Robotica: page 1 of 22. © Cambridge University Press 2019
doi:10.1017/S0263574719001231

Robust Geometric Navigation of a Quadrotor UAV on SE(3)

O. Garcia†*, E. G. Rojo-Rodriguez†, A. Sanchez‡, D. Saucedo§ and A. J. Munoz-Vazquez¶

†*Aeronautics Department, Aerospace Engineering Research and Innovation Center, Faculty of Mechanical and Electrical Engineering, Autonomous University of Nuevo Leon, Apodaca, NL, Mexico*
E-mails: octavio.garcias@uanl.mx; erik.rojordr@uanl.edu.mx

‡*Aeronautics Department, Robotics and Advanced Manufacturing Department, CINVESTAV, Saltillo, Coahuila, Mexico*
E-mail: anand.sanchez@cinvestav.mx

§*National Polytechnic Institute, UPIIG, Silao de la Victoria, Guanajuato, Mexico*
E-mail: dsaucedog@ipn.mx

¶*Computer Science Department, CONACYT-School of Engineering, Autonomous University of Chihuahua, Campus II, Chihuahua, Mexico*
E-mail: aldo.munoz.vazquez@gmail.com

(Accepted July 2, 2019)

SUMMARY
In this paper, a robust geometric navigation algorithm, designed on the special Euclidean group SE(3), of a quadrotor is proposed. The equations of motion for the quadrotor are obtained using the Newton–Euler formulation. The geometric navigation considers a guidance frame which is designed to perform autonomous flights with a convergence to the contour of the task with small normal velocity. For this purpose, a super twisting algorithm controls the nonlinear rotational and translational dynamics as a cascade structure in order to establish the fast and yet smooth tracking with the typical robustness of sliding modes. In this sense, the controller provides robustness against parameter uncertainty, disturbances, convergence to the sliding manifold in finite time, and asymptotic convergence of the trajectory tracking. The algorithm validation is presented through experimental results showing the feasibility of the proposed approach and illustrating that the tracking errors converge asymptotically to the origin.

KEYWORDS: Quadrotor UAV; Geometric navigation; Guidance frame; Super twisting algorithm.

Nomenclature

ξ	=	position variables
η	=	orientation variables
$\dot{\xi}$	=	linear velocities
Ω	=	angular velocities
R	=	rotation matrix
D_ξ, D_η	=	disturbances
T_T	=	total thrust
R_d	=	desired rotation matrix
τ_a	=	actuator moments
$\{f, b, n\}$	=	set of forward, binormal, and normal vectors
$\{f_g, b_g, n_g\}$	=	set of reference forward, binormal, and normal vectors
u_n	=	position control
u_a	=	orientation control

* Corresponding author. E-mail: octavio.garcias@uanl.mx

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

1. Introduction

In recent years, research efforts have been conducted to the stabilization of quadrotor unmanned aerial vehicles (UAVs) for regulation and trajectory tracking. The quadrotor UAV is considered as a highly underactuated nonlinear system with fast orientation and slow coupled position dynamics. Having four thrusters for performing hovering and navigation flight, the quadrotor UAV is a challenge for the stabilization in regulation and trajectory tracking. In this sense, asymptotic methods based on Lyapunov stability theory can demonstrate limited usefulness for a quadrotor due to the different convergence time in which the quadrotor reaches the desired point or trajectory. Linear and nonlinear controllers have been proposed in order to solve the underactuation system of a wide range of unmanned vehicles as fixed-wing and quadrotor UAVs, see refs.[1–3] In particular, control strategies based on geometric control theory have shown a convenient performance for quadrotors, which reach tasks with an asymptotic convergence to the trajectory and with low normal velocity. This paper presents a solution to the underactuation problem for a quadrotor UAV that has four independent control inputs for the six degrees of freedom of translational and rotational motion in the three-dimensional Euclidean space. Then, the navigation of a quadrotor is based on a geometric tracking control in SE(3) (special Euclidean group) which consists of a second-order sliding mode control based on super twisting (ST) control algorithm.

Research works related to geometric control for the quadrotor UAV are mentioned in ref.[4] in which the authors presented an analytical model and a geometric numerical integrator for a system of rigid bodies connected by ball joints. Simulation results showed the validation of the proposed approach for a system of three connected rigid bodies. In ref.[5] a nonlinear tracking controller on the special Euclidean group SE(3) of a quadrotor UAV rigid body dynamics was developed. Simulation results were presented in order to validate the proposed theory. In ref.[6] a nonlinear tracking control system was defined for a quadrotor UAV with a mathematical model of a quadrotor UAV on the special Euclidean group. The nonlinear output-tracking controllers were developed to follow an attitude and a position command for the vehicle. Numerical simulations showed the complex maneuvers that were performed by the quadrotor. In ref.[7] authors addressed the problem of position and attitude estimation based on landmark readings and velocity measurements, and a nonlinear observer on SE(3) was proposed using a Lyapunov function conveniently expressed as a function of the difference between the estimated and the measured landmark coordinates. Results in simulation illustrated the stability and convergence properties of the observer. In ref.[8] the modeling and control of miniature quadrotors with a special emphasis on attitude control were presented. A hybrid Backstepping control was designed based on Frenet–Serret theory for the attitude stabilization, introducing a desired attitude angle acceleration function dependent on aircraft velocity. Simulation and flight experiments were conducted in the Draganflyer quadrotor. Authors, in ref.[9] presented an attitude tracking control scheme that exhibits almost global asymptotically stable tracking of a desired attitude trajectory with perfect state measurements. These control and estimation schemes employed the global unique representation of rigid body attitude provided by rotation matrices. Simulation results showed the validation of the proposed approach. The problem of controlling an underactuated UAV subject to Newton–Euler dynamics following a maneuvering, unpredictable target in the three-dimensional Euclidean space was solved in ref.[10] and a nonlinear control law based on backstepping technique was proposed. Numerical simulation showed the capabilities of the proposed controller. In ref.[11] continuous finite-time stabilization of rigid body attitude dynamics was considered using a coordinate free representation of attitude on the Lie group of rigid body rotations in three dimensions, SO(3). Simulation results for a spacecraft in low Earth orbit, using a Lie group variational integrator, validated the stability and robustness properties of the attitude feedback stabilization scheme. In refs.[12, 13] an integrated guidance and feedback control scheme for steering an underactuated aerial vehicle through desired waypoints in three-dimensional space was presented in the framework of geometric mechanics in which the guidance problem was developed on the special Euclidean group of rigid body motions SE(3). Then, a feedback control law was obtained to steer the underactuated vehicle towards the desired trajectories in translation and rotation. Simulation results were presented in order to verify the almost global asymptotic stability. Authors, in ref.[14] designed a rigid body controller based on the tangent bundle TSE(3) of the Lie group SE(3) using a backstepping technique. Here, the geometric mechanics in the control design laid an almost asymptotic

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

stability of the resulting motion. Simulation results illustrated the effectiveness of the proposed control methodology. In ref.[15] authors presented a tutorial for the modeling, estimation, and control of quadrotors. The dynamic model is based on a rigid body motion in SE(3) and the simple aerodynamics is studied. In ref.[16] the output regulation problem of rigid bodies was addressed in which the kinematic configuration space lies on the Special Euclidean group SE(3). Simulation results are presented in order to validate the proposed backstepping control. In ref.[17] authors presented a strategy for the trajectory tracking of a quadrotor with collision avoidance and based on geometrical relations. Numerical simulations are performed to prove the proposed algorithm. The research works previously listed address geometric control strategies for quadrotors, which were validated by means of numerical results; however, a real-time implementation of these methods was not presented. In our work, the geometric navigation strategy, based on the special Euclidean group SE(3), of a quadrotor UAV subject to disturbances is presented in which a guidance frame is proposed, and a ST control with a cascade structure for the rotational and translational dynamics of the vehicle is designed. This control provides robustness against parameter uncertainty and disturbances guaranteeing an asymptotic convergence of the trajectory tracking. Real-time experiments are executed in order to demonstrate the feasibility and effectiveness of the proposed strategy.

For quadrotors with the sliding mode control, some publications can be cited. In ref.[18] authors proposed and implemented a ST controller in order to control only the orientation of a quadrotor platform fixed on a support. The algorithm was executed in a ground station computer, and the information was transmitted through data wires to the platform. It is important to mention that when an UAV is fixed on a support, even though it could have a pivoting joint for its movement, the response of the aerial vehicle does not correspond to that of a free flight, it is only an approximation because the joint provides damping in the orientation dynamics that does not exist in real free flights. In ref.[19] author presented a ST sliding mode controller designed for a quadrotor UAV. This paper uses a reduced model of the quadrotor and only shows simulation results. In addition, a geometric concept was not applied in the control strategy and a real-time implementation of this method was not presented. In ref.[20] authors presented a robust second-order sliding mode controller for the attitude (3 axis of rotation) stabilization of a quadrotor. The paper uses a reduced mathematical model of the quadrotor and shows experimental results in orientation (roll, pitch and roll – 3 axes). The control strategy of this paper is not based on geometric concept. In ref.[21] authors presented the design and real-time implementation of three second-order sliding mode controllers for the altitude tracking of a quadrotor without a geometric approach. In this paper, authors showed the simulation and experimental results for the altitude equation (z axis) of a quadrotor. In ref.[22] authors presented the control design and the stability analysis for the trajectory tracking of a quadrotor. This paper only showed simulation results considering disturbances. In ref.[23] the simultaneous tracking of position and time-varying heading was proposed, based on an integral sliding mode control with a quaternion-based sliding surface for the quadrotor UAV. In this research, only simulation results were presented. Consequently, in ref.[24] an embedded ST control for the orientation dynamics of a quadrotor and the real-time experiments during the autonomous flight were presented. Here a commercial flight computer was used and modified to implement the sliding mode controller. In ref.[25] autonomous trajectory tracking of quadrotors using monocular vision was presented using a second-order sliding mode control algorithm. Previous research works present the control and stability of the quadrotor without using the geometric approach. In our work, the navigation of the quadrotor is based on a geometric approach.

The novelty of this paper is twofold. On one hand, a guidance frame is designed in special Euclidean group SE(3), such that the quadrotor UAV reaches the desired trajectory with a high-degree of precision and an asymptotic convergence to the contour of the task with small normal velocity. On other hand, a robust controller, ST algorithm, with a cascade structure for the nonlinear rotational and translational dynamics of the quadrotor UAV is proposed, such that a virtual position controller guarantees that the sliding mode is enforced in finite time for the position dynamics and subsequently the attitude controller stabilizes the orientation dynamics in the desired orientation based on the guidance frame against disturbances due to gust winds, as well as transverse flow and ground effects.

Our approach may prove useful for more stringent or real applications, such as precision agriculture, since smooth contour tracking is obtained with a damped response, which may be needed for monitoring and mapping crop fields.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

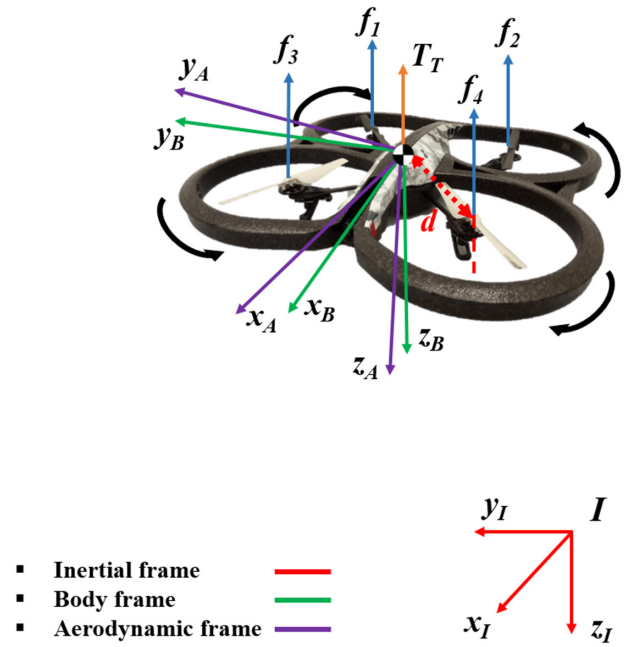

Fig. 1. Parrot AR.Drone 2.0 UAV.

