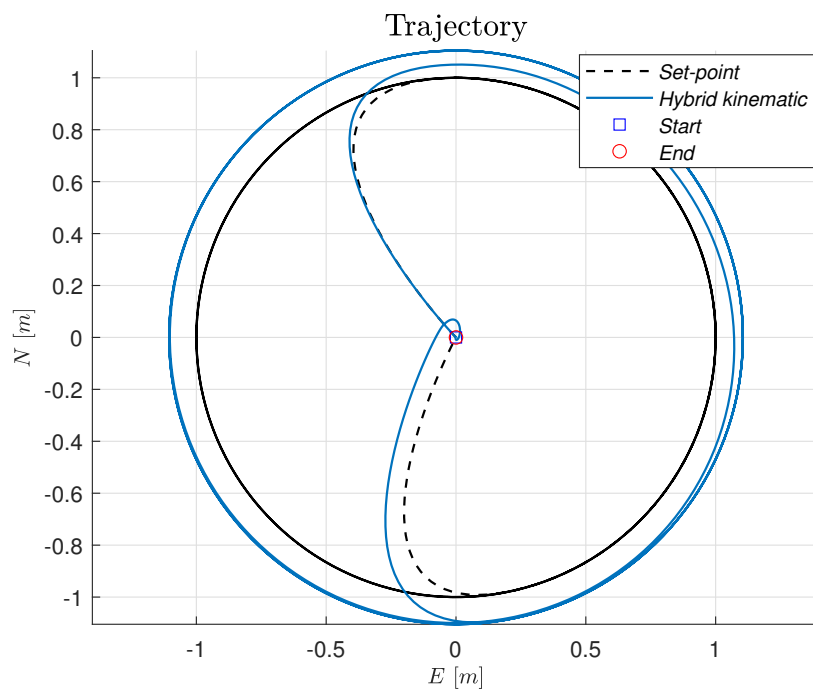


Figure 3.27: ANT-X simulator with hybrid kinematic controller. Circular trajectory.

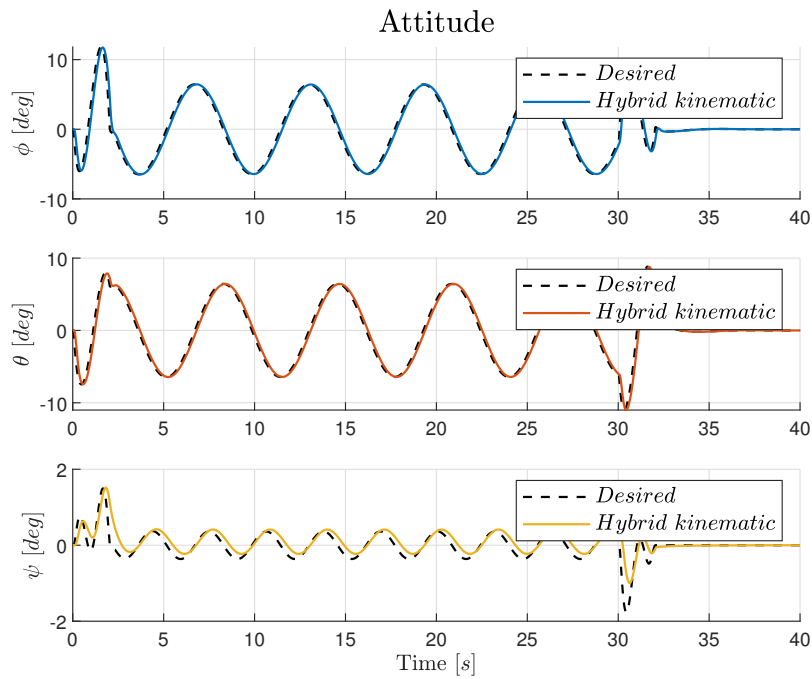


Figure 3.28: ANT-X simulator with hybrid kinematic controller. Circular trajectory: attitude.

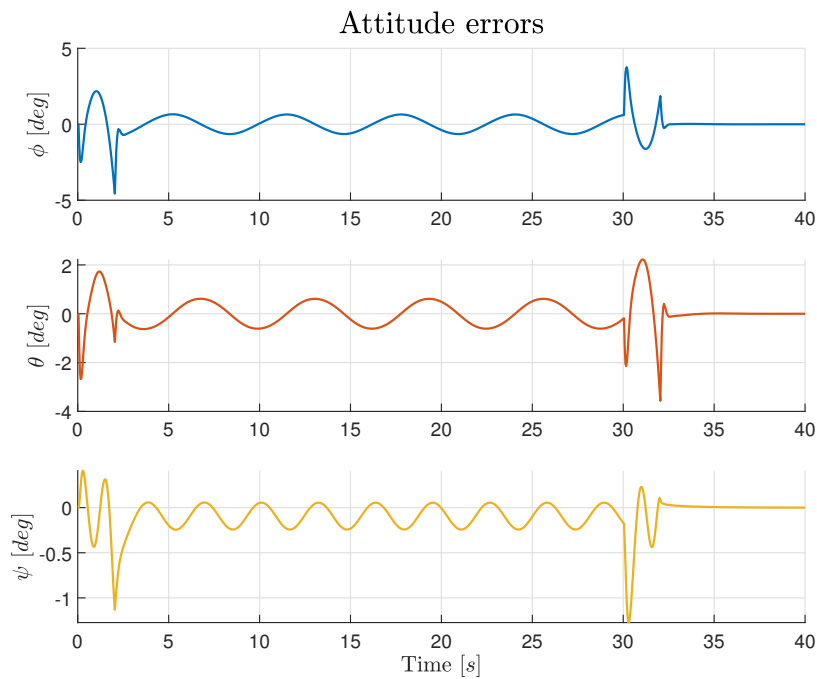


Figure 3.29: ANT-X simulator with hybrid kinematic controller. Circular trajectory: attitude errors.

3.3.2 Hybrid PID controller

As for the hybrid kinematic regulator, also the controller described in Section 3.2.2 has been implemented within the ANT-X simulator to verify its behavior in the complete simulation environment. The conditions for each simulation are the same described in Section 3.3.1.

Position and attitude set-point

As in the simulation with the hybrid kinematic regulator, a unitary position step on the East component is commanded to the drone. Figures 3.30 show that the controller stabilizes the desired position with an error that goes to zero after the initial transient (see Figure 3.31). As illustrated in Figure 3.32 the actual attitude tracks the desired one, output of the outer position and velocity control-loop, needed to move the drone from the initial to the final position. Figure 3.40 shows the attitude error.

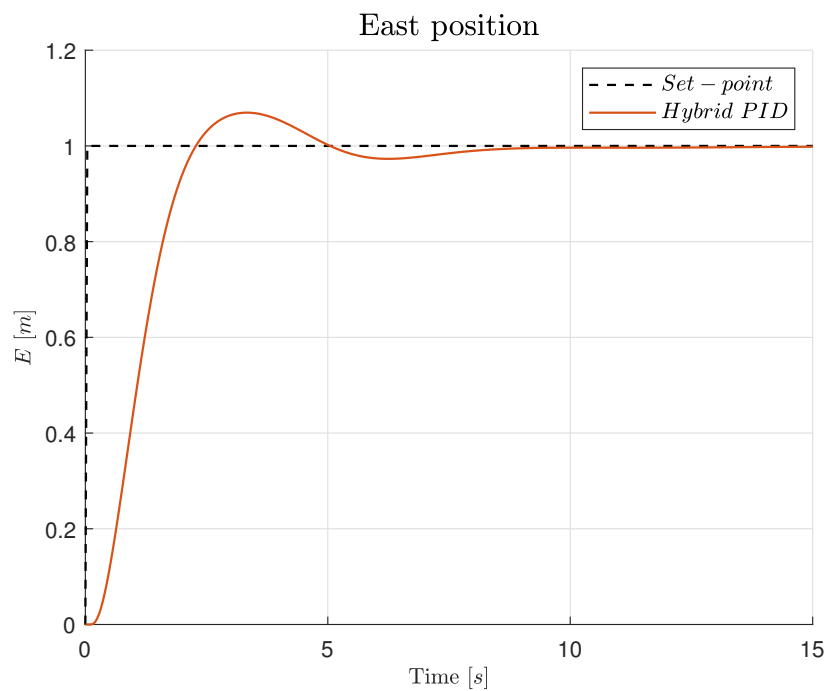


Figure 3.30: ANT-X simulator with Hybrid PID controller. Position set-point: East position.

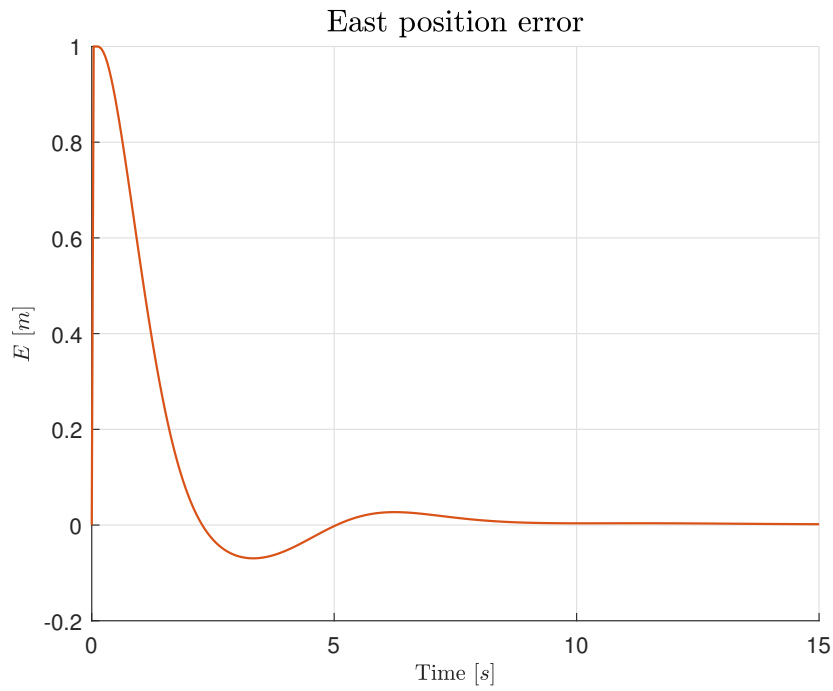


Figure 3.31: ANT-X simulator with hybrid PID controller. Position set-point: East position error.

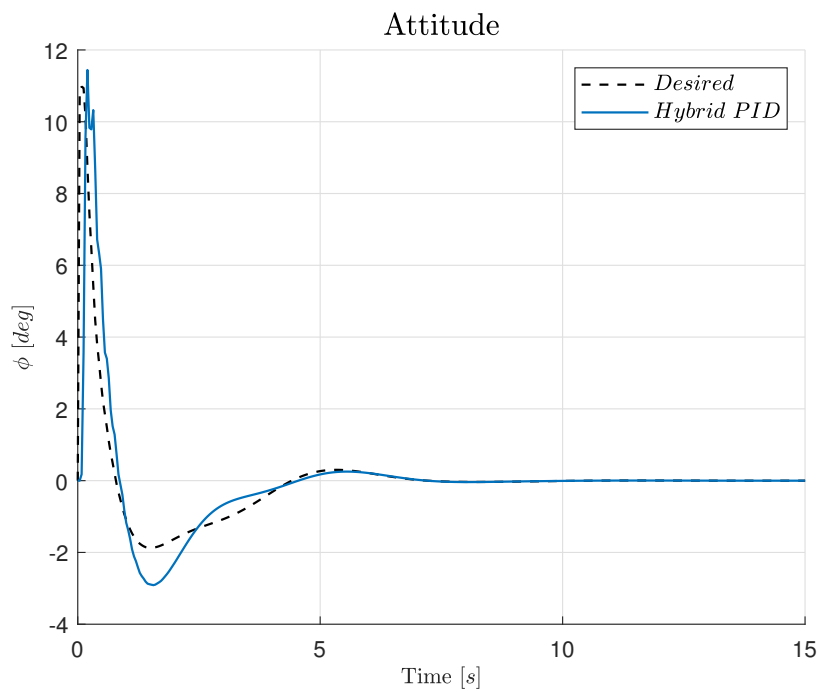


Figure 3.32: ANT-X simulator with hybrid PID controller. Position set-point: roll attitude.

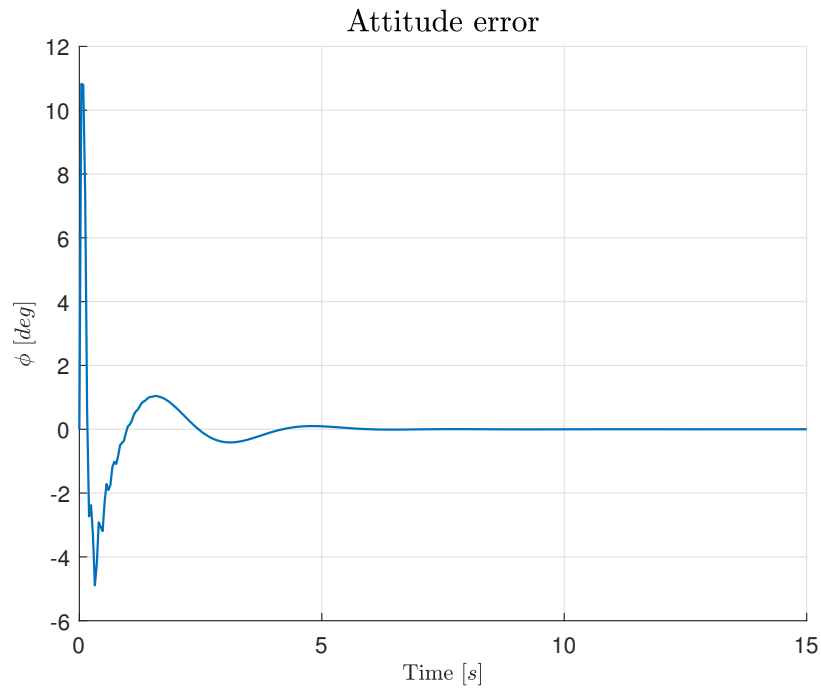


Figure 3.33: ANT-X simulator with hybrid PID controller. Position set-point: roll attitude error.

Attitude set-point As simulated with the hybrid kinematic controller, also in this case an attitude set-point of 90° with respect to the yaw axis is commanded. Figure 3.34 show the response of the system that correctly stabilize the desired attitude with an error that goes to zero after the initial transient (see Figure 3.35).

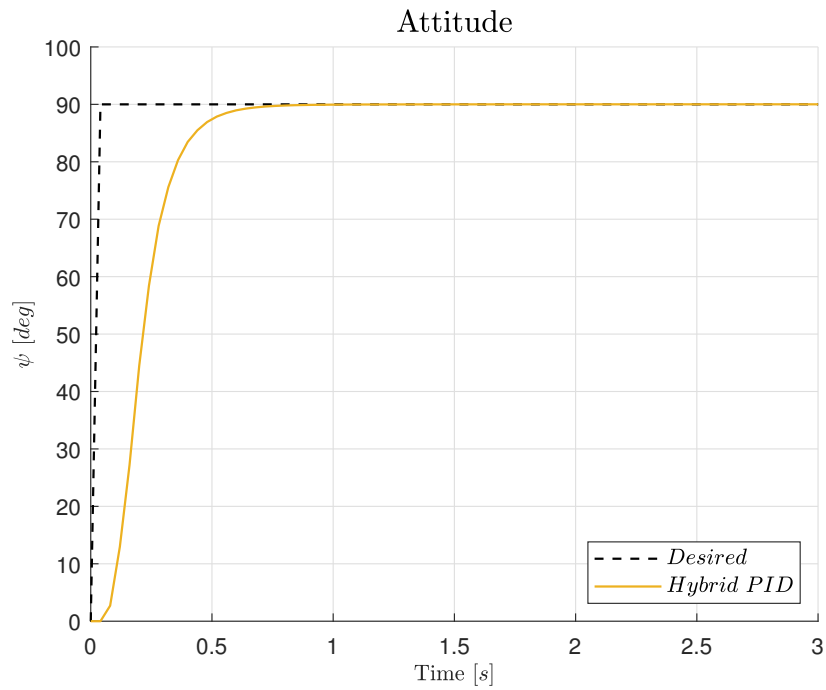


Figure 3.34: ANT-X simulator with hybrid PID controller. Attitude set-point: yaw attitude.

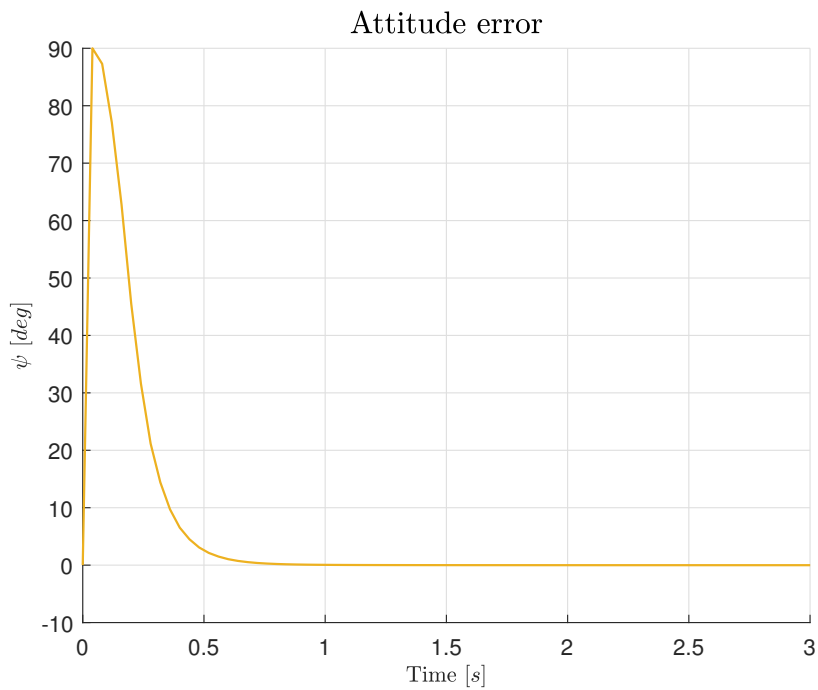


Figure 3.35: ANT-X simulator with hybrid PID controller. Attitude set-point: yaw attitude error.

Circular trajectory

As described also in the previous section, a circular trajectory is commanded to the drone. The desired trajectory has radius $R = 1 \text{ m}$ and a frequency $\omega = 1 \text{ rad/s}$, so that the North and East component of the position are computed as $R \cos(\omega t)$ and $R \sin(\omega t)$, respectively.

Figure 3.36 show that the system correctly tracks the desired trajectory with bounded error (see Figure 3.37. The E-N in-plane trajectory is reported in Figure 3.38. Figure 3.39 shows that the actual attitude tracks the desired one, output of the position and velocity control-loop, which is required to run the circular trajectory. Figure 3.40) reports the attitude error.

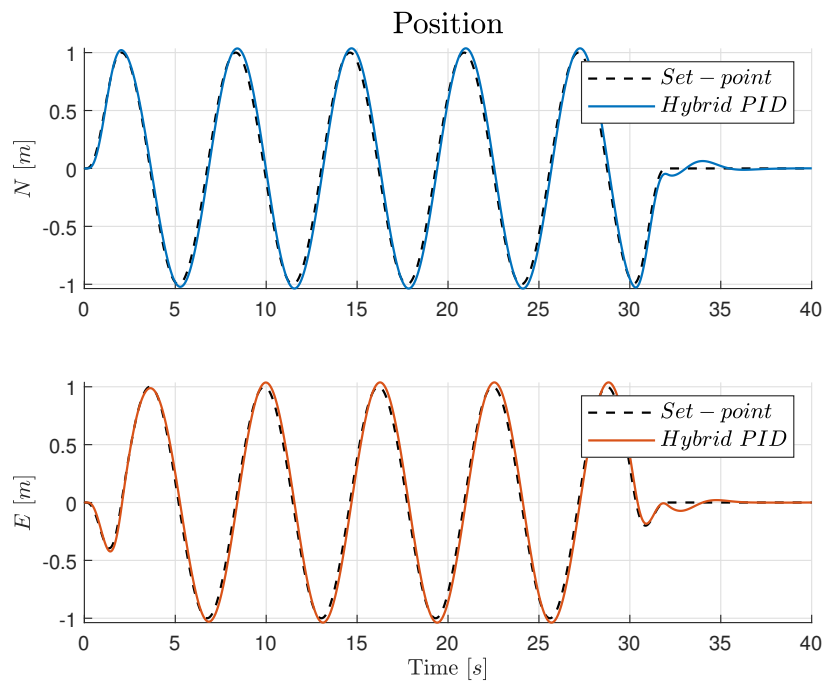


Figure 3.36: ANT-X simulator with hybrid PID controller. Circular trajectory: NE position.

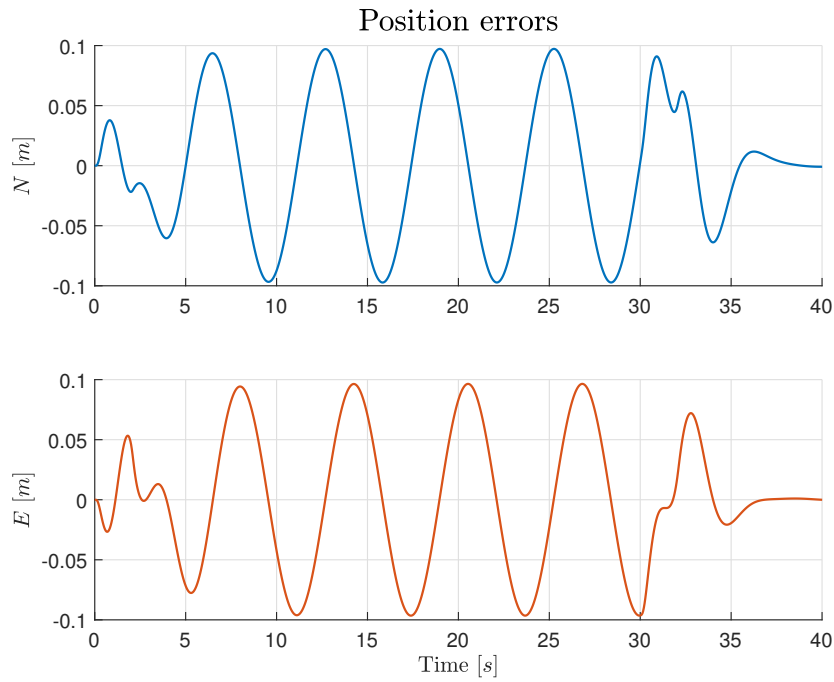


Figure 3.37: ANT-X simulator with hybrid PID controller. Circular trajectory: NE position errors.

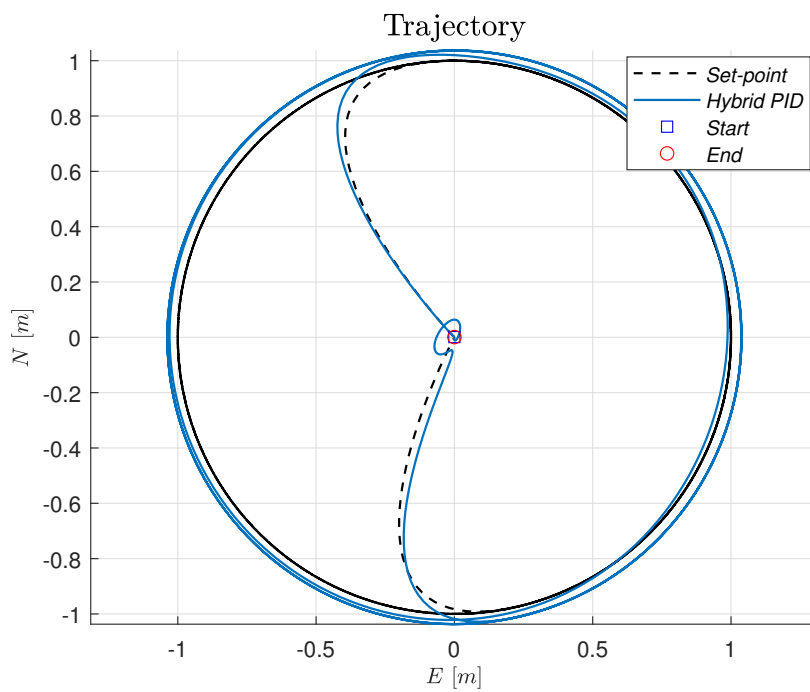


Figure 3.38: ANT-X simulator with hybrid PID controller. Circular trajectory.

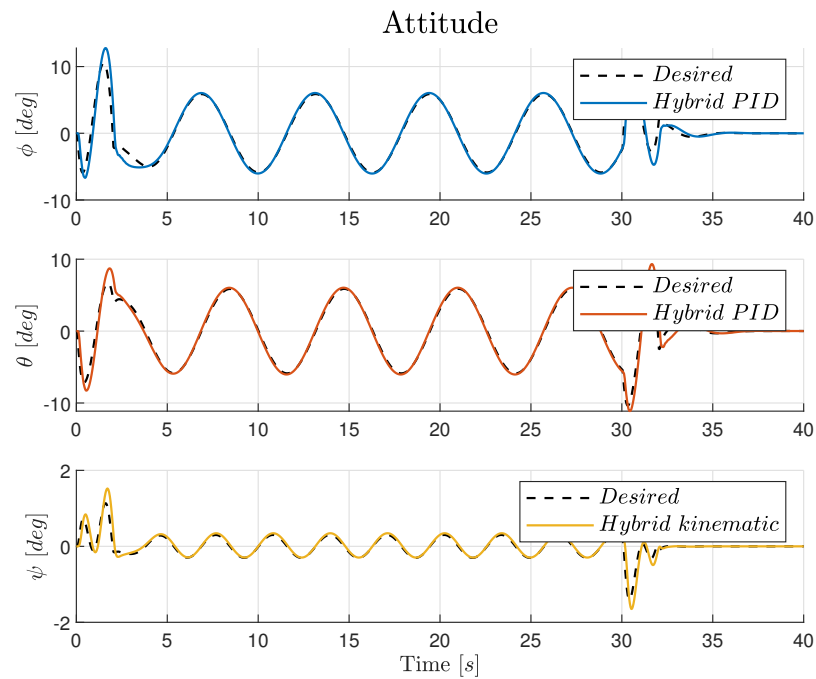


Figure 3.39: ANT-X simulator with hybrid PID controller. Circular trajectory: attitude.

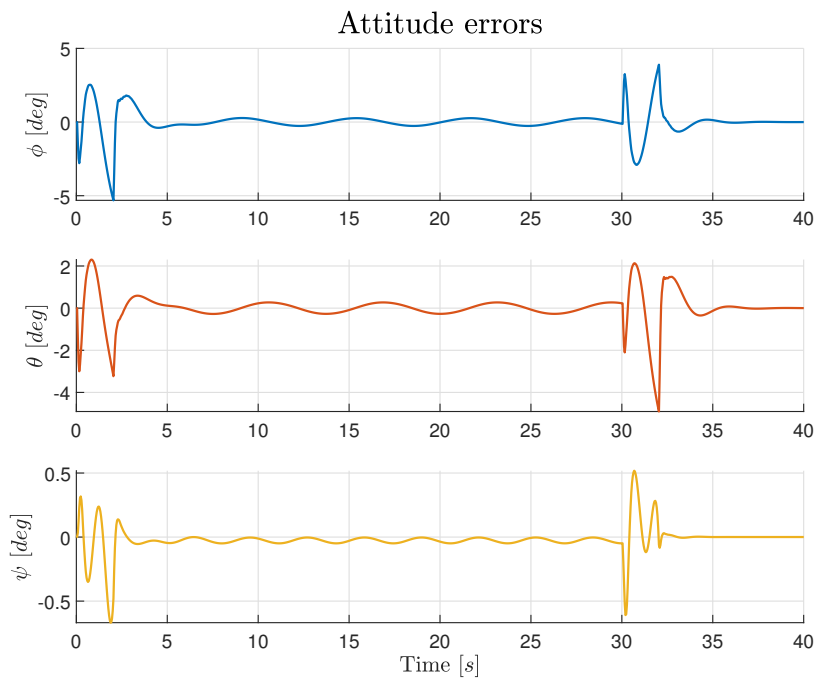


Figure 3.40: ANT-X simulator with hybrid PID controller. Circular trajectory: attitude errors.