1.1. Main contribution

The geometric navigation, based on the special Euclidean group SE(3), of a quadrotor UAV subject to disturbances is proposed. The navigation scheme of the quadrotor is based on a geometric tracking control in SE(3) in which a guidance frame is proposed and a ST control with a cascade structure for the rotational and translational dynamics of the vehicle is designed. The contribution of this paper is summarized as follows:

1. A scheme of navigation, based on the special Euclidean group SE(3), for a quadrotor is proposed.
2. A guidance frame is designed in order to achieve the navigation of the vehicle.
3. The proposed geometric navigation algorithm is implemented in real time to perform autonomous flights.

The rest of this paper is organized as follows: Section 2 presents the equations of motion for the quadrotor UAV using the Newton–Euler formulation. Section 3 describes the geometric navigation, based on the special Euclidean group SE(3), with a guidance frame and a ST control. Section 4 presents the experimental platform and the autonomous navigation results of the quadrotor UAV. Finally, conclusions are given in Section 5.

2. Equations of Motion for a Quadrotor UAV

The model of the quadrotor considers an inertial fixed frame as $\mathcal{I} = \{x_\mathcal{I}, y_\mathcal{I}, z_\mathcal{I}\}$ and a body frame fixed attached to the center of gravity of the vehicle as $\mathcal{B} = \{x_\mathcal{B}, y_\mathcal{B}, z_\mathcal{B}\}$. The wind frame $\mathcal{A} = \{x_\mathcal{A}, y_\mathcal{A}, z_\mathcal{A}\}$ is considered during the forward flight,[2, 26] (see Fig. 1). The configuration of the quadrotor UAV is defined by the location of the center of gravity and the attitude with respect to the inertial frame. Then, the configuration manifold is the special Euclidean group SE(3), which is the semidirect product of $\mathbb{R}^3$ and the special orthogonal group SO(3).

The Newton–Euler formulation, for rigid body, is used in order to obtain the dynamic model of the quadrotor

$$\dot{\xi} = V \tag{1}$$

$$m\dot{V} = (-T_T)Re_{b_3} + mge_3 + D_\xi(t) \tag{2}$$

$$\dot{R} = R\hat{\Omega} \tag{3}$$

$$J\dot{\Omega} = -\Omega \times J\Omega + \tau_a + D_\eta(t) \tag{4}$$

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

where $\xi = (x, y, z)^\top \in \mathbb{R}^3$ and $V = (\dot{x}, \dot{y}, \dot{z})^\top \in \mathbb{R}^3$ are the position coordinates and translational velocity relative to the inertial frame. $\eta = (\phi, \theta, \psi)^\top \in \mathbb{R}^3$ describes the rotation coordinates where ϕ, θ, and ψ represent the roll, pitch, and yaw or heading, respectively. The rotation matrix, $R \in \mathrm{SO}(3) : \mathcal{B} \to \mathcal{I}$, satisfies the $\mathrm{SO}(3) = \{R | R \in \mathbb{R}^{3\times 3}, \det[R] = 1, RR^\top = R^\top R = I\}$ and is parameterized by the Euler angles ϕ, θ, and ψ. The rotation matrix is written as

$$R = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}$$

where the shorthand notation of $s_a = \sin(a)$ and $c_a = \cos(a)$ is used. $\Omega = (p, q, r)^\top \in \mathbb{R}^3$ is the angular velocity in $\mathcal{B}$, where the hat map $\hat{\cdot} : \mathbb{R}^3 \to \mathfrak{so}(3)$ is defined by the condition that $\hat{a}b = a \times b$ for all $a, b \in \mathbb{R}^3$.

$$\hat{\Omega} = \begin{pmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{pmatrix}$$

e_1, e_2, and e_3 are the vectors of the canonical basis of $\mathbb{R}^3$ in $\mathcal{I}$. e_{b_1}, e_{b_2}, and e_{b_3} are the vectors of the canonical basis of $\mathbb{R}^3$ in $\mathcal{B}$. $J \in \mathbb{R}^{3\times 3}$ contains the moments of inertia and $m \in \mathbb{R}$ denotes the mass of the quadrotor UAV. $T_T \in \mathbb{R}_{>0}$ is the total thrust and $\tau_a = (\tau_\phi, \tau_\theta, \tau_\psi)^\top \in \mathbb{R}^3$ is the actuator moments of the quadrotor. $D_\xi = (d_{\xi_1}, d_{\xi_2}, d_{\xi_3})^\top \in \mathbb{R}^3$ and $D_\eta = (d_{\eta_1}, d_{\eta_2}, d_{\eta_3})^\top \in \mathbb{R}^3$ are time-varying and state-dependent disturbances.[1,23]

The actuator moments τ_a generated by the four rotors are described as follows:

$$\tau_a = \begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \begin{pmatrix} d[(f_2 + f_4) - (f_1 + f_3)] \\ d[(f_3 + f_4) - (f_1 + f_2)] \\ Q_2 + Q_3 - Q_1 - Q_4 \end{pmatrix}$$

where d denotes the distance from the center of mass to the rotor axis and f_i is the lift force or thrust force of the propeller for $i = 1, 2, 3, 4$. $Q_i = \rho_a A_{d_i} \omega_i^2 r_i^3 c_{Q_i}$ is the reaction moment for the rotors with c_{Q_i} that denotes the rotor shaft moment coefficient, ρ_a is the air density, A_{d_i} denotes the disk area, ω_i denotes the angular velocity of the rotor, and r_i is the rotor radius for $i = 1, 2, 3, 4$.

Gyroscopic moment The gyroscope moment generated by the rotation of the airframe and the four rotors is described as

$$d_{\eta_{gyro}} = \sum_{i=1}^{4} (-1)^{i+1} I_{r_i} [\Omega \times (\omega_i e_{b_3})]$$

where I_{r_i} is the moment of inertia of the rotor i and ω_i denotes the angular velocity of the rotor i, with $i = 1, 2, 3, 4$.

Disturbances due to wind

The *aerodynamic forces* produced during the flight are written as

$$D_\xi(t) = \begin{pmatrix} d_{\xi_1} \\ d_{\xi_2} \\ d_{\xi_3} \end{pmatrix} = RW^T \begin{pmatrix} L(t) \\ Y(t) \\ D(t) \end{pmatrix}$$

with the rotation aerodynamic matrix $W : \mathcal{B} \to \mathcal{A}$ that transforms a force from the body frame to aerodynamic frame is described as

$$W = \begin{pmatrix} c_\alpha c_\beta & s_\beta & s_\alpha c_\beta \\ -c_\alpha s_\beta & c_\beta & -s_\alpha s_\beta \\ -s_\alpha & 0 & c_\alpha \end{pmatrix}$$

where α is the angle of attack and β is the sideslip angle. L, Y, and D are the aerodynamic forces: lift, side force, and drag, respectively.[27]

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

The *Aerodynamic moments* The aerodynamic moments generated during the flight are written as described as

$$d_{\eta_{aero}}(t) = \begin{pmatrix} \mathcal{L}(t) \\ \mathcal{M}(t) \\ \mathcal{N}(t) \end{pmatrix} \tag{5}$$

where $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{N}$ are the aerodynamic rolling, pitching, and yawing moments, respectively.[2, 26–28]

Rewriting the disturbances in the time-varying vectors $D_\xi(t)$ and $D_\eta(t)$, it results

$$D_\xi(t) = \begin{pmatrix} d_{\xi_1} \\ d_{\xi_2} \\ d_{\xi_3} \end{pmatrix}, \quad D_\eta(t) = d_{\eta_{gyro}}(t) + d_{\eta_{aero}}(t) = \begin{pmatrix} d_{\eta_1} \\ d_{\eta_2} \\ d_{\eta_3} \end{pmatrix}$$

which are continuously differentiable.

In this paper, the advantages of using the Euler angles are that (i) these angles are used to design the desired attitude, based on the guidance frame as a geometric approach, (ii) these angles provide a high degree of coordination between the degrees of freedom of the quadrotor system since the columns of the rotation matrix are constructed to achieve the required position-thrust controller, which in turn enforces the tracking of the desired position trajectory, and (iii) these angles are manipulated directly in order to achieve an asymptotic convergence to the contour of the task with small normal velocity. The reason of using these angles is that the quadrotor cannot perform aggressive maneuvers such as loops due to the singularity existing in conventional Euler angles schemes; nonetheless, quaternion-based schemes solve the singularity problems and can be employed in the control design for aggressive trajectories of quadrotors.[29]

Remark 1. As the rotation of the four propellers on the quadrotor is balanced, gyroscopic moment (15) will essentially be zero. The only case in which gyroscopic moments will not be zero is if there is a significant difference in the RPM of the four motors and in the presence of a strong sideways cross-wind.

Remark 2. The real-time flight scenario consists of a quadrotor operating at a determined position and attitude. The ground effect and sideways cross-wind may cause undesired changes in the SE(3)-navigation of the aerial vehicle. In this work, the aerodynamic forces and moments are considered as time-varying disturbances.

3. Navigation of the Quadrotor

The navigation of the quadrotor is based on a geometric tracking control in SE(3) (special Euclidean group) which is a second-order sliding mode control known as ST control. The ST control is a continuous second-order sliding mode control that provides finite-time convergence of the sliding variable and an asymptotic convergence of the output tracking. In effect, the twisting controller is a discontinuous second-order scheme that provides finite-time convergence, and it is called in this way due to the twisting of the position and velocity errors of trajectories in the phase plane. The super term is given to the continuous second-order scheme. This algorithm allows us to obtain asymptotic convergence of the trajectory tracking even in the presence of wind disturbances. For this purpose, the Eqs. (2)–(4) can be rewritten as

$$\dot{\xi} = V \tag{6}$$
$$\dot{V} = u_n + d_\xi(t) \tag{7}$$
$$\dot{R} = R\hat{\Omega} \tag{8}$$
$$\dot{\Omega} = u_a + d_R(t) \tag{9}$$

where $u_n \in \mathbb{R}^3$ and $u_a \in \mathbb{R}^3$ are virtual control inputs for the position and orientation dynamics. $d_\xi(t) = \frac{D_\xi(t)}{m}$ and $d_R(t) = J^{-1}[-\Omega \times J\Omega + D_\eta(t)]$. Then, these inputs can be written as

$$mu_n = mge_3 - T_T(Re_{b_3}) \tag{10}$$

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

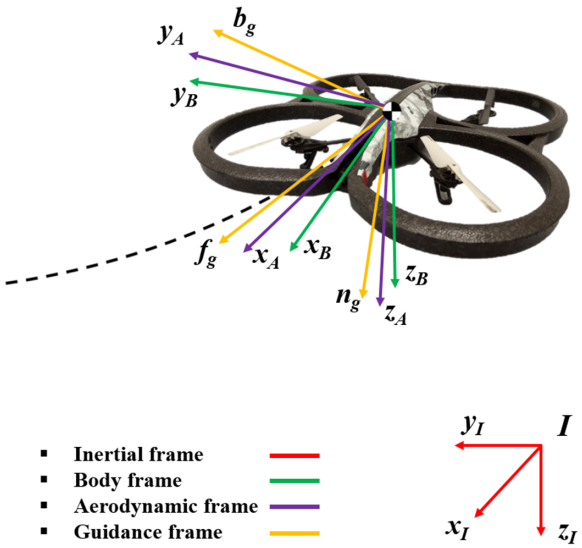

Fig. 2. Frames on the quadrotor UAV.

$$\tau_a = Ju_a \tag{11}$$

Definition 1. *A guidance frame $\mathcal{G} = \{f_g, b_g, n_g\}$ is a reference frame that consists of the control forward vector f_g, the control binormal vector b_g, and the control normal vector n_g. This frame satisfies the NED (North East Down) system and considers the terminology from the names of the three unit vectors in the reference frame for a curve in $\mathbb{R}^3$.*

The three vectors are defined as follows:

- The control normal vector n_g is defined as a function of the position and velocity errors.

$$n_g = \frac{ge_3 - u_n}{\|ge_3 - u_n\|} \tag{12}$$

- The control forward vector f_g is defined as a unit vector in the (n_g, t_d) plane and is orthogonal to n_g such as $n_g \cdot t_d > 0$ with $t_d = \frac{\xi_d}{\|\xi_d\|}$. Then

$$f_g = \frac{n_g \times e_1}{\|n_g \times e_1\|} \tag{13}$$

- The control binormal vector b_g is defined as

$$b_g = -(f_g \times n_g) \tag{14}$$

Definition 2. *A desired rotation matrix $R_d \in SO(3)$ is defined as $R_d = [f_g \;\; b_g \;\; n_g]$ corresponding to reference frame or guidance frame where $f_g = R_d e_1$, $b_g = R_d e_2$ and $n_g = R_d e_3$.*

Frames on the quadrotor UAV during the trajectory tracking are shown in Fig. 2 as follows:

Theorem 1. Let $T_T > 0$ and $u_n \in \mathbb{R}^3$, which can be written as a linear combination of $\{e_1, e_2, e_3\}$. Then, the following statements are true:

1. $u_n \neq ge_3$
2. n_g is a well-defined unit vector.
3. f_g is a well-defined unit vector.
4. $\{f_g, b_g, n_g\}$ is orthonormal and the matrix $R_d = [f_g \; b_g \; n_g]$ is orthogonal.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

Proof. 1. It can be proven by contradiction. Suppose $u_n = ge_3$. Then, from (10) it results

$$mge_3 = mge_3 - T_T R e_{b_3} \Rightarrow R e_{b_3} = 0$$

since $T_T > 0$. However, e_{b_3} is not a solution of the homogeneous equation $Re_{b_3} = 0$ since $e_{b_3} \neq 0$ and R is nonsingular, implying that $Re_{b_3} \neq 0$. This is a contradiction. Therefore $u_n \neq ge_3$.

2. From statement 1, it follows that $\|ge_3 - u_n\| \neq 0$. Therefore n_g is well defined.
3. Also, it can be proven by contradiction. Let $u_n = ae_1 + be_2 + ce_3$ for some $a, b, c \in \mathbb{R}$. Suppose f_g is not well defined, that is, $\|n_g \times e_1\| = 0 \Rightarrow n_g \times e_1 = 0$. From (12), this condition can be expressed as

$$\frac{(ge_3 - u_n) \times e_1}{\|ge_3 - u_n\|} = \frac{g\,(e_3 \times e_1) - a\,(e_1 \times e_1) - b\,(e_2 \times e_1) - c\,(e_3 \times e_1)}{\|ge_3 - u_n\|} = 0$$

which yields

$$(g - c)\, e_2 + be_3 = 0$$

this implies that $c = g$ and $b = 0$ since e_1 and e_2 are linearly independent. Then u_n is given by

$$u_n = ae_1 + ge_3$$

From (10), it results

$$ae_1 + \frac{T_T}{m} R e_{b_3} = R\left(ae_{b_1} + \frac{T_T}{m} e_{b_3}\right) = 0 \Rightarrow ae_{b_1} + \frac{T_T}{m} e_{b_3} = 0$$

since R is nonsingular. This implies that $a = 0$ and $\frac{T_T}{m} = 0$, since e_{b_1} and e_{b_3} are linearly independent, which in turn implies $u_n = ge_3$. However, $T_T > 0$ and $u_n \neq ge_3$ from statement 1. This is a contradiction. Therefore $\|n_g \times e_1\| \neq 0$, which in turn implies that f_g is well defined.

4. A direct computation shows that $\{f_g, b_g, n_g\}$ is orthonormal and $R_d = [f_g\ b_g\ n_g]$ is orthogonal. In fact, it is evident that the magnitude of both n_g and f_g is 1. Therefore $n_g^T n_g = f_g^T f_g = 1$. In addition,

$$n_g^T f_g = \frac{n_g^T\,(n_g \times e_1)}{\|n_g \times e_1\|} = \frac{e_1^T\,(n_g \times n_g)}{\|n_g \times e_1\|} = 0$$

which implies that f_g is orthogonal to n_g. Then,

$$b_g^T b_g = \|f_g \times n_g\|^2 = 1$$

since $\|f_g \times n_g\| = \|f_g\|\ \|n_g\| = 1$. Furthermore,

$$n_g^T b_g = -n_g^T\,(f_g \times n_g) = -f_g^T\,(n_g \times n_g) = 0$$
$$f_g^T b_g = -f_g^T\,(f_g \times n_g) = -n_g^T\,(f_g \times f_g) = 0$$

In consequence, $\{f_g, b_g, n_g\}$ is orthonormal. Finally,

$$R_d^T R_d = \begin{bmatrix} f_g^T f_g & f_g^T b_g & f_g^T n_g \\ b_g^T f_g & b_g^T b_g & b_g^T n_g \\ n_g^T f_g & n_g^T b_g & n_g^T n_g \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I$$

and R_d is nonsingular since $\{f_g, b_g, n_g\}$ is linearly independent. This implies that $R_d^{-1} = R_d^T$, then $R_d R_d^{-1} = R_d R_d^T = I$. Therefore, R_d is orthogonal.

Remark 3. The Theorem 1 provides a basis for the construction of a rotation matrix that encodes the tracking of position references, such that the quadrotor vehicle is required to be orientated in order to achieve the desired trajectory. The vector that points along the thrust is computed by means of the ST control algorithm, allowing the construction of an orthonormal basis for the desired rotation matrix, which in turn encodes the desired orientation of the vehicle body.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

3.1. Control of the position dynamics

For the control design of the position dynamics, Eqs. (6) and (7) are considered and a second sliding mode control is proposed. In effect, the relation among the virtual control input u_n, the thrust T_T, and the normal vector $n = Re_{b_3}$ is defined as

$$mu_n = mge_3 - nT_T \tag{15}$$

From the (15), T_T and n result as

$$\begin{aligned} T_T &= m\|ge_3 - u_n\| \\ n &= \tfrac{ge_3 - u_n}{\|ge_3 - u_n\|} \end{aligned} \tag{16}$$

Note that the current attitude is defined as $R = [f \;\; b \;\; n]$; thus, the forward, binormal, and normal vectors are described as $f = Re_{b_1}$, $b = Re_{b_2}$, and $n = Re_{b_3}$.

Now, a ST control u_n is proposed for the translational dynamics (ξ, V) in order to reach an asymptotic convergence of the trajectory tracking. For the tracking problem, the tracking error is defined as

$$\begin{aligned} e_\xi &= \xi_d - \xi \\ \dot{e}_\xi &= \dot{\xi}_d - \dot{\xi} \end{aligned} \tag{17}$$

where ξ_d and $\dot{\xi}_d$ are the desired position and velocity coordinates, respectively. Then, the sliding variable is defined as

$$s_n = \dot{e}_\xi + k_n e_\xi \tag{18}$$

Thus, the control normal vector $n_g(e, \dot{e})$ is proposed as a function of the position and velocity errors. In this case, the navigation problem is related to the tracking problem with $(\xi_d, \dot{\xi}_d, \ddot{\xi}_d) = (\xi_0, 0, 0)$. Then, $n_g(e, \dot{e})$ is defined as

$$n_g(e, \dot{e}) = \frac{ge_3 - u_n(s_n)}{\|ge_3 - u_n(s_n)\|} \tag{19}$$

In order to achieve the asymptotic tracking, the attitude controller must guarantee that the normal vector $n = Re_{b_3}$ asymptotically tracks the control normal vector $n_g = R_d e_3$.

Taking the sliding variable, it results

$$s_n = \dot{e}_\xi + k_n e_\xi \tag{20}$$

where $k_n > 0$, and differentiating

$$\begin{aligned} \dot{s}_n &= \ddot{e}_\xi + k_n \dot{e}_\xi \\ &= \ddot{\xi}_d + k_n \dot{\xi}_d - k_n \dot{\xi} - d_\xi - u_n \\ &= \rho(t, s) - u_n \end{aligned} \tag{21}$$

Assuming that $|\dot{\rho}(t, s)| \leq H_n$ the following control is described as

$$\begin{aligned} u_n &= c_n |s_n|^{1/2} \text{sign}(s_n) + w_n \\ \dot{w}_n &= b_n \text{sign}(s_n) \end{aligned} \tag{22}$$

where $c_n = 1.5\sqrt{H_n}$ and $b_n = 1.1H_n$. Thus, this controller and the s_n dynamics become

$$\begin{aligned} \dot{s}_n &= \rho(t, s) - c_n |s_n|^{1/2} \text{sign}(s_n) - w_n \\ \dot{w}_n &= b_n \text{sign}(s_n) \end{aligned} \tag{23}$$

Theorem 2. Consider the translational dynamics (2) and (3) and the controller along with the s_n dynamics (23). Then, the ST control guarantees an asymptotic convergence of the output tracking and sliding variable convergence in finite time

$$s_n \to 0 \Rightarrow \xi \to \xi_d \;\; \dot{\xi} \to \dot{\xi}_d \tag{24}$$

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

Remark 4. Since the virtual position controller needs to be induced via the robust tracking of attitude references, the sliding mode enforcement is sacrificed to obtain a smooth controller, which is performed by means of the smooth approximation $\tanh(kx) \approx \text{sign}(x)$, assuring the convergence of s_n into a compact convex set of centered at the origin, whose Lebesgue measure can be predefined via $k \gg 1$.

For more details about the stability analysis, see refs.[18,30,31]

3.2. Control of the orientation dynamics

In order to propose the control for the orientation dynamics, the desired rotation matrix $R_d = [f_g \;\; b_g \;\; n_g]$, corresponding to guidance frame or reference frame, is considered. Then, the desired angular velocity in the reference frame is defined as $\hat{\Omega}_d = R_d^\top \dot{R}_d$.[32] Based on the group operation of SO(3), the attitude and the angular velocity errors are defined as $R_e = RR_d^\top$ and $\Omega_e = \Omega - \Omega_d$, see Fig. 3.

Therefore, the error function on the rotation group is written as

$$\Upsilon(RR_d^\top) = \frac{1}{2}\text{tr}(I - RR_d^\top) \tag{25}$$

where $\Upsilon(RR_d)$ is a distance measure between two SO(3) matrices R and R_d. Note that $\Upsilon(RR_d) = 0 \Leftrightarrow R = R_d$. In addition, the following expression results:

$$\frac{d}{dt}\Upsilon(RR_d^\top) = \text{Skew}(RR_d^\top)^\vee(\Omega - \Omega_d) \tag{26}$$

where $\text{Skew}(A) = \frac{1}{2}(A - A^\top)$ and the operator $(\cdot)^\vee$ is the inverse of the "hat" operator (for the definition of asymptotic tracking on manifolds, see refs[32,33]).