Chapter 4

Experimental Results

This chapter opens with an overview of the experimental setup (hardware and software). In the second part, the experimental results of the flight tests performed inside the Flying Arena for Rotorcraft Technologies (Fly-ART) of the Aerospace Systems and Control Laboratory [7] of Politecnico di Milano are reported.

4.1 System architecture

This section aims to describe the hardware and software components of the system architecture used for flight session. The attitude controllers implemented onboard the UAV have been tested inside (Fly-Art) which is an indoor facility with flight volume of $6 \times 12 \times 4$ m, equipped with a motion capture system (Optitrack). Hereinafter, the Flight Control Unit (FCU) and the companion computer used by the UAV, the motion capture system (Mo-Cap), and the Ground Control Station (GCS) are presented.

4.1.1 Flight Control Unit

The Flight Control Unit (FCU) is the part dedicated to control simultaneously the revolutions per minute (RPM) of each motor of the UAV to stabilize it. ANT-X runs the PX4 firmware [32] on top of a Real Time Operating System (RTOS). Serial communication between FCU and companion relies on the MAVLink protocol [33].

4.1.2 Companion computer

The main function of the companion computer is to interface and communicate with PX4 on the FCU. During the flight test in the closed arena, it receives:

- the position from the Ground Control Station, connected to the Mo-Cap system;

Feature	ANT-X
Weight	240 <i>g</i>
Size	20 × 20 × 4 <i>cm</i>
Arm	16 <i>cm</i>
Frame	Carbon fiber
Battery	LiPo 950 <i>mAh</i> 3S
Battery weight	70 <i>g</i>

Table 4.1: ANT-X characteristics.

- the commands coming from the Ground Control Station.

These information are elaborated and sent to the FCU via the serial communication.

4.1.3 Drone

ANT-X [28] (see Figure 4.1) is the drone employed for the flight test activity. It is a quadrotor UAV designed for research and educational purposes. The attitude and position control laws run onboard the FCU and can be customized by the user by means of the SLXtoPX4 software tool, which allows to implement controllers at high level in Simulink, and automatically generate the controller code integrating it with the firmware. Communications via Wi-Fi with the ground control station are allowed thanks to the Flight Control Computer (FCC) running a Linux distribution, which provides high-level computational capabilities. Table 4.1 summarize ANT-X characteristics, where arm indicates the distance between the motors along the diagonal.

4.1.4 Motion Capture system (Mo-Cap)

In order to track and define the position of the drone in the space, the UAV mounts markers sensitive to infrared light to be detected by the Motion Capture system. The Mo-Cap is composed of 12 Infra-Red (IR) sensitive Optitrack cameras [34] which incorporate IR flood lights. The cameras are fixed at calibrated positions and orientations so that the measurement subject is into the field of view of multiple cameras. The motion capture system is controlled by the Motive software installed on the ground station which not only allows the user to calibrate the system, but it also provides interfaces for capturing and processing 3D data. The available wide range of frequency (30 - 240 *Hz*), from which the user can pick a frequency, influences the accuracy of the estimated position of the UAV.



Figure 4.1: ANT-X.



Figure 4.2: Optitrack IR camera and IR markers.

4.1.5 Ground Control Station

The Ground Control Station architecture is composed of two different OS's: Windows 10, in which Motive is installed, and Linux OS that is used to execute ROS [35] and MATLAB [36]. The GCS has two main functionalities: to provide the attitude and the position measured by the Mo-Cap system (at a frequency of 100 Hz) and to send the position and heading trajectories to the UAV using dedicated MATLAB functions.

4.2 Flight test results

This section presents the results obtained during the flight testing phase. In particular, two maneuvers have been tested: a position and attitude set-point and a circular trajectory.

4.2.1 Discontinuous controller

This regulator implements the discontinuous logic discussed in Section 1.4.3 and it is characterized by the same control parameters of the controller described in Section 3.1.2.

Position set-point

In this test, a position set-point is commanded to the drone, in particular, a unitary step in the East component.

Consider as initial conditions the position $(0, -1, -1.5)$ m expressed in the NED reference frame, null attitude, and null angular rates. The final conditions are set to be the position $(0, 0, -1.5)$ m and null attitude.

Figure 4.3 shows that the system correctly stabilizes the commanded position with an error that goes to zero after a brief transient (see Figure 4.4). In Figure 4.5 it is possible to appreciate how the desired attitude, computed by the outer position and velocity control loop, evolves and how the actual attitude tracks it. Figure 4.6 illustrates the error between desired and actual attitude.

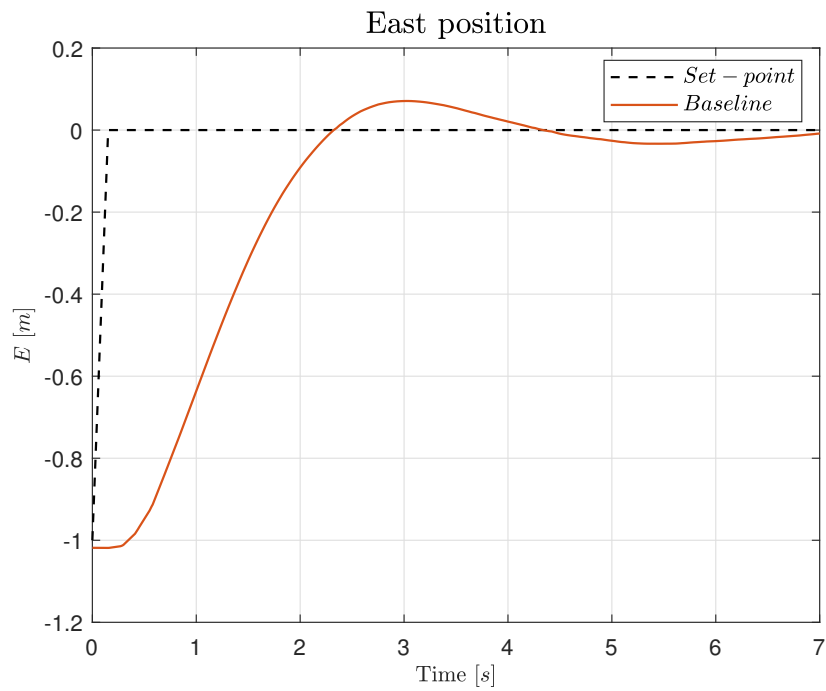


Figure 4.3: Experimental position set-point. Discontinuous controller. East position.

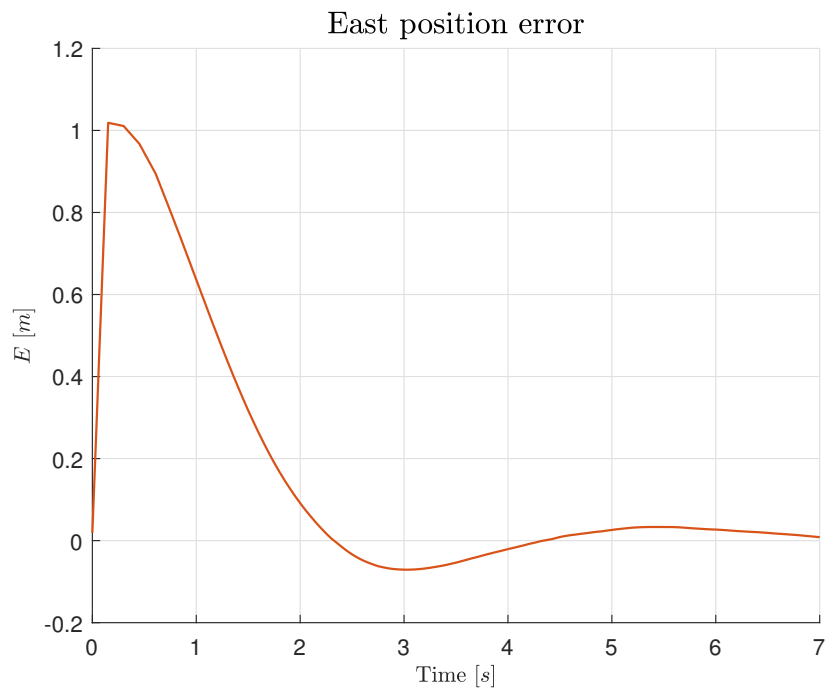


Figure 4.4: Experimental position set-point. Discontinuous controller. East position errors.

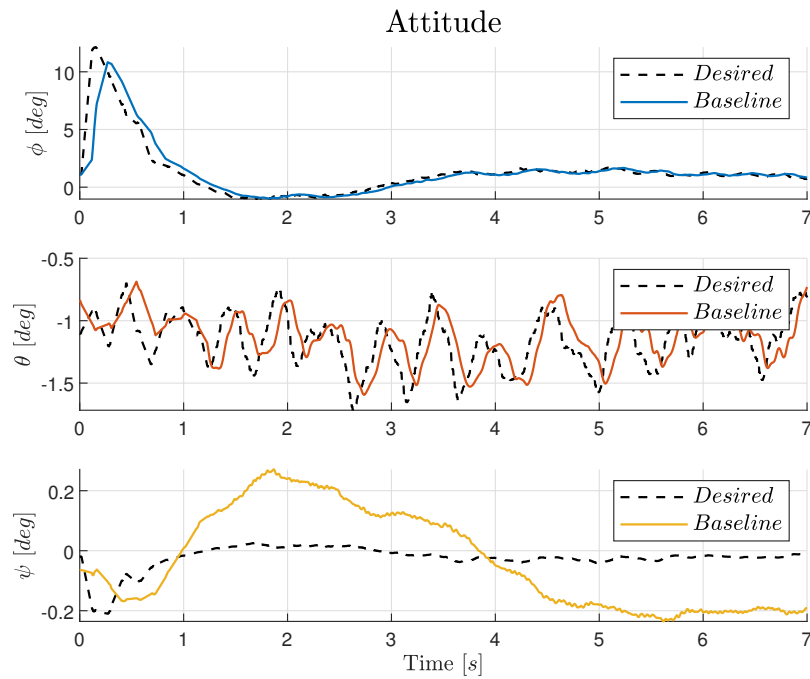


Figure 4.5: Experimental position set-point. Discontinuous controller. Attitude.

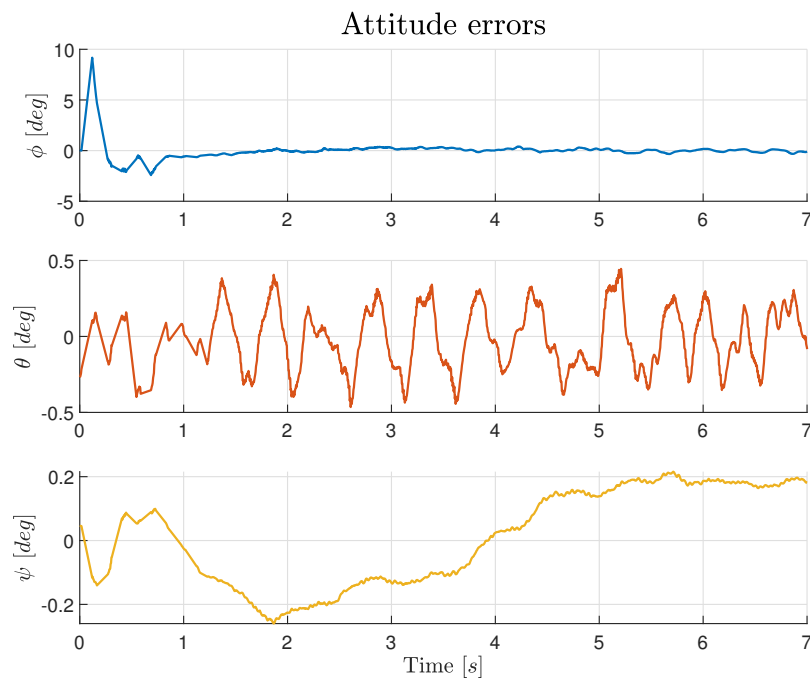


Figure 4.6: Experimental position set-point. Discontinuous controller. Attitude errors.

Circular trajectory

In this test, a circular trajectory is commanded. In particular, it is characterized by a radius $R = 1 \text{ m}$ and a frequency $\omega = 1 \text{ rad/s}$, so that the North and East position components are computed as $R \cos(\omega t)$ and $R \sin(\omega t)$, respectively.

Consider as initial conditions the position $(1, 0, -1.5) \text{ m}$ expressed in the NED reference frame, null attitude, and null angular rates. After one circle and a half, the drone has to reach the final conditions set to be the center of the circle, namely the position $(0, 0, -1.5) \text{ m}$, and null attitude.

Figure 4.7 shows that the position is correctly tracked with errors (see Figure 4.8) caused by the delay due to the lack of feed-forward of velocity and acceleration. In Figure 4.7 it is possible to note a vertical segment of the East component of the set-point around 15 s: this is caused by the function that commands the end set-point position to the drone and that computes the trajectory starting from the actual position. Figure 4.11 illustrates that the attitude successfully tracks the desired attitude, output of the outer loop, necessary to make the drone run the circular trajectory. In Figure 4.12 the error between desired and actual attitude is reported. Figures 4.9 and 4.10 illustrate the trajectory in the $E-N$ plane and the 3D trajectory, respectively: the actual trajectory differs from the desired one only during the entry phase due to a non sufficiently high velocity at the beginning of the trajectory, and during the exit maneuver due to the inertia of the drone that is not braking enough to stop and change its position.