Now, for control of the UAV orientation, $e_R = \text{Skew}(RR_d^\top)^\vee$ and $e_\Omega = -\Omega_e$ are proposed and the sliding variable is defined as

$$s_a = e_\Omega + k_a e_R \tag{27}$$

where $k_a > 0$, and differentiating

$$\begin{aligned} \dot{s}_a &= \dot{e}_\Omega + k_a \dot{e}_R \\ &= \dot{\Omega}_d + k_a(\dot{R}R_d^\top + R\dot{R}_d^\top - \dot{R}^\top R_d - R^\top \dot{R}_d)^\vee - d_R - u_a \\ &= \rho_R(t, s) - u_a \end{aligned} \tag{28}$$

Assuming that there exist some disturbances due to aerodynamic effects and gyroscopic moments $|\dot{\rho}_R(t, s)| \leq H_a$, then following control is given by

$$\begin{aligned} u_a &= c_a|s_a|^{1/2}\text{sign}(s_a) + w_a \\ \dot{w}_a &= b_a \text{sign}(s_a) \end{aligned} \tag{29}$$

where $c_a = 1.5\sqrt{H_a}$ and $b_a = 1.1H_a$.[34] Thus, the controller and the s_a dynamics become

$$\begin{aligned} \dot{s}_a &= \rho_R(t, s) - c_a|s_a|^{1/2}\text{sign}(s_a) - w_a \\ \dot{w} &= b_a \text{sign}(s_a) \end{aligned} \tag{30}$$

Remark 5. This methodology is not only restricted to the ST controller so that it can be applied to other affine controller strategies.

3.3. Algorithm simulation

In order to validate the proposed algorithm, a simulation was run considering wind gusts as disturbances. These disturbances consists of a *Gaussian*-shape signals simulating 2.5 m/s in the X and Y axes, and of 1.5 m/s in the Z axis; and a persistent wind disturbance with the velocity $v_{wind} = sin(0.8\pi t) + cos(0.2\pi t) + sin(0.4\pi t) + cos(0.6\pi t)$ in the X and Y axes.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

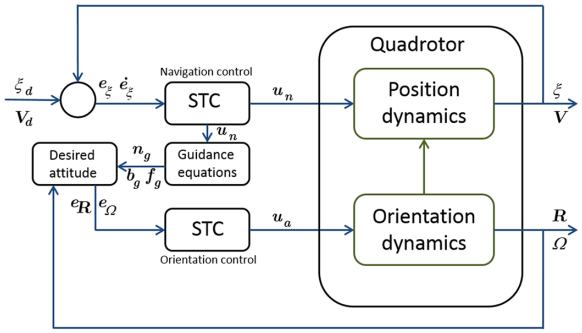

Fig. 3. Navigation structure.

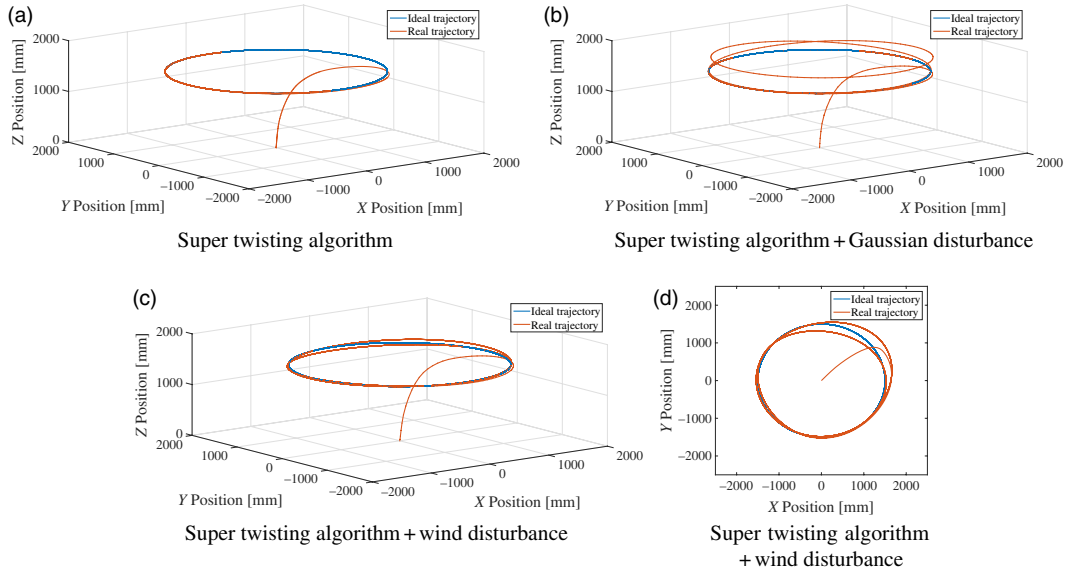

Fig. 4. Super twisting (ST) algorithm without/with disturbances. (a) ST algorithm. (b) ST algorithm + Gaussian disturbance. (c) ST algorithm + wind disturbance. (d) ST algorithm + wind disturbance.

In Fig. 4, the simulation of the ST algorithm without/with disturbances is illustrated. In these results, it can be seen that the proposed algorithm tracks satisfactorily the reference and reacts faster against the disturbance.

In Fig. 5, responses of the *Forward, Normal,* and *Binormal* components of the R_d and R are shown as well as the controller + wind disturbances. The proposed algorithm provides a robust response against disturbances, and an asymptotic convergence of the trajectory tracking.

3.4. Numerical comparison with other methods

A comparison with other methods in simulation is presented in order to demonstrate the feasibility and effectiveness of the proposed algorithm. In Fig. 6 a comparison of three strategies is shown and consists of the proposed algorithm using the ST controller, a PD tuning with the *Ziegler–Nichols* method, and a conventional sliding mode controller. Note that the ST controller tracks the trajectory faster and more accurate than the PD controller, and with more accurate and smoother than the conventional sliding mode controller.

In addition to these simulations, two different scenarios with disturbances were simulated. In the first, a *Gaussian*-shape disturbance is added to the system, simulating a fast wind gust of 2.5 m/s in the X and Y axes, and of 1.5 m/s in the Z axis. It can be seen in Fig. 7 that the ST algorithm reacts faster against the disturbance, maintaining a smooth response, in comparison with the PD controller and the conventional sliding mode controller.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

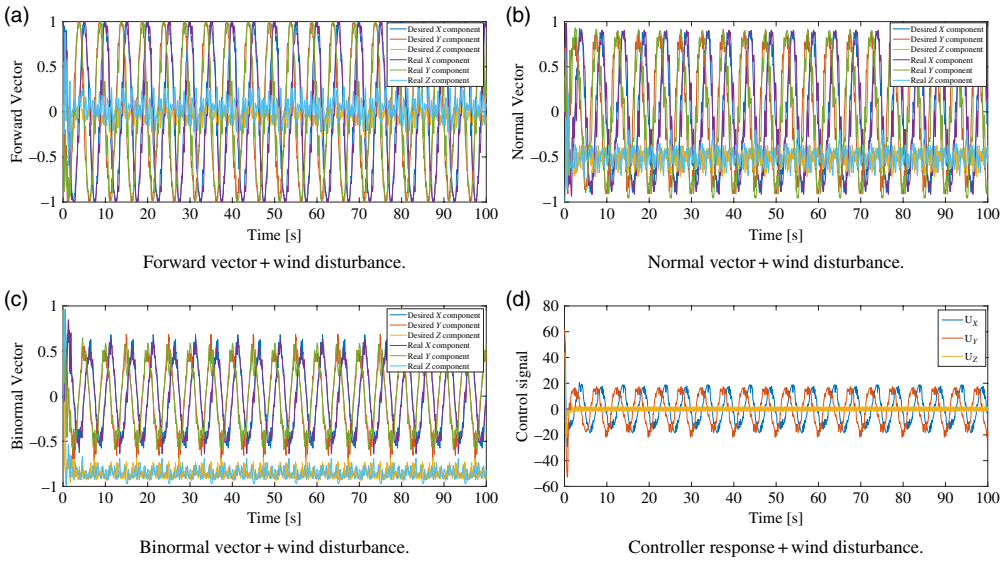

Fig. 5. Responses of the (a) *Forward*, (b) *Normal,* and (c) *Binormal* vectors and (d) controller + wind disturbance.

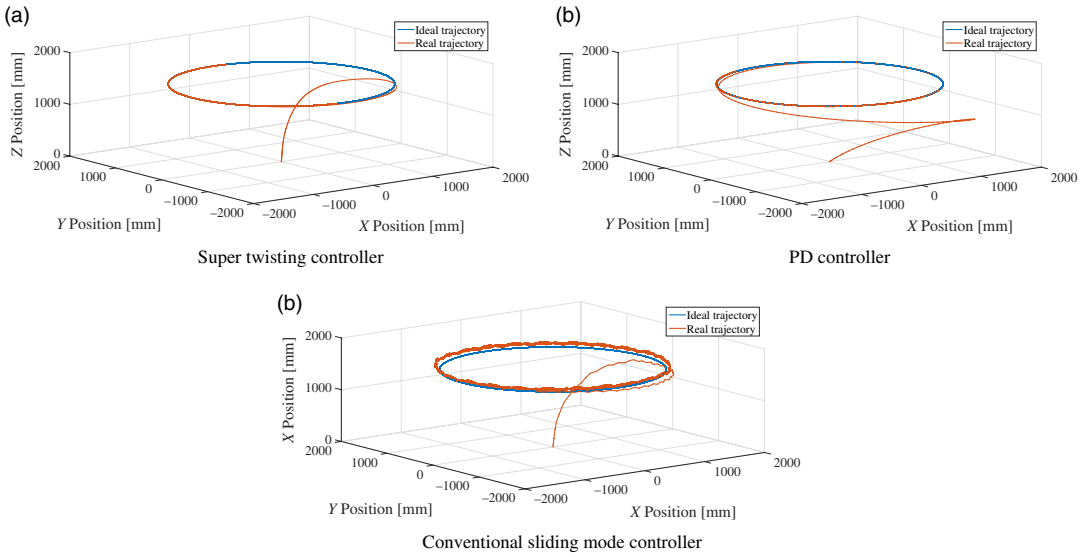

Fig. 6. Comparison of the trajectory between different control strategies. (a) Super twisting controller. (b) PD controller. (c) Conventional sliding mode controller.

As a second scenario, a persistent wind disturbance with the velocity $v_{wind} = sin(0.8\pi t) + cos(0.2\pi t) + sin(0.4\pi t) + cos(0.6\pi t)$ in the X and Y axes is simulated along with the closed-loop system. The response of the three strategies is shown in Fig. 8, in which the ST controller rejects the wind disturbances more efficiently and with less tracking error.

4. Experimental Results

The real-time experiments were conducted in the Navigation Laboratory at Aerospace Engineering Research and Innovation Center (CIIIA), and the experimental platform consists of a quadrotor UAV and a motion capture system with 16 VICON T40 cameras. The UAV employed for the implementation of the proposed algorithm is the Parrot AR.Drone 2.0 UAV, which is a quadrotor controlled through a WiFi connection. The technical characteristics of this aerial vehicle such as the processor type and sensors can be found in detail in refs,[35,36] see Fig. 9.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

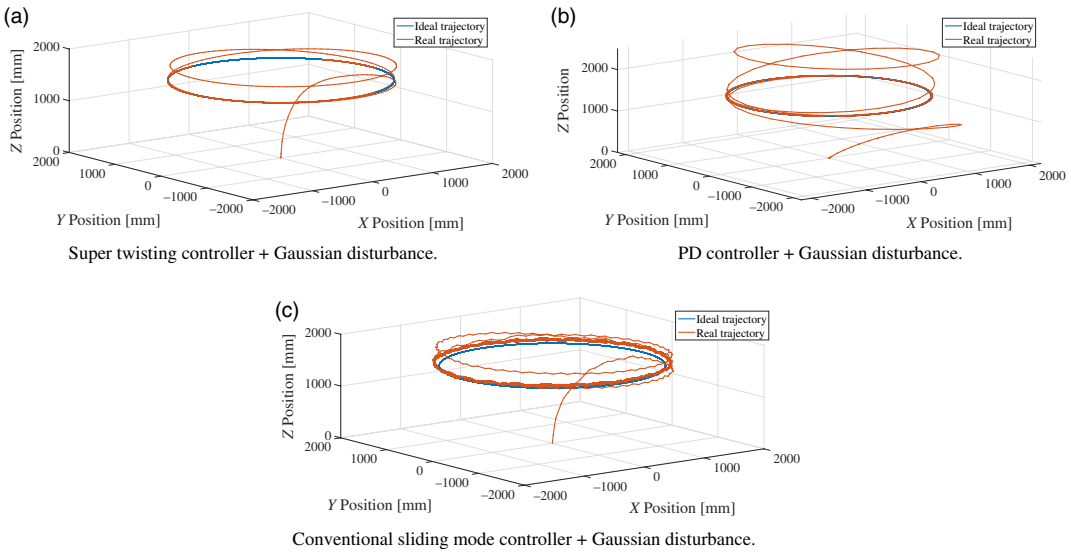

Fig. 7. Comparison of the trajectory between different control strategies subjected to a Gaussian disturbance. (a) Super twisting controller + Gaussian disturbance. (b) PD controller + Gaussian disturbance. (c) Conventional sliding mode controller + Gaussian disturbance.