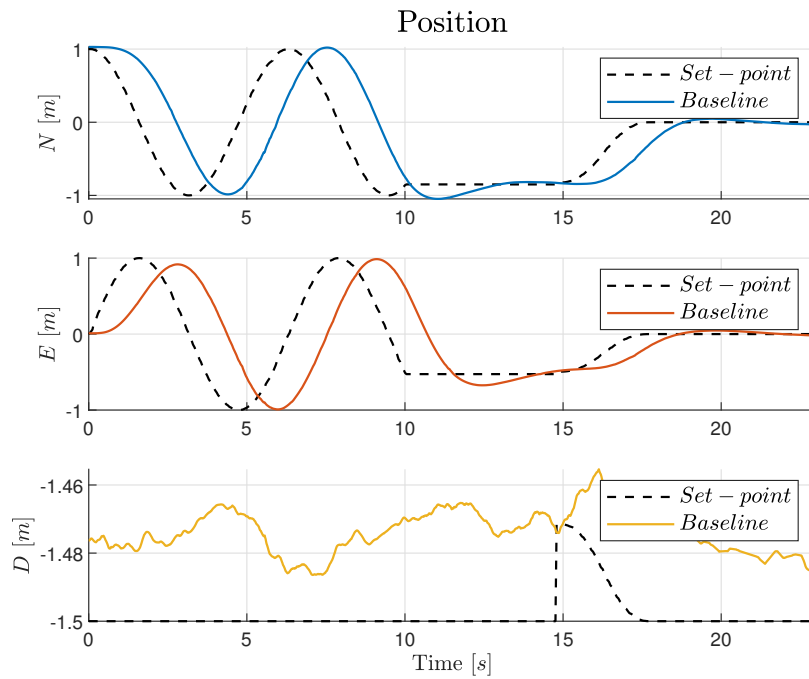


Figure 4.7: Experimental circular trajectory. Discontinuous controller. NED position.

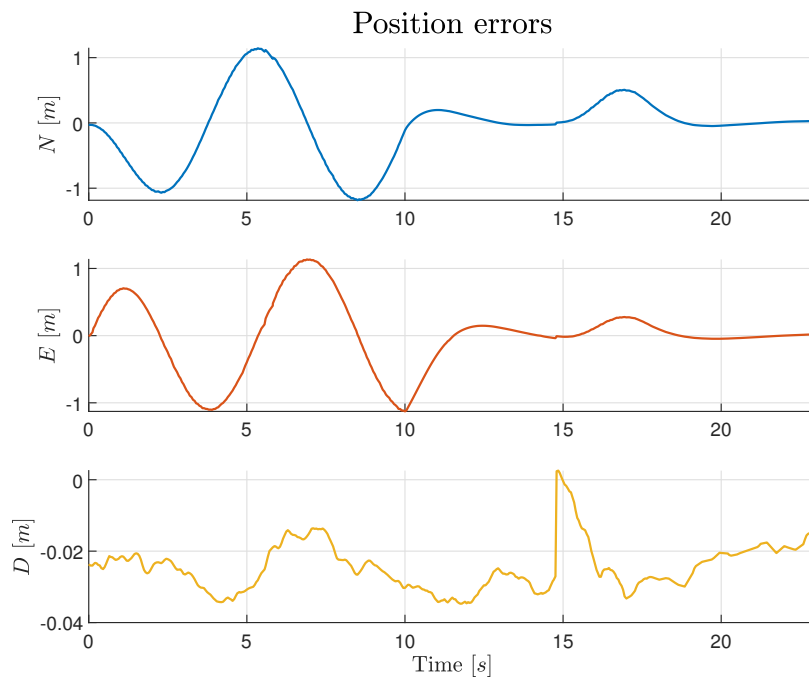


Figure 4.8: Experimental circular trajectory. Discontinuous controller. NED position errors.

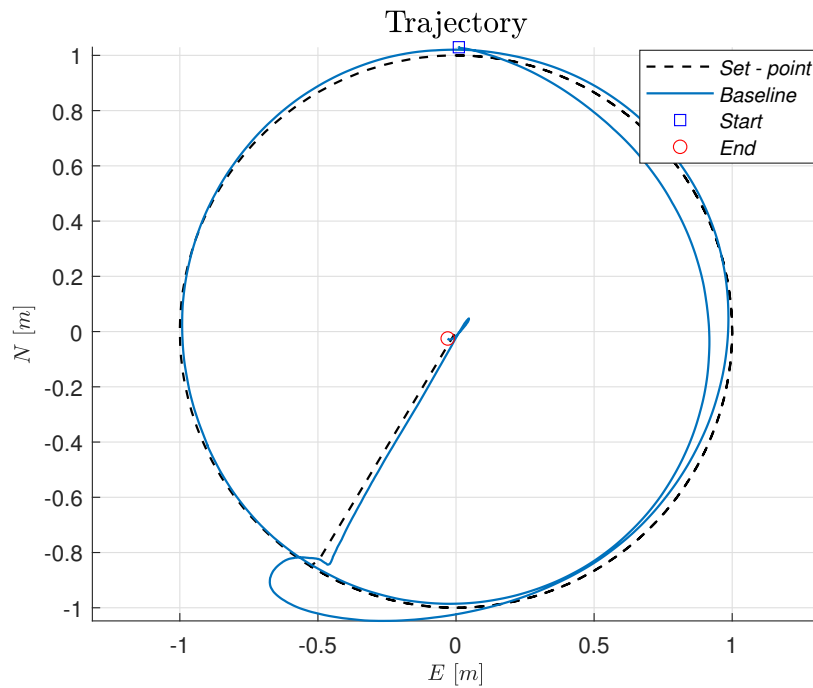


Figure 4.9: Experimental circular trajectory. Discontinuous controller. Trajectory.

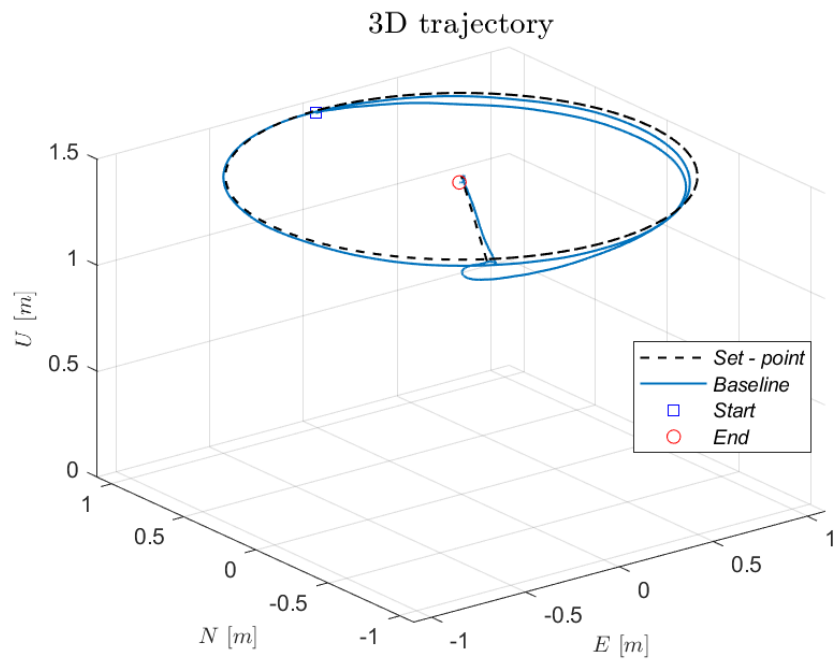


Figure 4.10: Experimental circular trajectory. Discontinuous controller. 3D trajectory.

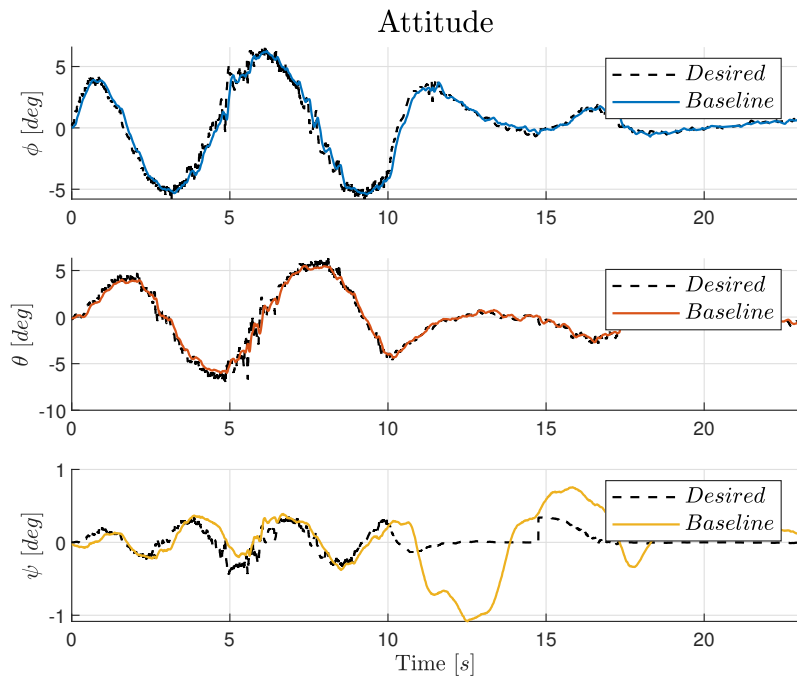


Figure 4.11: Experimental circular trajectory. Discontinuous controller. Attitude.

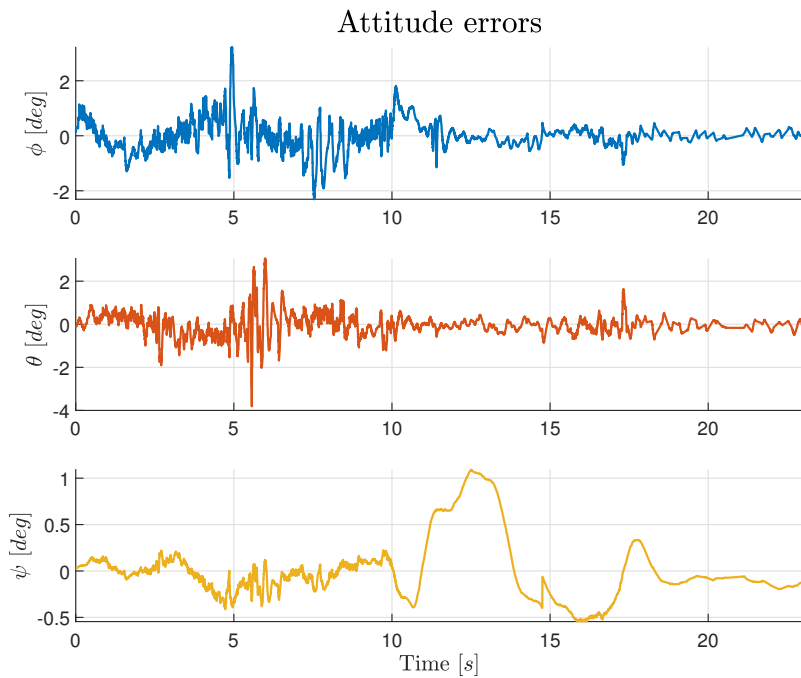


Figure 4.12: Experimental circular trajectory. Discontinuous controller. Attitude errors.

4.2.2 Hybrid kinematic controller

This regulator implements the hybrid kinematic logic discussed in Section 2.4.1 and it is characterized by the same control parameters of the controller described in Section 3.1.2. The conditions for each test are the same as described in Section 4.2.1.

Position and attitude set-point

Figure 4.13 shows the correct stabilization of the step response with an error (see Figure 4.14) which goes to zero after a brief transient. Once again a comparison between the actual and the desired attitude computed by the outer loop, is available in Figure 4.15: the actual attitude tracks the desired with good margin, as confirmed by the attitude error presented in Figure 4.16.

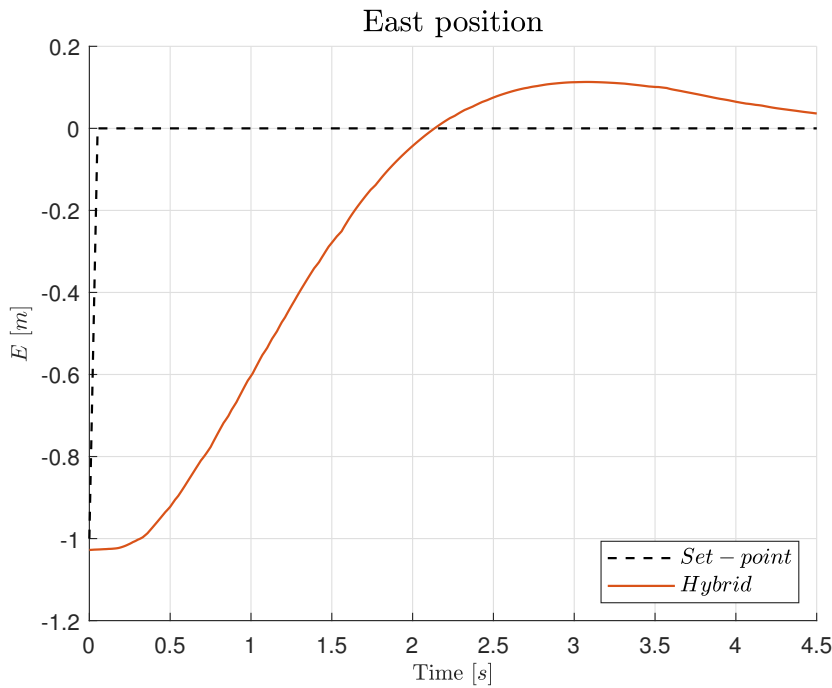


Figure 4.13: Experimental position set-point. Hybrid kinematic controller. East position.

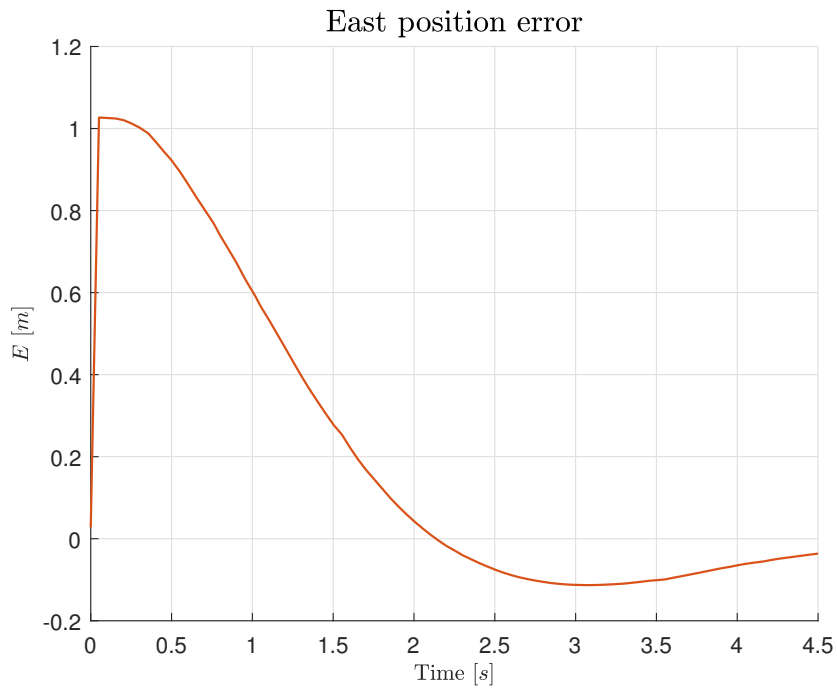


Figure 4.14: Experimental position set-point. Hybrid kinematic controller. East position error.

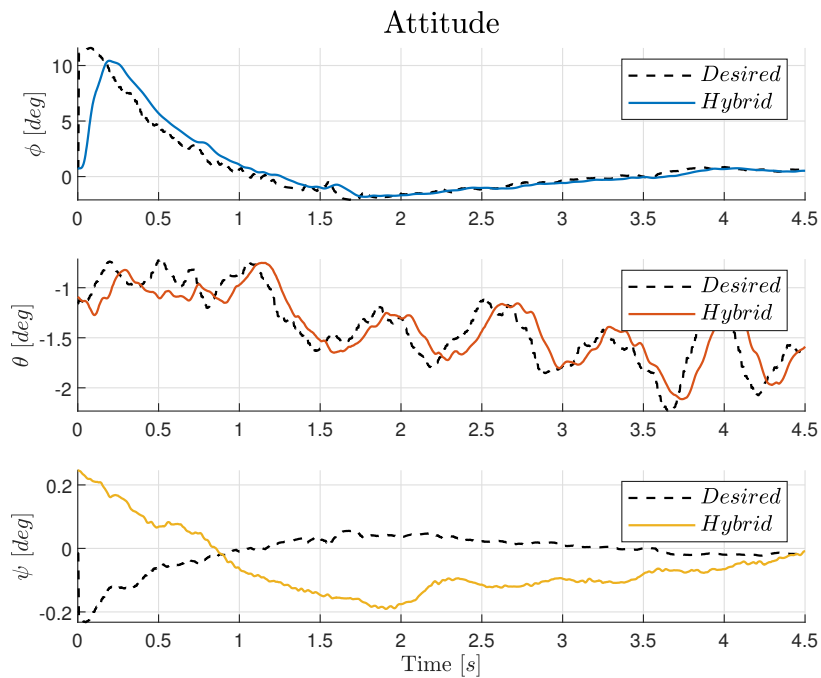


Figure 4.15: Experimental position set-point. Hybrid kinematic controller. Attitude.

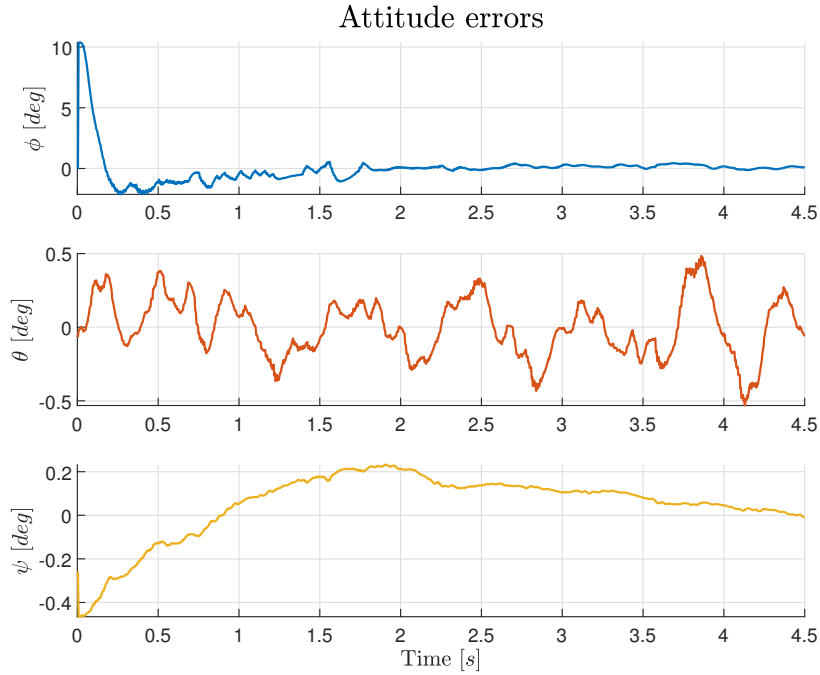


Figure 4.16: Experimental position set-point. Hybrid kinematic controller. Attitude errors.

Circular trajectory

The same circular trajectory considered for the system with the baseline controller has been commanded to the system with the hybrid kinematic controller. The circular trajectory has radius $R = 1 \text{ m}$ and angular velocity $\omega = 1 \text{ rad/s}$, so that the North and East component of the position are computed as $R \cos(\omega t)$ and $R \sin(\omega t)$, respectively.

Figure 4.17 shows that the actual position tracks the desired one, even though a delay caused by the lack of velocity and acceleration feed-forward leads to the errors reported in Figure 4.18. As in the case of the baseline controller, a vertical set-point in the East component is present due to the function that commands the final position. Figures 4.19 and 4.20 illustrate the circular trajectory in the $E-N$ plane and in 3D, respectively. As for the case of the discontinuous controller, the actual trajectory differs from the desired one during the entry and exit phases. The actual and the desired attitude, computed by the position and velocity outer loop, are shown in Figure 4.21: the real attitude successfully tracks the desired one with bounded error (see Figure 4.22).

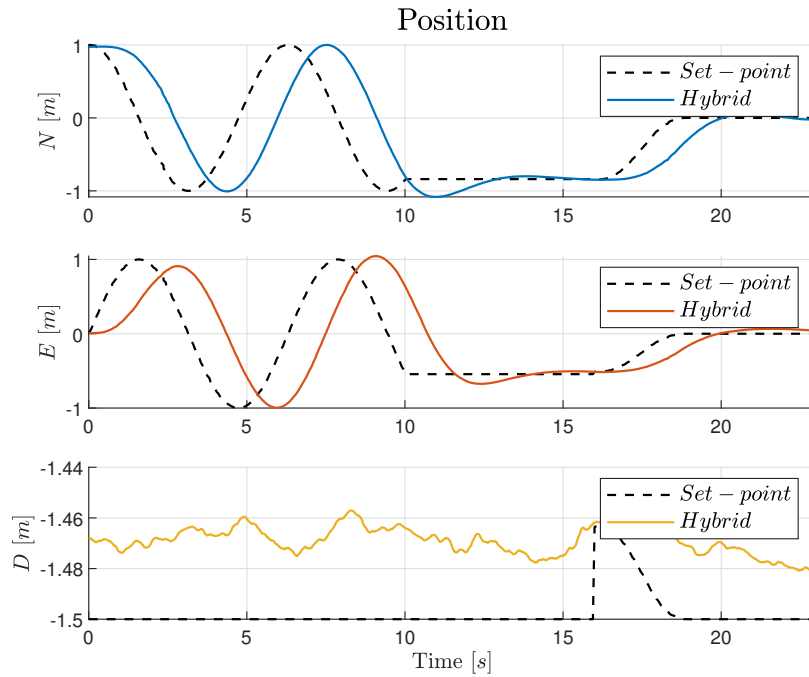


Figure 4.17: Experimental circular trajectory. Hybrid kinematic controller. NED position.

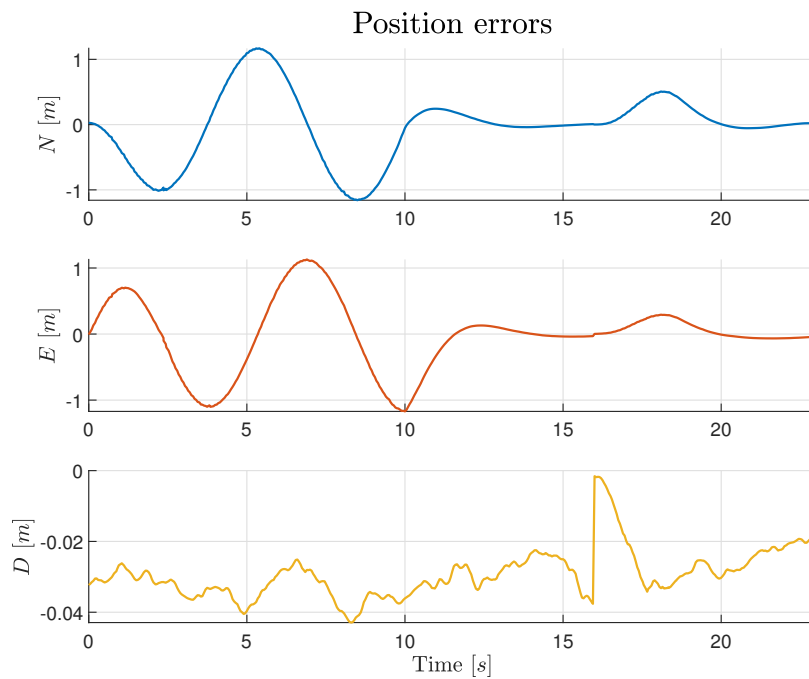


Figure 4.18: Experimental circular trajectory. Hybrid kinematic controller. NED position errors.

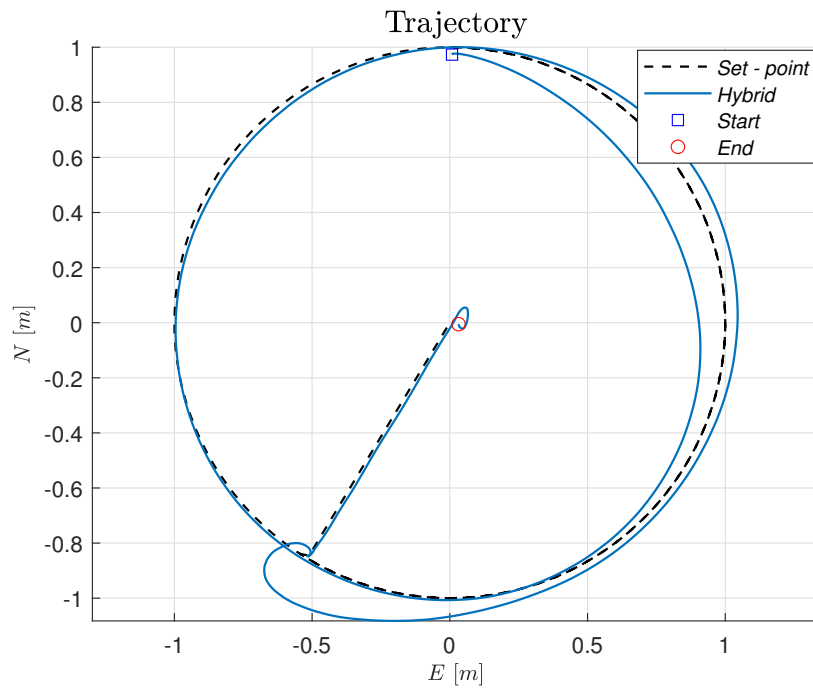


Figure 4.19: Experimental circular trajectory. Hybrid kinematic controller. Trajectory.

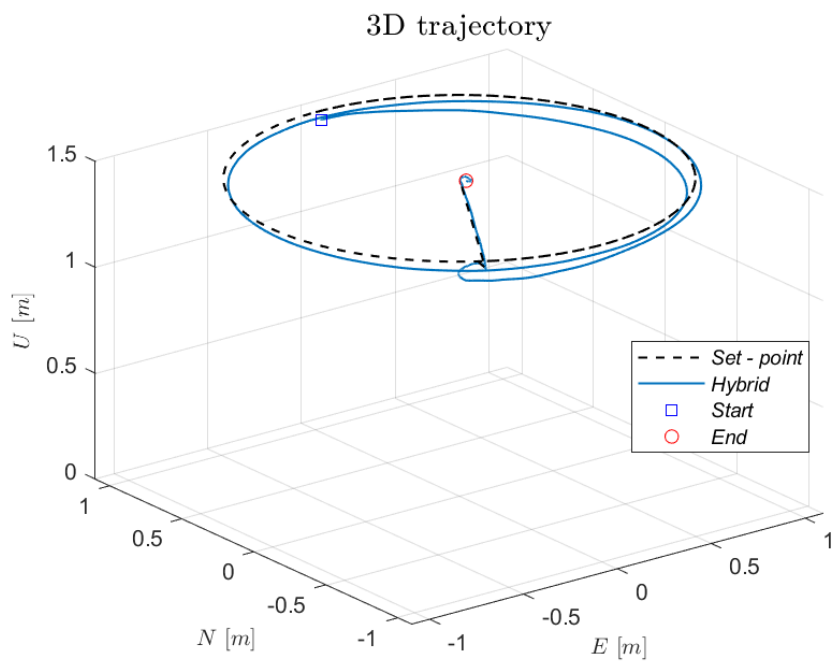


Figure 4.20: Experimental circular trajectory. Hybrid kinematic controller. 3D trajectory.