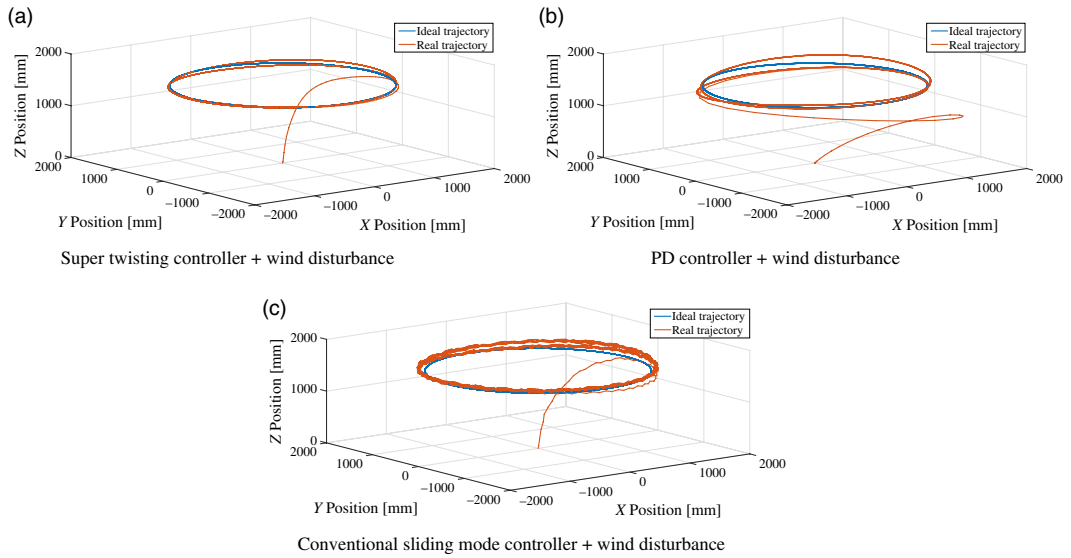

Fig. 8. Comparison of the trajectory between different control strategies subjected to a wind disturbance. (a) Super twisting controller + wind disturbance. (b) PD controller + wind disturbance. (c) Conventional sliding mode controller + wind disturbance.

For experimental purposes, derivatives such as $\dot{R}_d$ were computed through numerical approximation. In order to demonstrate the capabilities of the proposed algorithm, a series of experiments was executed.

4.1. Circular trajectory

In this experiment, a circular path is followed by a quadrotor. The radius of the circumference formed is of 1500 mm, while the angular velocity at which the vehicle travels around the path is of 2.4 rpm. The 3D trajectory followed by the quadrotor is illustrated in Fig. 10.

The individual-axis trajectory is shown in Fig. 11, where it can be seen that the proposed navigation algorithm correctly tracks the desired reference.

For the yaw angle ψ of the vehicle, it can be seen in Fig. 12 that it changes tangentially according to the trajectory defined, in this case the circumference. It is noted that the angle has a discontinuity

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

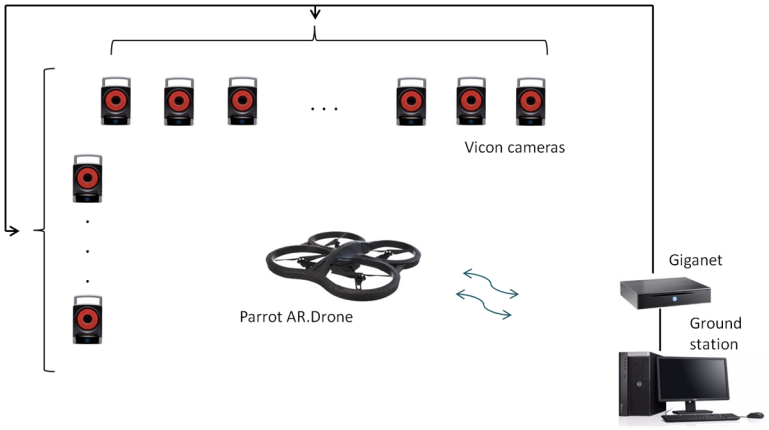

Fig. 9. Experimental platform.

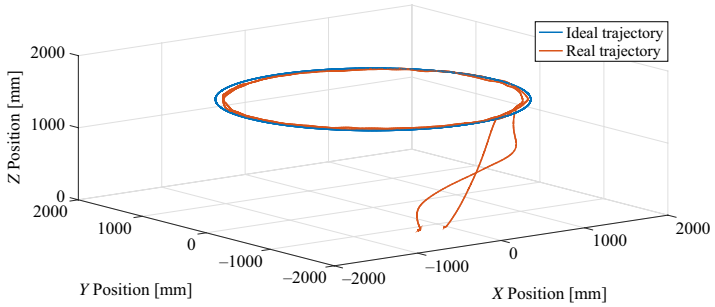

Fig. 10. Circular 3D trajectory.

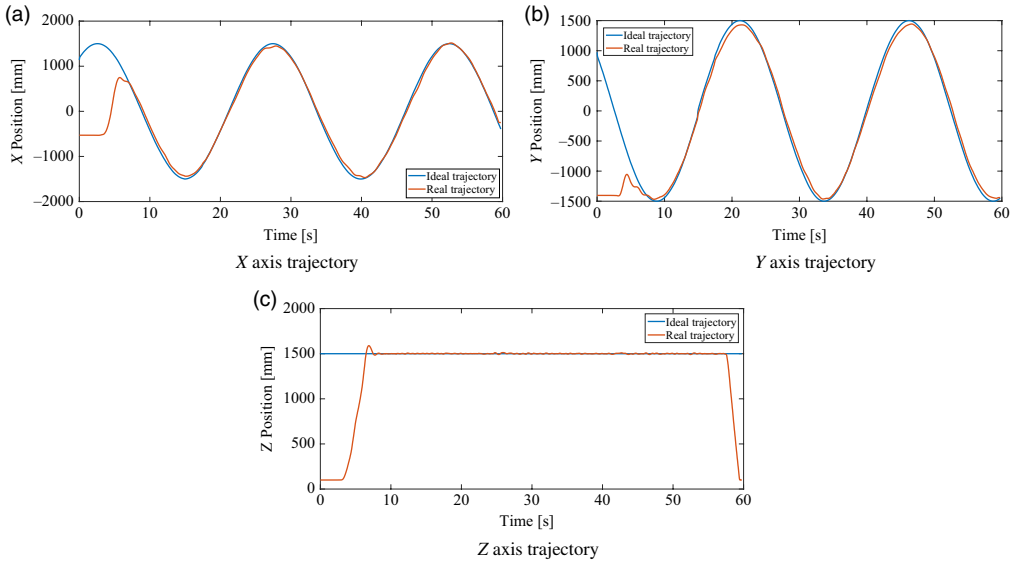

Fig. 11. *X,Y*, and *Z* axes components (a)–(c) of the circular trajectory.

every 360°, and this is due to that the trajectory requires the vehicle to do multiple turns, meaning that in each discontinuity, the vehicle begins a new turn.

The rotation matrix R and the desired rotation matrix R_d are shown in Fig. 13 in the form of their three main vectors, such as the binormal control vector, forward control vector, and the normal control vector, and each of these three decomposed in its scalar components.

The sliding variables calculated, and thus the control signals generated by the controller are shown in Fig. 14.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

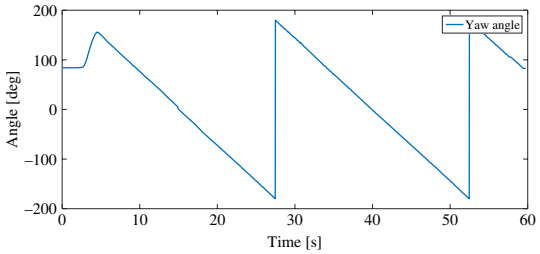

Fig. 12. Yaw angle performed during the circular trajectory.

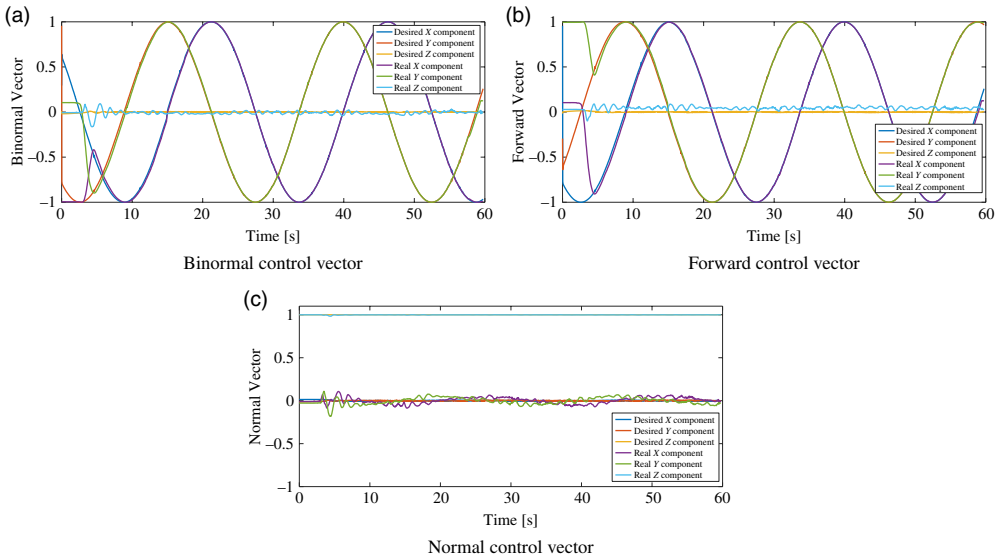

Fig. 13. Control vectors for the circular trajectory. (a) Binormal control vector. (b) Forward control vector. (c) Normal control vector.

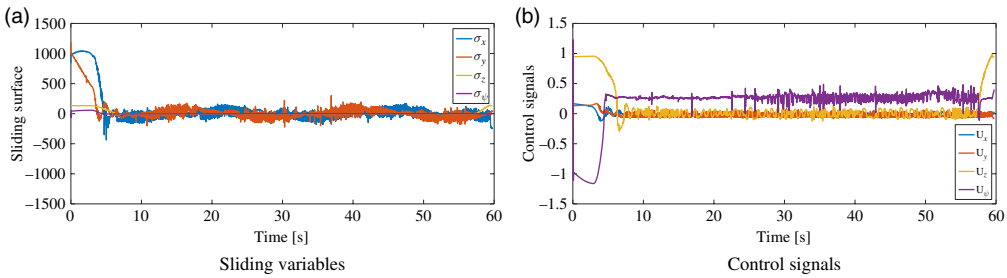

Fig. 14. (a) Sliding variables and (b) control signals for the circular trajectory.

4.2. Disturbed circular trajectory

This experiment focuses on the disturbance rejection by the vehicle. A main circular trajectory is followed by the quadrotor, while a wind disturbance is induced in the environment with gusts up to 6 m/s (21.6 km/h). This disturbance came from a fan located at coordinates $X = 0$, $Y = 3600$, and $Z = 1500$. The circumference described has a radius of 1500 mm and a velocity of 2.4 rpm.