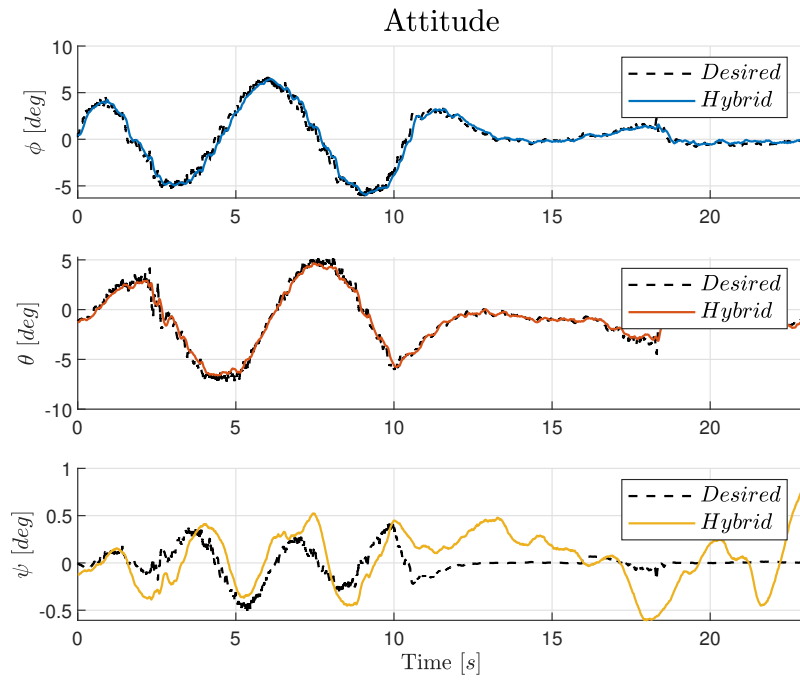


Figure 4.21: Experimental circular trajectory. Hybrid kinematic controller. Attitude.

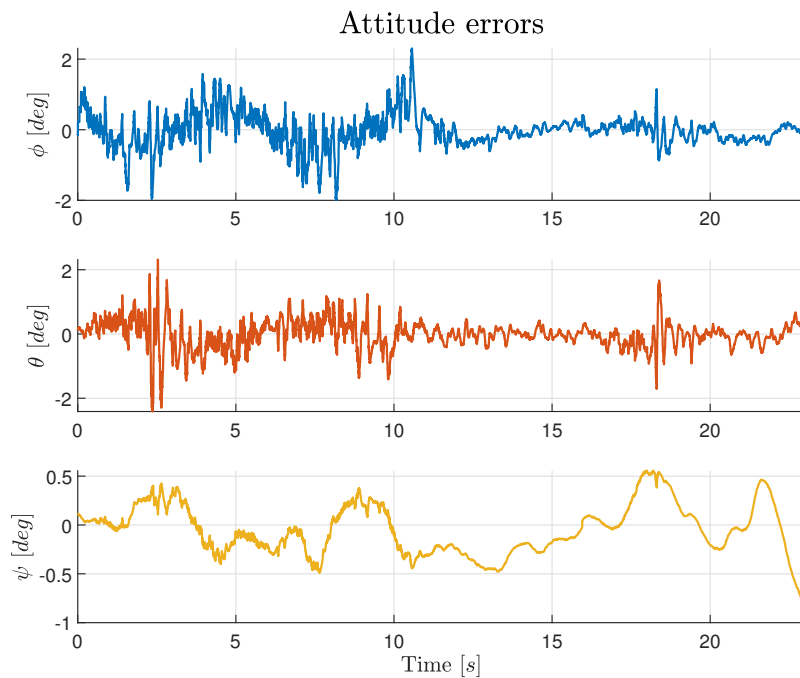


Figure 4.22: Experimental circular trajectory. Hybrid kinematic controller. Attitude errors.

4.2.3 Hybrid PID controller

This regulator implements the hybrid PID logic discussed in Section 2.4.2 and it is characterized by the same control parameters of the controller described in Section 3.1.2. The conditions for each test are the same as described in Section 4.2.1.

Position and attitude set-point

Figure 4.23 shows the position response to the step input. The desired position is correctly stabilized and the errors are reported in Figure 4.24. Figure 4.25 illustrates that the actual attitude tracks the desired one with an error that goes to zero after a brief transient (see Figure 4.26).

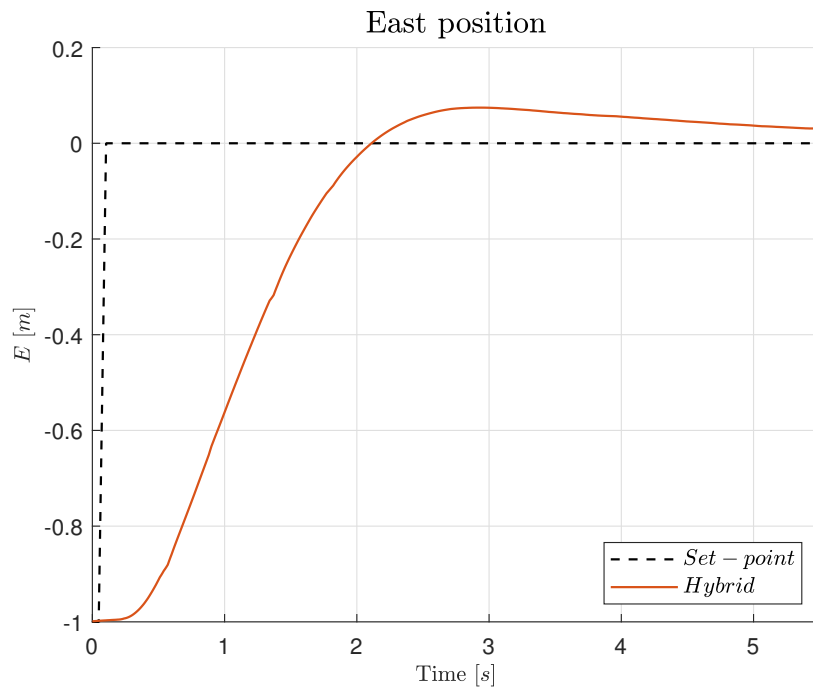


Figure 4.23: Experimental position set-point. Hybrid PID controller. East position.

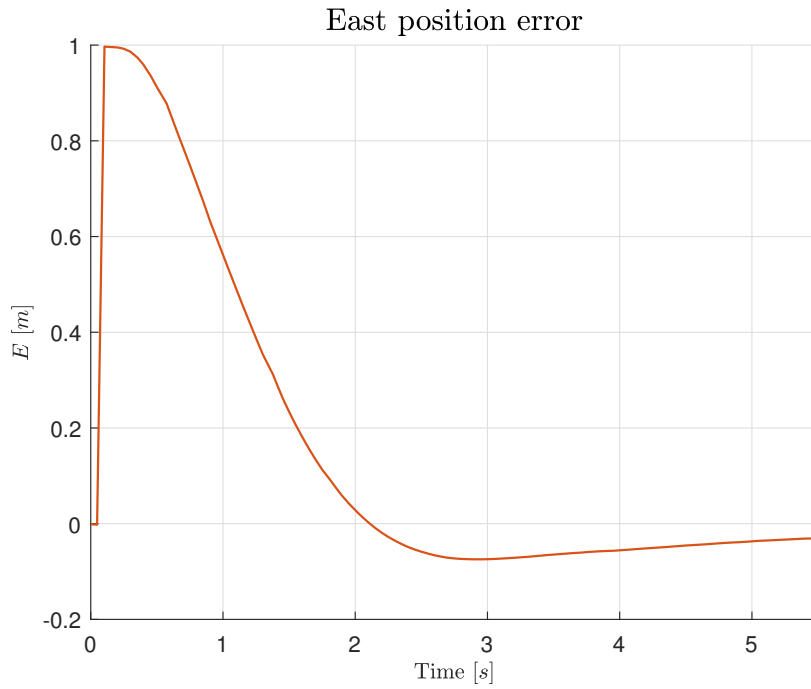


Figure 4.24: Experimental position set-point. Hybrid PID controller. East position error.

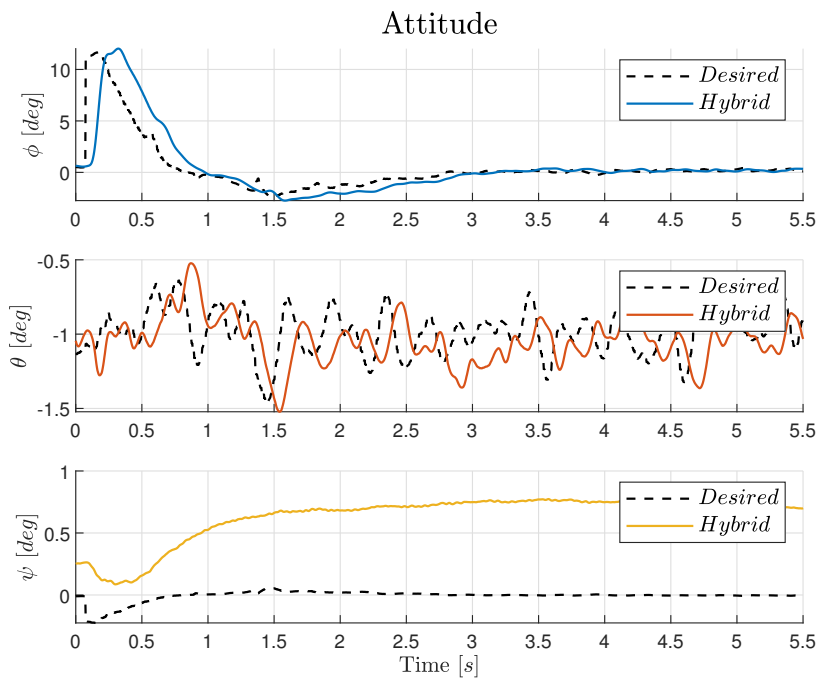


Figure 4.25: Experimental position set-point. Hybrid PID controller. Attitude.

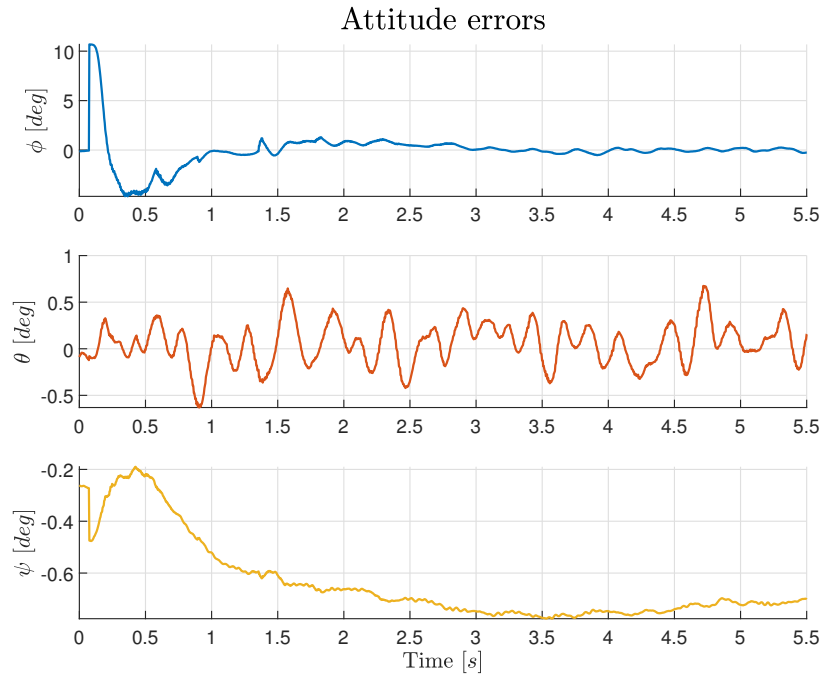


Figure 4.26: Experimental position set-point. Hybrid PID controller. Attitude errors.

Circular trajectory

Figures 4.27 shows the position response of the drone; the position errors are reported in Figure 4.28 and are mainly caused by a delay due to the lack of velocity and acceleration feed-forward. Once again, in 4.27 the vertical segment of the set-point in the East component is present due to the function that commands the end position starting from the actual one. Figures 4.29 and 4.30 illustrate the circular trajectory in the $E-N$ plane and in 3D, respectively. Note that, this controller is less performing than the hybrid kinematic and discontinuous controllers, since the trajectory mismatch is not limited to the initial and final phases only. This issue is caused by a non optimal tuning of the parameters, which was not possible to achieve in the context of this thesis due to the complex control architecture. Nonetheless, it is still able to track the desired attitude with good margin, as shown in Figures 4.31 and 4.32.

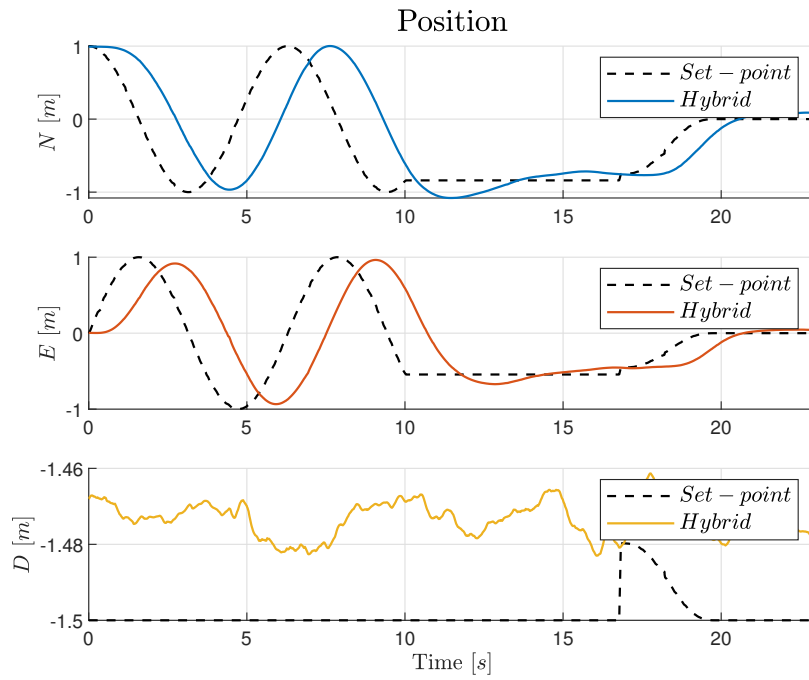


Figure 4.27: Experimental circular trajectory. Hybrid PID controller. NED position.

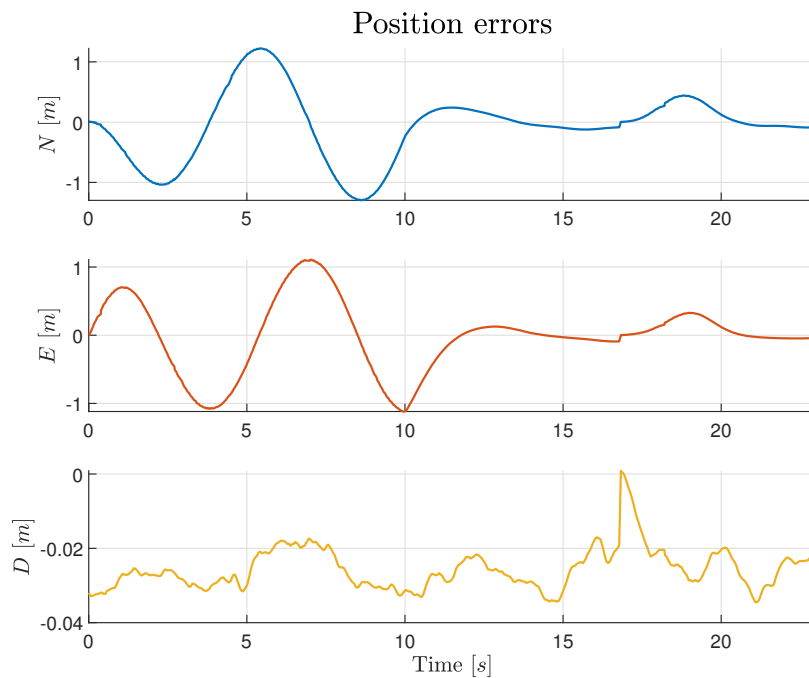


Figure 4.28: Experimental circular trajectory. Hybrid PID controller. NED position errors.

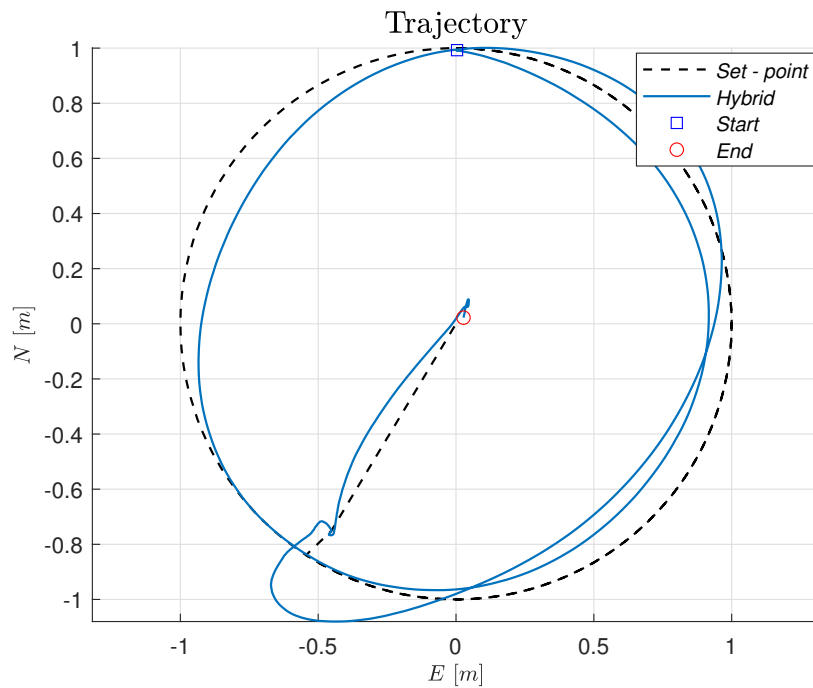


Figure 4.29: Experimental circular trajectory. Hybrid PID controller. Trajectory.

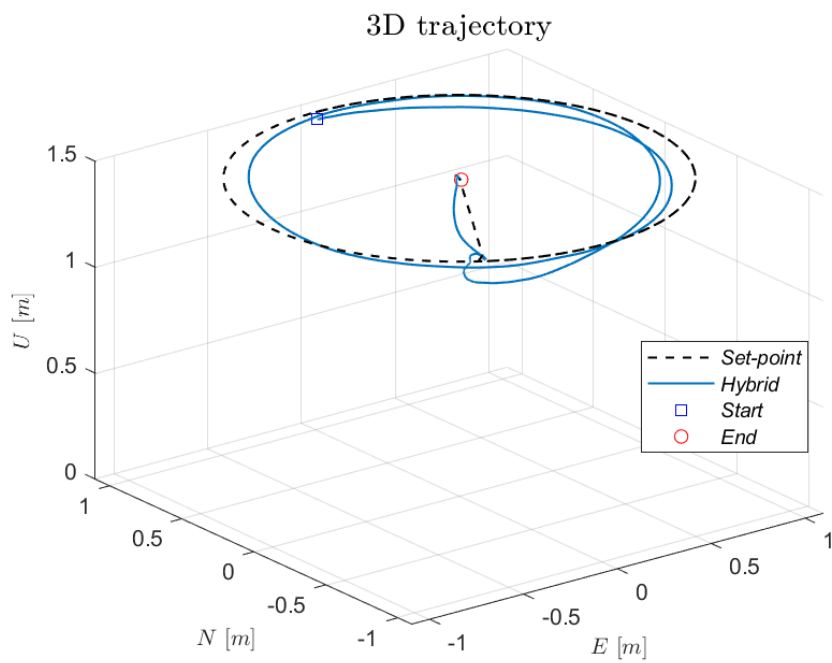


Figure 4.30: Experimental circular trajectory. Hybrid PID controller. 3D trajectory.

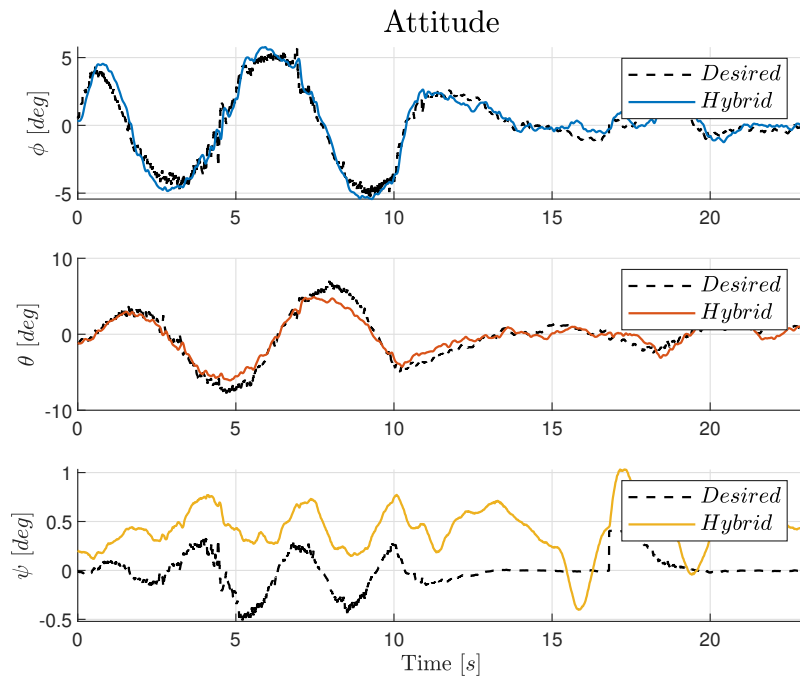


Figure 4.31: Experimental circular trajectory. Hybrid PID controller. Attitude.

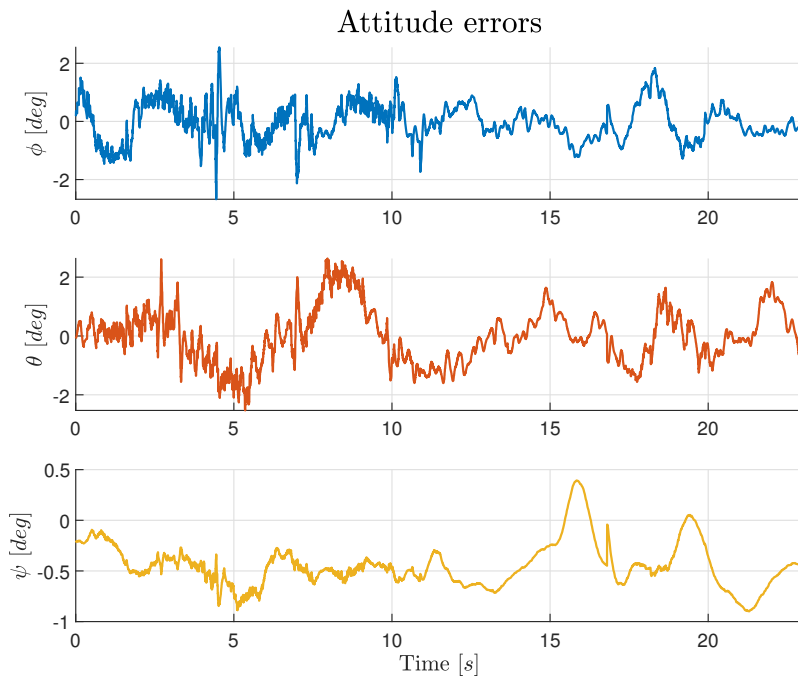


Figure 4.32: Experimental circular trajectory. Hybrid PID controller. Attitude errors.

4.3 Comparison of circular trajectories

This section presents a direct comparison of the three controllers for which the circular trajectory has been tested. In order to give a detailed view, the final position set-point has been neglected so that the comparison focuses only on the pure circular trajectory.

Figures 4.33 and 4.34 show the comparison in terms of North and East position and position errors, respectively: baseline and hybrid kinematic controllers exhibit the same behavior, whereas the hybrid PID controller introduces some delay. Figure 4.35 illustrates the attitude comparison in terms of roll and pitch angles: once again, baseline and hybrid kinematic controllers are comparable, whereas the hybrid PID controller exhibits a more aggressive behavior that prevents the correct attitude tracking causing the error in position. Note that, unlike the case of the position for which the set-point is the same for all the three controllers, the desired attitude is different for each controller and therefore a single reference cannot be considered. Figure 4.36 shows the attitude errors.

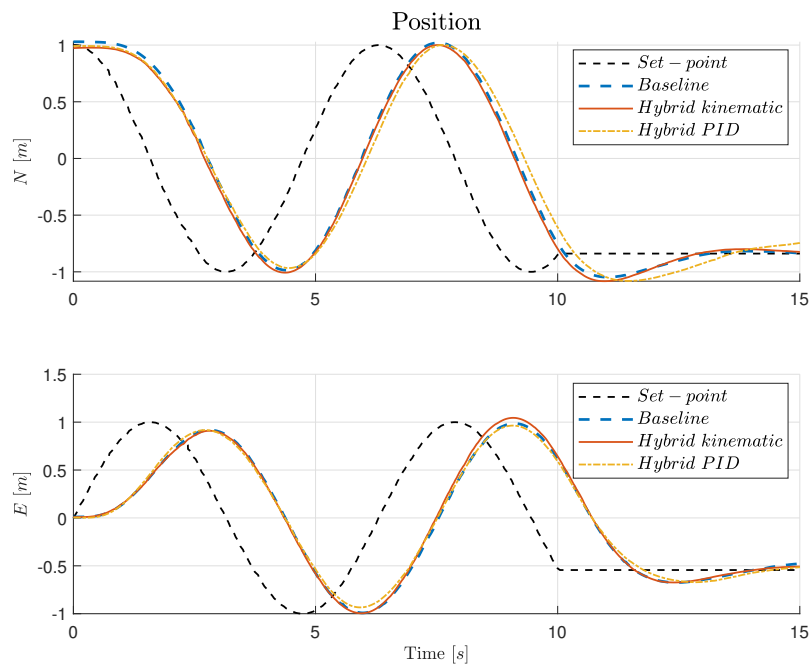


Figure 4.33: Experimental circular trajectory. Comparison of the three controllers. NE position.