In Fig. 15 it is possible to see the 3D trajectory of the vehicle. Note how the quadrotor is moved out of the ideal trajectory by the presence of the wind gusts, but despite the strong force induced in the body of the vehicle, the controller is able to return it to the trajectory and keep track of the circumference.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

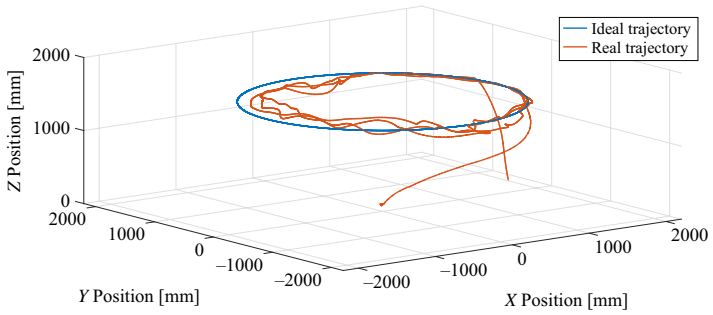

Fig. 15. Disturbed circular 3D trajectory.

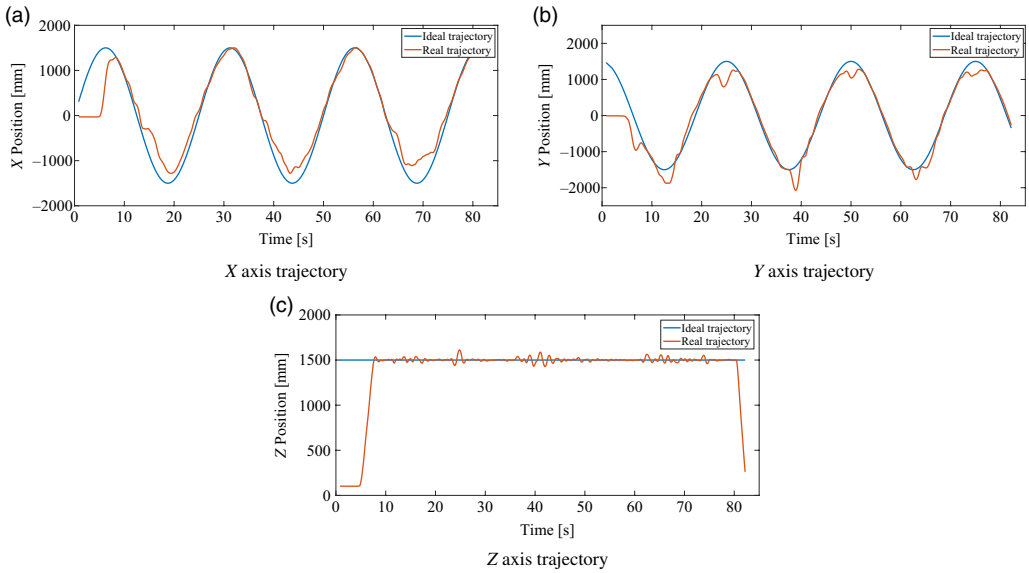

Fig. 16. *X*, *Y*, and *Z* axes components (a)–(c) of the disturbed circular trajectory.

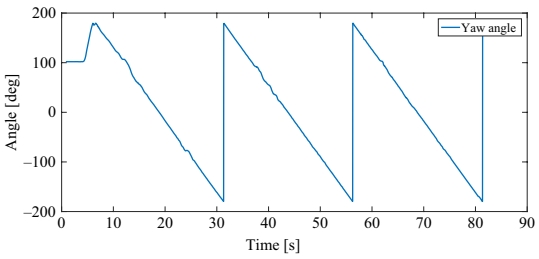

Fig. 17. Yaw angle performed during the disturbed circular trajectory.

For a clearer view of the components of movement, in Fig. 16 the individual-axis trajectory is shown. It is possible to see in Fig. 16b, which is the direction of the wind, the principal induced movement due to the disturbance. Furthermore, in the other axis, there is also a noticeable error due to the wind but less prominent than in the Y-axis.

The yaw angle ψ of the vehicle can be seen in Fig. 17, and it changes tangentially according to the trajectory defined, in this case the circumference as well. Each discontinuity seen in the graph means a complete turn performed by the vehicle.

In the same way, the rotation matrix R and the desired rotation matrix R_d can be seen in Fig. 18 in the form of their three main vectors each one decomposed in their scalar components. Despite the strong disturbance present during the experiment, the rotation matrix tracks the desired rotation matrix with an acceptable tracking error.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

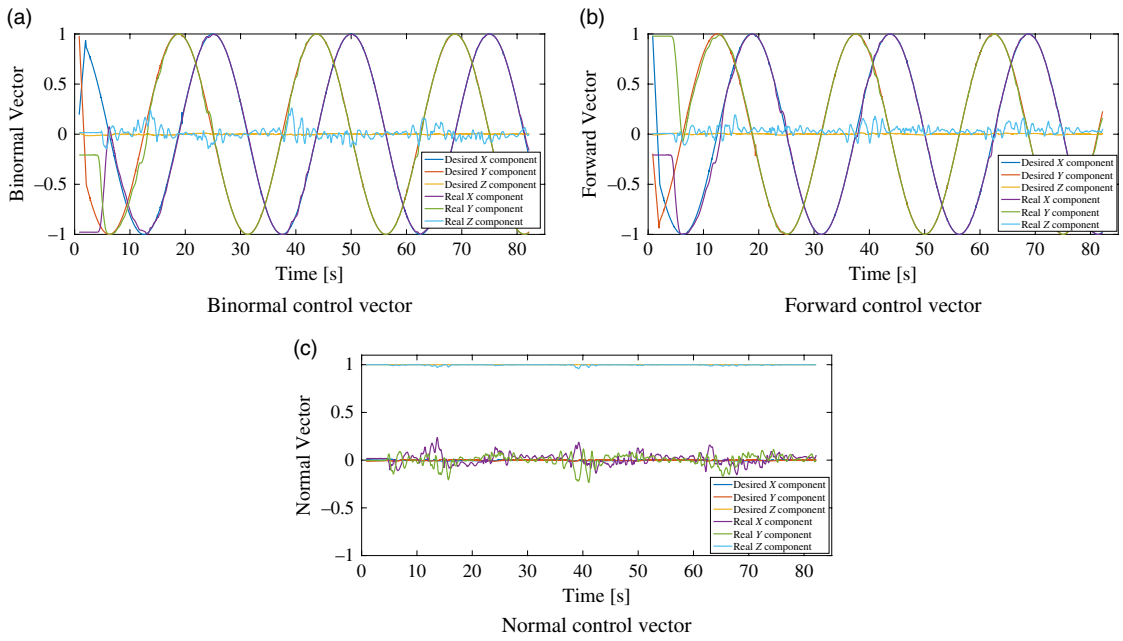

Fig. 18. Control vectors for the disturbed circular trajectory. (a) Binormal control vector. (b) Forward control vector. (c) Normal control vector.

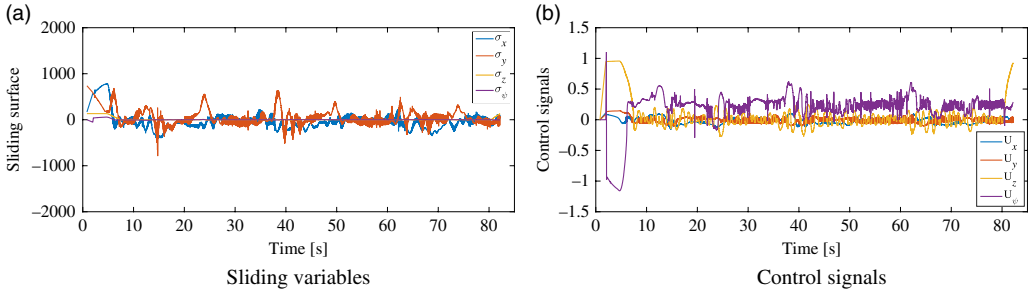

Fig. 19. (a) Sliding variables and (b) control signals for the disturbed circular trajectory.

Thus, in Fig. 19 the sliding variables calculated and the corresponding control signals generated are shown. As in the same way than in the Y-axis displacement, in the sliding variable for y, there are higher values in certain times, and as stated earlier, this is due to the disturbance acting mainly in the Y-axis direction.

In Fig. 20, the experimental results have been plotted for the three axes of movement in the circular trajectory. In this figure, the performance of the proposed algorithm, without any disturbances and with a wind disturbance, highlights the robustness against the disturbance and the feasibility of the proposed strategy.

4.3. Lemniscate trajectory (Infinity symbol)

A lemniscate-like trajectory is performed in this experiment. Geometrically, the lemniscate has a focal distance of 2500 mm and a side elongation of 1500 mm. The vehicle performs the trajectory at a rate of 1.75 laps/min. The real vehicle movement is compared with the ideal trajectory in Fig. 21.

In addition, the three individual-axis trajectory components are shown in Fig. 22, where a correct tracking of the trajectory can be appreciated.

The yaw angle ψ described by the vehicle is shown in Fig. 23. This angle changes tangentially according to the lemniscate trajectory. For the case of this experiment, it is decided to show the angle continuously to simplify the visualization of the graph. It is also noted that the vehicle performs multiple laps, meaning that the angle repeats its cycle.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

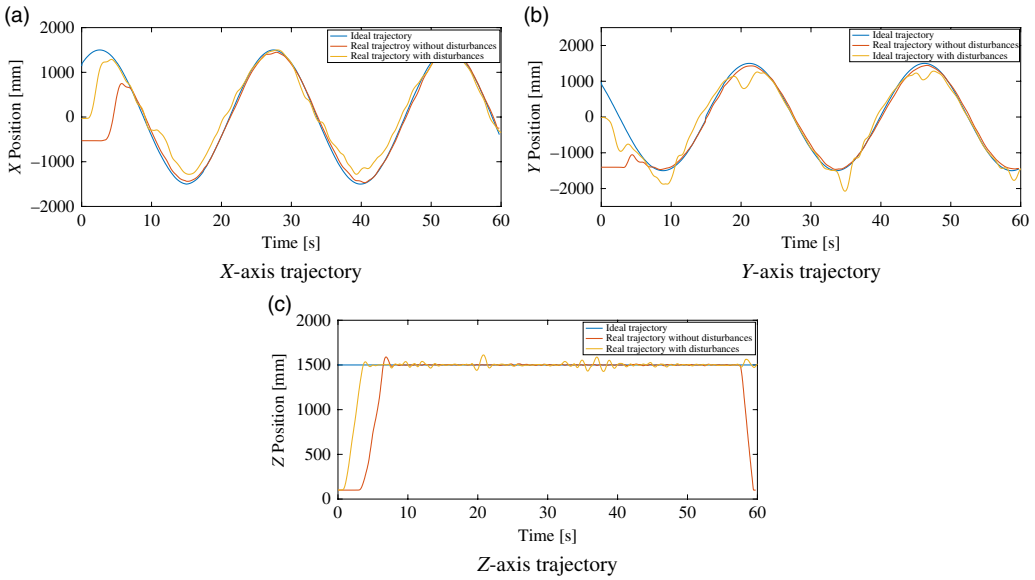

Fig. 20. Experimental results with/without disturbance. (a) X-axis trajectory. (b) Y-axis trajectory. (c) Z-axis trajectory.

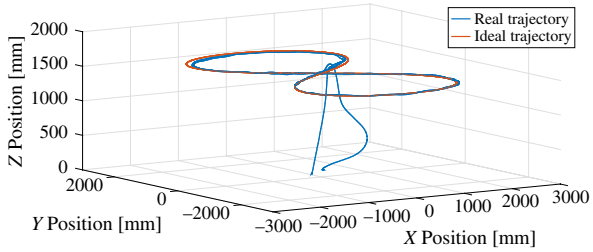

Fig. 21. Lemniscate 3D trajectory.