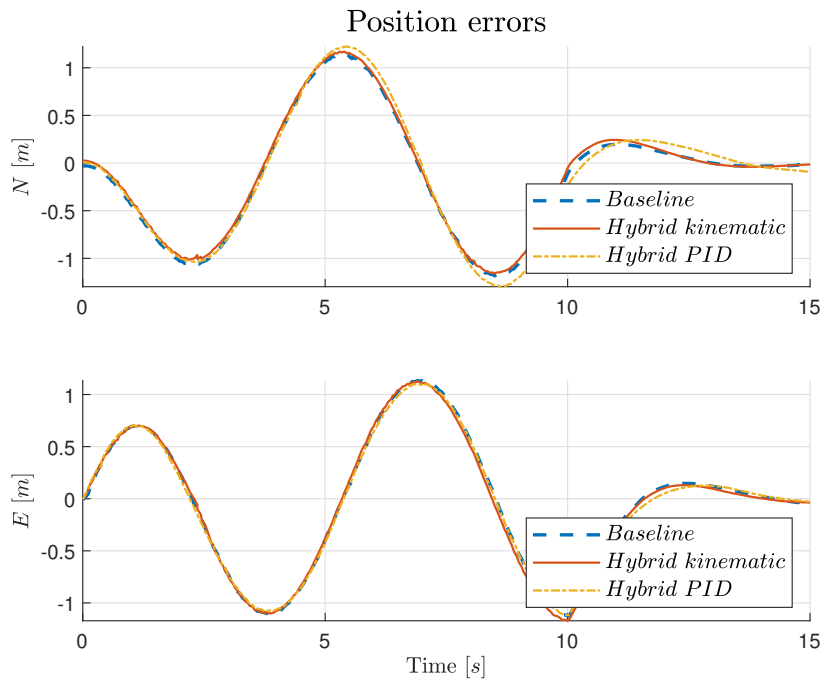


Figure 4.34: Experimental circular trajectory. Comparison of the three controllers. NE position errors.

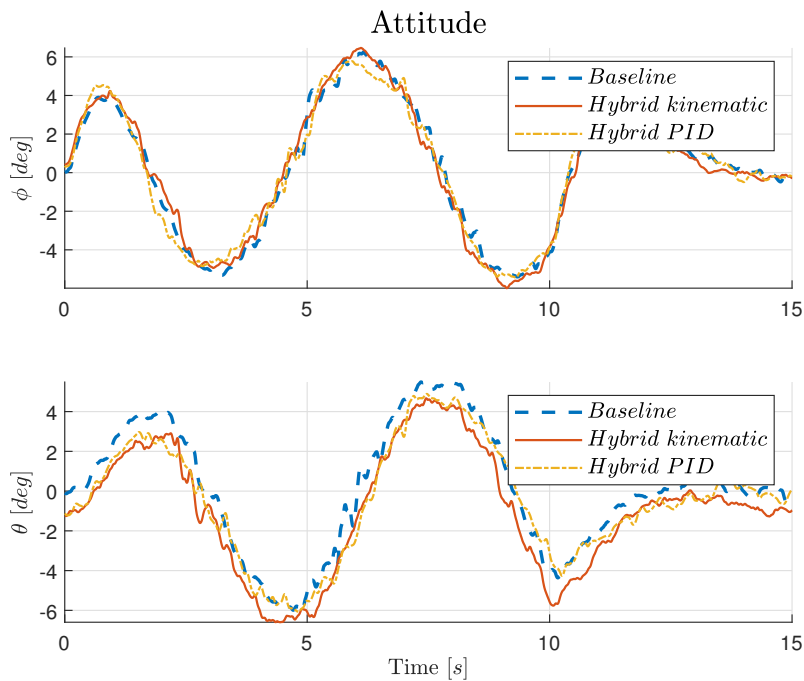


Figure 4.35: Experimental circular trajectory. Comparison of the three controllers. Attitude.

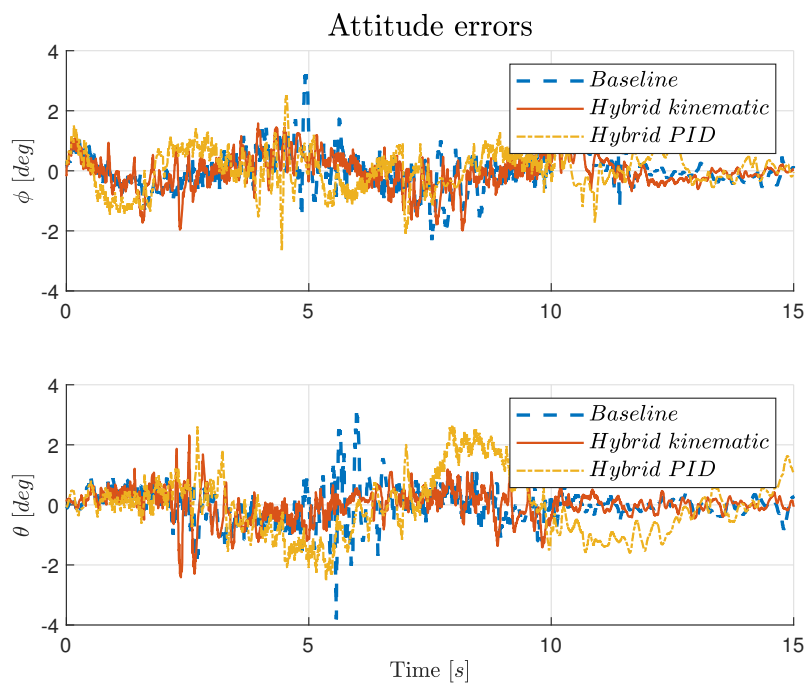


Figure 4.36: Experimental circular trajectory. Comparison of the three controllers. Attitude errors.

Conclusions

The aim of this thesis was to investigate the problems related to the attitude parametrization by use of unit-quaternions and to implement an attitude controller robust to measurement noise for a UAV.

The conducted activity is mainly inspired by the work of C. G. Mayhew et al. [3],[5],[24], who analyzed how the asymptotic stability properties of quaternion-based controllers translate to the actual rigid-body space, and proposed a quaternion-based hysteretic feedback that robustly globally asymptotically stabilizes the attitude of a rigid body. Starting from this, the purpose of this thesis was to develop and implement a hybrid attitude controller to be mounted on ANT-X, a drone employed by Aerospace Systems and Control Laboratory (ASCL) of Politecnico di Milano.

The identified dynamics of the drone have been simulated in the context of a set-point of 180° in yaw in presence of measurement noise considering two different control strategies. The first control strategy implements discontinuous control law relying on the sign of the quaternion error scalar component; in the context of this thesis, this controller is called "baseline" since it is the control law already present on ANT-X. The second control strategy implements a hybrid and hysteretic logic which allows the controller to preserve the global attractivity. The results of the simulation showed that the hybrid kinematic controller is impervious to the noise and outperforms the discontinuous strategy, which exhibited a lag in the response due to a noise-induced chattering that forced the UAV to initially oscillate around the initial condition by continuously applying positive and negative control actions.

In the context of attitude tracking, a second hybrid regulator has been introduced with a more complex control architecture, as described by J. Su [6]: unlike the hybrid kinematic controller, this regulator implements a feed-forward term proportional to the desired angular velocity to improve tracking performance.

The three controllers have been simulated in the complete ANT-X simulator in MATLAB environment for a set-point in position, a set-point in attitude and a circular trajectory.

Finally, the three controllers have been experimentally tested and validated in the Flying Arena for Rotorcraft Technologies (Fly-ART) of the Aerospace Systems and Control Laboratory (ASCL) of Politecnico di Milano. The results showed

that the hybrid kinematic controller resembled the performance of the baseline regulator, however the hybrid logic improves its performances for its robustness to measurement noise. The hybrid PID controller resulted less performing than the other two due to a non-optimal tuning which was not possible to achieve in the context of this thesis.

Starting from the results obtained, recommendations and possible future development are:

- to find a systematic design methodology to tune in an optimal way the hybrid PID-like controller to achieve better performances,
- to perform simulations and flight tests to study the behavior of the controllers during more aggressive attitude maneuvers,
- to test different structures for the hybrid controller attitude controllers of the UAV, *e. g.*, taking into account only the reduced attitude so as to prioritize roll and pitch over the yaw.

Bibliography

- [1] Chrobotics.
<http://www.chrobotics.com/library/understanding-euler-angles>.
- [2] PX4 documentation.
https://docs.px4.io/en/airframes/airframe_reference.html.
- [3] C. G. Mayhew. Hybrid control for topologically constrained systems, December 2010.
- [4] R. Goebel, R. G. Sanfelice, and A. R. Teel. Hybrid dynamical systems: Robust stability and control for systems that combine continuous-time and discrete-time dynamics. *IEEE Control Systems Magazine*, 29(2):28–93, 2009.
- [5] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Robust global asymptotic attitude stabilization of a rigid body by quaternion-based hybrid feedback. In *Proceedings of the 2009 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 2522–2527, 2009.
- [6] J. Su and K. Cai. Globally stabilizing proportional-integral-derivative control laws for rigid-body attitude tracking. *Journal of Guidance, Control, and Dynamics*, 34(4):1260–1264, July-August 2011.
- [7] Aerospace Systems and Control Laboratory.
<http://ascl.daer.polimi.it/>.
- [8] R. Goebel, R. G. Sanfelice, and A. R. Teel. *Hybrid dynamical systems*. Princeton University Press, 2012.
- [9] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Robust global asymptotic stabilization of a 6-DOF rigid body by quaternion-based hybrid feedback. In *Proceedings of the 2009 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 1094–1099, 2009.
- [10] C. G. Mayhew and A. R. Teel. Global asymptotic stabilization of the inverted equilibrium manifold of the 3-D pendulum by hybrid feedback. In *Proceedings*

- of the 2010 49th IEEE Conference on Decision and Control (CDC), pages 679–684, 2010.
- [11] C. G. Mayhew and A. R. Teel. Synergistic hybrid feedback for global rigid-body attitude tracking on $\text{SO}(3)$. *IEEE Transactions on Automatic Control*, 58(11):2730–2742, 2013.
- [12] P. Casau, R. G. Sanfelice, R. Cunha, D. Cabecinhas, and C. Silvestre. Global trajectory tracking for a class of underactuated vehicles. In *Proceedings of the 2013 American Control Conference (ACC)*, pages 419–424, 2013.
- [13] P. Casau, R. G. Sanfelice, and C. Silvestre. Adaptive backstepping of synergistic hybrid feedbacks with application to obstacle avoidance. In *Proceedings of the 2019 American Control Conference (ACC)*, pages 1730–1735, 2019.
- [14] P. Casau, R. Cunha, and C. Silvestre. Improved maneuverability for multirotor aerial vehicles using globally stabilizing feedbacks. In *Proceedings of the 2020 American Control Conference (ACC)*, pages 3822–3827, 2020.
- [15] B. P. Malladi, R. G. Sanfelice, E. Butcher, and J. Wang. Robust hybrid supervisory control for rendezvous and docking of a spacecraft. In *Proceedings of the IEEE 55th Conference on Decision and Control (CDC)*, pages 3325–3330, December 2016.
- [16] G. Gozzini, D. Invernizzi, S. Panza, M. Giurato, and M. Lovera. Air-to-air automatic landing of unmanned aerial vehicles: A quasi time-optimal hybrid strategy. *IEEE Control Systems Letters*, 4(3):692–697, 2020.
- [17] J. I. Poveda, M. Benosman, and A. R. Teel. Hybrid online learning control in networked multiagent systems: A survey. *International Journal of Adaptive Control and Signal Processing*, 33(2):228–261, 2019.
- [18] C. Prieur. Uniting local and global controllers with robustness to vanishing noise. *Mathematics of Control, Signals and Systems*, 14(2):143–172, 2001.
- [19] R. G. Sanfelice, A. R. Teel, and R. Goebel. Supervising a family of hybrid controllers for robust global asymptotic stabilization. In *Proceedings of the 2008 47th IEEE Conference on Decision and Control (CDC)*, pages 4700–4705, 2008.
- [20] J. R. Crane, C. W. T. Roscoe, B. P. Malladi, G. Zucchini, E. Butcher, R. G. Sanfelice, and I. I. Hussein. Hybrid control for autonomous spacecraft rendezvous proximity operations and docking. *IFAC-PapersOnLine*, 51(12):94–99, 2018.

-
- [21] G. Zucchini, B. P. Malladi, R. G. Sanfelice, and E. A. Butcher. Robust hybrid supervisory control for a 3-DOF spacecraft in close-proximity operations. *IFAC-PapersOnLine*, 51(12):88–93, 2018.
- [22] G. Gozzini. UAV autonomous landing on moving aerial vehicle, 2019.
- [23] D. Fragopoulos and M. Innocenti. Stability considerations in quaternion attitude control using discontinuous Lyapunov functions. *IEE Proceedings - Control Theory and Applications*, 151(3):253–258, 2004.
- [24] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. On quaternion-based attitude control and the unwinding phenomenon. pages 299–304, 2011.
- [25] B. L. Stevens, F. L. Lewis, and E. N. Johnson. *Aircraft Control and Simulation: Dynamics, Control Design and Autonomous Systems*. 2015.
- [26] R. Kristiansen, P. J. Nicklasson, and J. T. Gravdahl. Satellite attitude control by quaternion-based backstepping. *IEEE Transactions on Control System Technology*, 17(1):227–232, January 2009.
- [27] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Quaternion-based hybrid control for robust global attitude tracking. *IEEE Transactions on Automatic Control*, 56(11):2555–2566, November 2011.
- [28] ANT-X.
<https://antx.it/>.
- [29] G. van der Veen, J. W. van Wingerden, M. Bergamasco, M. Lovera, and M. Verhaegen. Closed-loop subspace identification methods: an overview. *IET Control Theory and Applications*, 10(7):1339–1358, 2013.
- [30] D. Invernizzi, S. Panza, and M. Lovera. Robust tuning of geometric attitude controllers for multirotor unmanned aerial vehicles. *Journal of Guidance, Control, and Dynamics*, 43(7):1332–1343, July 2020.
- [31] D. Invernizzi, M. Lovera, and L. Zaccarian. Geometric trajectory tracking with attitude planner for vectored-thrust VTOL UAVs. In *Proceedings of the 2018 American Control Conference (ACC)*, pages 3609–3614, 2018.
- [32] PX4 Autopilot - Open Source Autopilot for Drones.
<https://px4.io/>.
- [33] L. Meier. MAVLink.
<https://mavlink.io/en/>.
- [34] Motive. Optitrack.
<https://optitrack.com/products/motive/>.

- [35] Open Source Robotic Foundation. Ros.
<http://wiki.ros.org/Documentation>.
- [36] MATLAB. MathWorks.
<https://it.mathworks.com/products/matlab.html>.