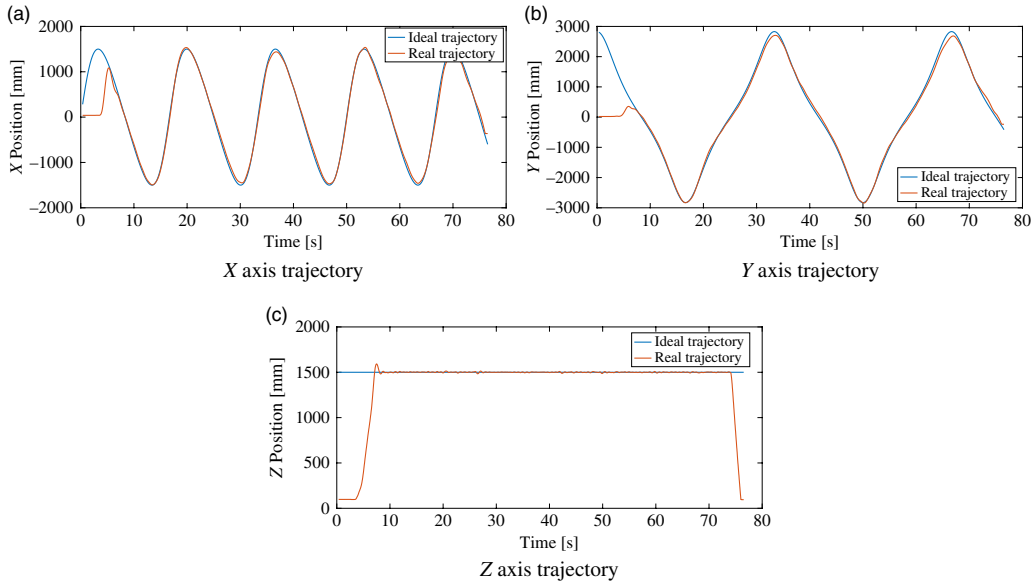

Fig. 22. X, Y, and Z axes components (a)–(c) of the lemniscate trajectory.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

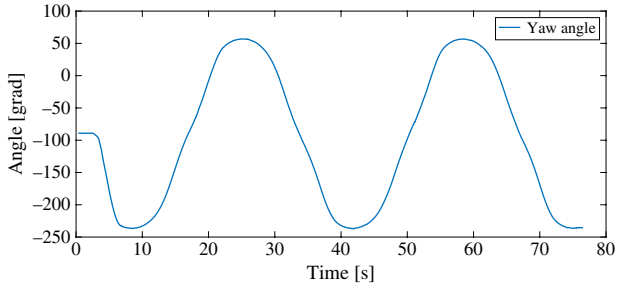

Fig. 23. Yaw angle performed during the lemniscate trajectory.

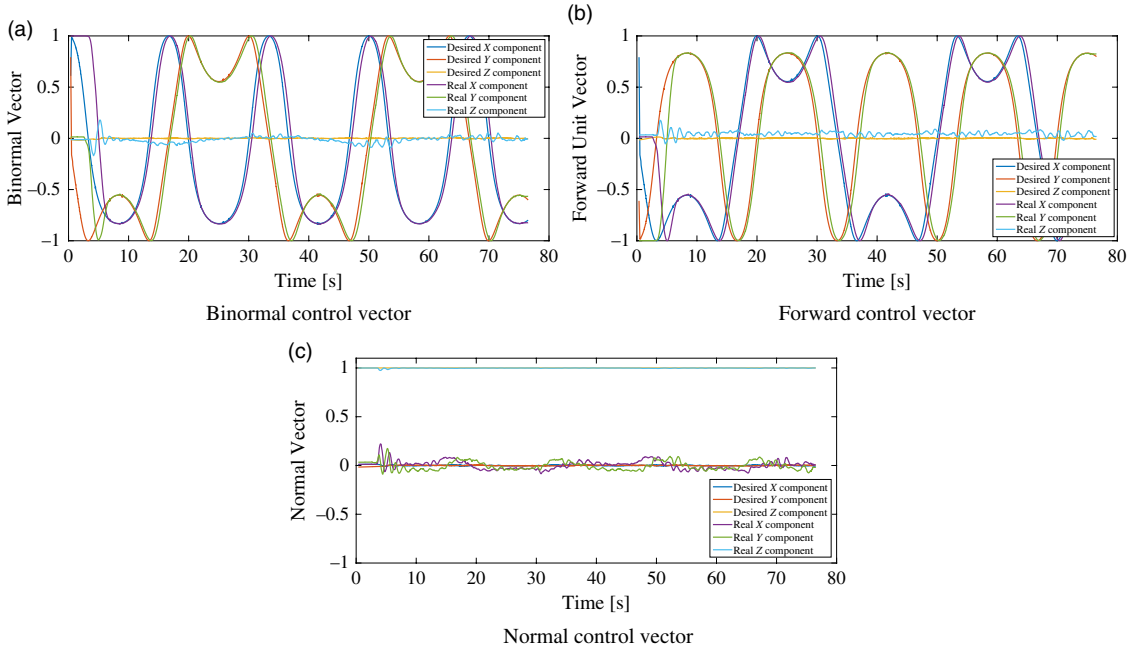

Fig. 24. Control vectors for the lemniscate trajectory. (a) Binormal control vector. (b) Forward control vector. (c) Normal control vector.

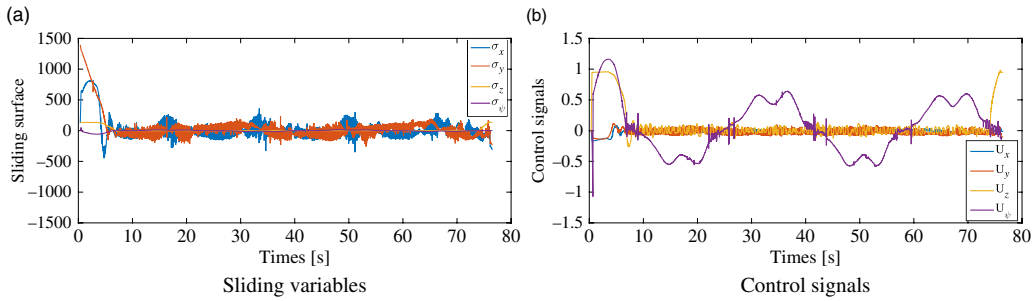

Fig. 25. (a) Sliding variables and (b) control signals for the lemniscate trajectory.

Also for this case, the rotation matrix R and the desired rotation matrix R_d are represented in Fig. 24 decomposed in the scalar components of their three main vectors.

Using these data, the corresponding sliding variables and thus the control signals are calculated and shown in Fig. 25.

4.4. Hover response to a constant disturbance

This experiment is aimed to test the behavior of the controller against a constant disturbance, while in hover flight. This disturbance is produced by a wind gust generated using a fan which is located at coordinates $X = 0$, $Y = 3600$, and $Z = 1500$ mm. These wind gusts are up to 6 m/s (21.6 km/h).

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

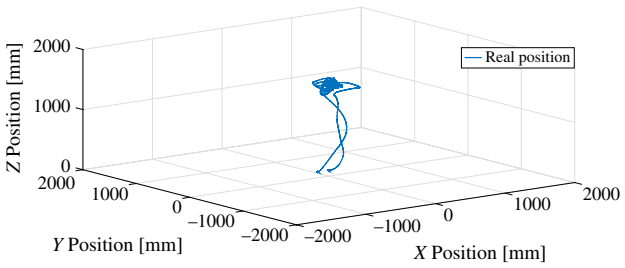

Fig. 26. Disturbed hover flight 3D positions.

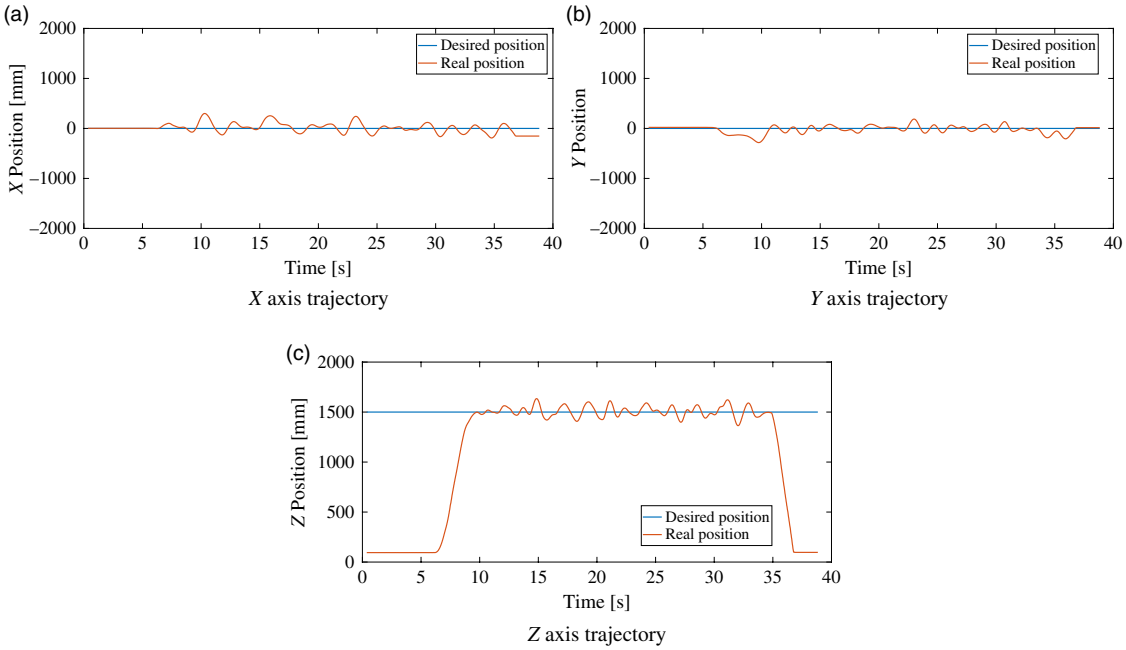

Fig. 27. *X,Y*, and *Z* axes components (a)–(c) of the disturbed hover flight.

During the experiment, the vehicle maintains its position in hover flight even in presence of the wind disturbance, as it can be seen in Fig. 26. As noted in Fig. 27, the errors due to the disturbance are relatively low considering the magnitude of the wind disturbances. In Fig. 28 the maximum error values for each axis are shown.

Due to the persistent wind disturbance, the sliding variable and thus the control signals are being constantly calculated to compensate the induced force. The above mentioned is seen in Fig. 29.

Readers can see the real-time performance of the proposed algorithm on the following link: https://youtu.be/qLK_5IUcmAY

5. Conclusions

In this paper, a robust geometric navigation algorithm for a quadrotor UAV was developed on the special Euclidean group SE(3). The nonlinear dynamic model for a quadrotor UAV was presented using the Newton–Euler formulation. This algorithm of geometric navigation for quadrotor UAV was designed considering a guidance frame, which is proposed to perform the autonomous navigation with an asymptotic convergence to the trajectory. In this sense, the ST algorithm, as a cascade structure, was implemented in order to control the nonlinear rotational and translational dynamics of the quadrotor UAV guaranteeing that the sliding mode is enforced in finite time, and providing robustness against parameter uncertainty and asymptotic convergence of trajectory tracking against disturbances due to gust winds. Finally, experimental results of the autonomous navigation have demonstrated the feasibility and effectiveness of the proposed algorithm.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

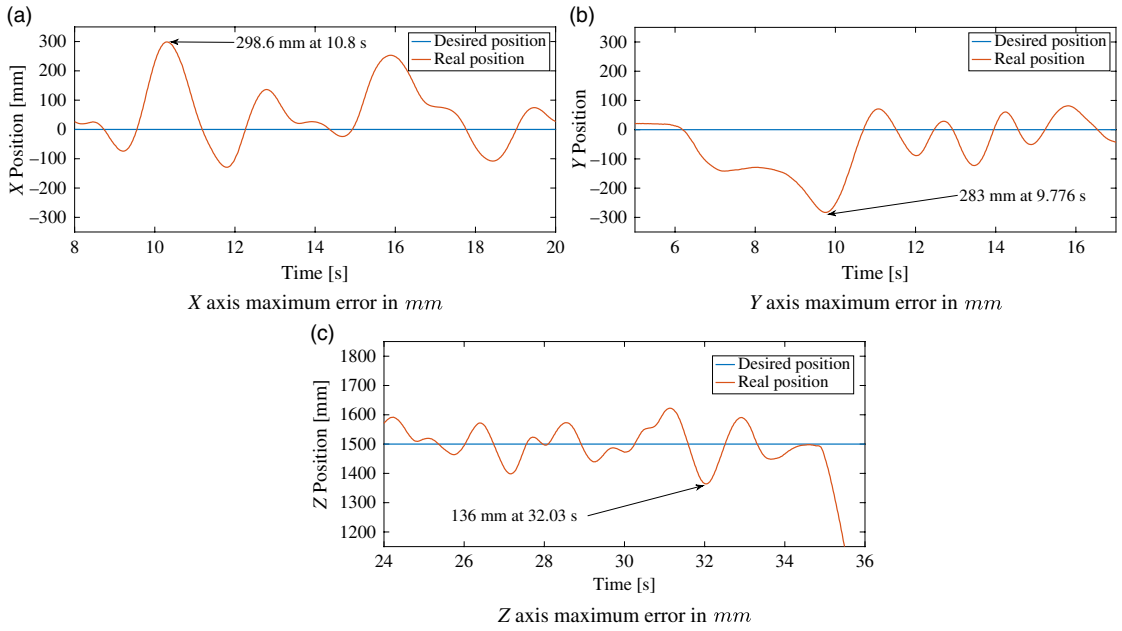

Fig. 28. *X,Y*, and *Z* axes maximum errors (a)–(c) having as reference 0 for *X* and *Y*, and 1500 mm for *Z*.

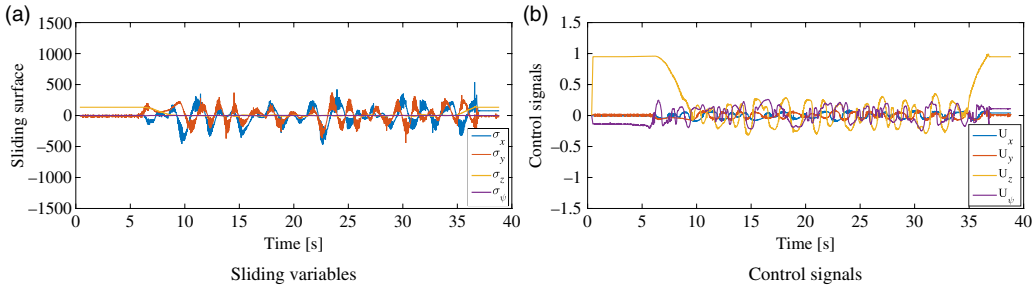

Fig. 29. (a) Sliding variables and (b) control signals for the disturbed hover flight.

Acknowledgments

This work was partially supported by the Mexican National Council for Science and Technology (CONACYT) Mexico with the project No. 204363 and the project No. 1086 – Cátedras CONACYT: *Ambientes Inteligentes*.

References

1. L. A. Amezquita-Brooks, E. Liceaga-Castro, M. Gonzalez-Sanchez, O. Garcia-Salazar and D. Martinez-Vazquez, "A towards a standard design model for quad-rotors: A review of current models, their accuracy and a novel simplified model," *Progress Aerospace Sci.*, Elsevier **95**(8), 1–23 (2017).
2. R. Lozano, *Unmanned Aerial Vehicles Embedded Control* (John Wiley-ISTE Ltd, USA, 2010).
3. A. Sanchez, V. Parra-Vega, O. Garcia, F. Ruiz-Sanchez and L. E. Ramos-Velasco, "Time-Parametrization Control of Quadrotors with a Robust Quaternion-based Sliding Mode Controller for Aggressive Maneuvering," *2013 European Control Conference (ECC)*, Zurich, Switzerland (2013) pp. 3876–3881.
4. T. Lee, M. Leok and N. H. McClamroch, "Dynamics of Connected Rigid Bodies in a Perfect Fluid," *Proceedings of the IEEE American Control Conference (ACC)*, St. Louis, MO (2009).
5. T. Lee, M. Leok and N. H. McClamroch, "Geometric Tracking Control of a Quadrotor UAV on SE(3)," *Proceedings of the 49th IEEE Conference on Decision and Control*, Atlanta, GA (2010) pp. 5420–5425.
6. T. Lee, M. Leok and N. H. McClamroch, "Nonlinear robust tracking control of a quadrotor UAV on SE(3)," *Asian J. Control* **15**(2), 391–408 (2013).
7. J. F. Vasconcelos, R. Cunha, C. Silvestre and P. Oliveira, "A nonlinear position and attitude observer on SE(3) using landmark measurements," *Syst. Control Lett.* **59**(3–4), 155–166 (2010).
8. J. Colorado, A. Barrientos, A. Martinez, B. Lafaverges and J. Valente, "Mini-Quadrotor Attitude Control Based on Hybrid Backstepping and Frenet–Serret Theory," *Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*, Anchorage, AK (2010) pp. 1617–1622.

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231

9. A. K. Sanyal and N. Nordkvist, "Attitude state estimation with multi-rate measurements for almost global attitude feedback tracking," *AIAA J. Guidance Control Dyn.* **35**(3), 868–880 (2012).
10. D. Pylorof and E. Bakolas, "Tracking a Maneuvering Target with an Underactuated UAV in the SE(3) Space," *AIAA Guidance, Navigation, and Control Conference*, AIAA SciTech Forum (2015) pp. 1–12.
11. J. Bohn and A. K. Sanyal, "Almost global finite-time stabilization of rigid body attitude dynamics using rotation matrices," *Int. J. Robust Nonlinear Control* **26**(9), 2008–2022 (2016).
12. S. P. Viswanathan, A. K. Sanyal and E. Samiei, "Integrated guidance and feedback control of underactuated robotics system in SE(3)," *J. Intell. Robot. Syst.* **89**(1–2), 251–263 (2018).
13. S. P. Viswanathan, A. K. Sanyal and M. Izadi, "Integrated Guidance and Nonlinear Feedback Control of Underactuated Unmanned Aerial Vehicles in SE(3)," *AIAA Guidance, Navigation and Control Conference*, AIAA SciTech Forum (2017) pp. 1–12.
14. M. Nazari, M. Maadani, E. A. Butcher and T. Yucelen, "Morse-Lyapunov-Based Control of Rigid Body Motion on TSE(3) via Backstepping," *2018 AIAA Guidance, Navigation and Control Conference*, AIAA SciTech Forum (2018) pp. 1–12.
15. R. Mahony, V. Kumar and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robot. Auto. Mag.* **19**(3), 20–32 (2012).
16. S. de Marco, L. Marconi, T. Hamel and R. Mahony, "Output Regulation on the Special Euclidean Group SE(3)," *IEEE 55th Conference on Decision and Control (CDC 2016)*, Las Vegas, NV (2016).
17. L. N. N. T. Ha, D. H. P. Bui and S. K. Hong, "Nonlinear control for autonomous trajectory tracking while considering collision avoidance of UAVs based on geometric relations," *Energies* **12**(8), 1551 (2019).
18. L. Derafa, A. Benallegue and L. Fridman, "Super twisting control algorithm for the attitude tracking of a four rotors UAV," *J. Franklin Inst.* **349**(2), 685–699 (2012).
19. H. J. Jayakrishnan, "Position and Attitude Control of a Quadrotor UAV Using Super Twisting Sliding Mode," *IFAC 2016* (2016) pp. 284–289.
20. M. Bouchoucha, S. Seghour and M. Tadjine, "Classical and Second Order Sliding Mode Control Solution to an Attitude Stabilization of a Four Rotors helicopter: From Theory to Experiment," *2011 IEEE International Conference on Mechatronics* (ICM), Istanbul, Turkey (2011).
21. F. Munoz, I. Gonzalez-Hernandez, S. Salazar and E. S. Espinoza,"Second order sliding mode controllers for altitude control of a quadrotor UAS: Real-time implementation in outdoor environments," *Neurocomputing* **233**, 61–71 (2017).
22. L. Luque-Vega, B. Castillo-Toledo and A. G. Loukianov, "Robust block second order sliding mode control for a quadrotor," *J. Franklin Inst.* **349**(2), 719–739 (2012).
23. A. Sanchez, V. Parra-Vega, C. Izaguirre and O. Garcia, "Position-yaw tracking of quadrotors," *J. Dyn. Syst. Measurement Control* **137**(6), 1–12 (2015).
24. A. G. Escobar, H. Alazki, J. E. Valenzuela and O. Garcia, "Embedded super twisting control for the attitude of a quadrotor," *IEEE Latin America Trans.* **14**(9), 3974–3979 (2016).
25. D. Mercado, P. Castillo and R. Lozano, "Sliding mode collision-free navigation for quadrotors using monocular vision," **36**(10), 1–17 (2018).
26. R. F. Stengel, *Flight Dynamics* (Princeton University Press, USA, 2004).
27. J. G. Leishman, *Principles of Helicopter Aerodynamics* (Cambridge University Press, USA, 2006).
28. O. Martinez, L. Amezquita-Brooks, E. Liceaga-Castro, O. Garcia and D. Martinez, "Experimental Assessment of Wind Gust Effect on PVTOL Aerial Vehicles Using a Wind Tunnel," *Proceedings of the 2015 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, Ixtapa, Mexico (2015).
29. F. Oliva-Palomo A. Sanchez-Orta, P. Castillo and H. Alazki, "Nonlinear ellipsoid based attitude control for aggressive trajectories in a quadrotor: Closed-loop multi-flips implementation," *Control Eng. Pract.* **77**(8), 150–161 (2018).
30. J. P. Aubin and A. Cellina, *Differential Inclusions: Set-Valued Maps and Viability Theory* (Springer Science & Business Media, Germany, 2012).
31. J. A. Moreno and M. Osorio, "Strict Lyapunov functions for the super-twisting algorithm," *IEEE Trans. Auto. Control* **57**(4), 1035–1040 (2012).
32. F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems: Modeling, Analysis, and Design for Simple Mechanical Control Systems* (Springer, New York, 2005).
33. E. Frazzoli, M. A. Dahleh and E. Ferron, "Trajectory Tracking Control Design for Autonomous Helicopters Using a Backstepping Algorithm," *Proceedings of the IEEE American Control Conference (ACC)*, Chicago IL (2000) pp. 4102–4107.
34. R. Seeber and M. Horn, "Stability proof for a well-established super-twisting parameter setting," *Automatica* **84**, 241–243 (2017).
35. P. J. Bristeau, F. Callou, D. Vissiere and N. Petit, "The Navigation and Control Technology Inside the AR.Drone Micro UAV," *Proceedings of the 18th IFAC World Congress*, Milano, Italy (2011) pp. 1477–1484.
36. F. Munoz Palacios, E. S. Espinoza Quesada, G. Sanahuja, S. Salazar, O. Garcia Salazar and L. R. Garcia Carrillo, "Test bed for applications of heterogeneous unmanned vehicles," *Int. J. Adv. Robot. Syst. (IJARS)* **14**(1), 1–14 (2017).

Downloaded from https://www.cambridge.org/core. UNIVERSIDAD AUTONOMA DE NUEVO LEON, on 05 Aug 2019 at 14:50:11, subject to the Cambridge Core terms of use, available at https://www.cambridge.org/core/terms. https://doi.org/10.1017/S0263574719001231