Wind rejection via quasi-continuous sliding mode technique to control safely a mini drone

Gabriele Perozzi, Denis Efimov, Jean-Marc Biannic, Laurent Planckaert, Patricia Coton

Wind rejection via quasi-continuous sliding mode technique to control safely a mini drone

Gabriele Perozzi[⋆†]*, Denis Efimov*[⋆⋆]*, Jean-Marc Biannic*[⋆⋆⋆]*, Laurent Planckaert*[⋆] *and Patricia Coton*[⋆]
[⋆]*ONERA - The French Aerospace Lab, DAAA, F-59014 Lille, France,*
5 Bd Paul Painlevé BP 21261 59014 Lille Cedex France.
[⋆]*Inria, Non-A team,*
Parc Scientifique de la Haute Borne, 40 avenue Halley, Bât. A, Park Plaza, 59650 Villeneuve d'Ascq France.
[⋆]*ONERA - The French Aerospace Lab, DTIS, F-31055 Toulouse, France,*
BP74025 2 avenue Edouard Belin 31055 Toulouse Cedex 4 France.
gabriele.perozzi@onera.fr · denis.efimov@inria.fr · jean-marc.biannic@onera.fr
laurent.planckaert@onera.fr · patricia.coton@onera.fr
[†]Corresponding author

Abstract

The objective of this paper is to show how to build a nonlinear robust control law, which ensures trajectory tracking for a drone quadrotor under unpredictable wind perturbations. The first step is to find the aerodynamic forces and moments using a combination of momentum and blade element theory. Then the model is rewritten in state-space form, where the control inputs are selected to be proportional to the squares of rotor angular velocities. The other terms dependent linearly on rotors and wind velocities are considered as disturbances. Such a decomposition of thrust and selection of disturbances are almost exact in the hover flight. In literature, fixed bounds are often assumed on each component of the disturbance input vector, but for synthesis of the proposed control law, the big issue is that the disturbance depends on wind signals, the control itself, and state of the system. Chattering effects and their reduction are analysed and investigated in the last part of the paper by introducing rotors dynamics in control design. High order sliding mode control is applied and the recent tool of quasi-continuous sliding mode control is analyzed. Results of numeric experiments demonstrate the effectiveness of the proposed controls.

1. Introduction

Small unmanned aerial vehicles are becoming very useful tools for both military and commercial applications. They are used in different domains such as monitoring and inspection in urban areas or nearby buildings and dangerous interiors, for natural calamities such as earthquakes. These robots are often required to move in unfamiliar environments in terms of geography and in terms of aerological conditions. In addition, since the low mass of such units (comparing to the forces generated by the air disturbances) reduces significantly the domain of stable flight, additional constraints have to be considered in the control design. Thus if we want to let unmanned aerial vehicles (UAVs) operate in urban environments, inside turbulent air flow patterns for which accurate prediction is not possible *a priori* with limited resources, we need to focus on detailed aerodynamic models and sophisticated control laws. This paper is a part of project at Onera lab of Lille, which aims to use the drone itself as a wind sensor, and this research is carried out in parallel with development of a wind estimation tool by Perozzi et al[1] . The considered problem consists in design of a nonlinear robust control law, which ensures a stable and efficient navigation of a small UAV under unpredictable wind perturbations. The model described in this article can be used to estimate wind velocity, in this way the estimated values can themselves be used as inputs to the control to properly adapt the gains on-line and use this estimation in path planning and trajectory control to smartly avoid collisions. In literature, the quadrotor physical models with different grades of complexity are presented and used for various tasks. For example, to mention the most relevant in our study, in Sydney et al[2] , aerodynamic moments and forces varying in time affected by the perturbations are considered, then linear PID controllers are applied. A deeper study of the dynamic equations than the previous ones is given in Bouadballah et al[3] , but it does not take into account the wind effects. There exist many control design techniques to counteract the effects of wind perturbations on flight of small UAVs, among which sliding mode control (SMC) plays a keyrole. The main reason consists in its excellent robustness properties against matched perturbations and measurement

noises, and finite-time rate of convergence also. Many methods have been proposed in the literature, for instance, some principal SMCs with their relative sliding surfaces and Lyapunov functions are illustrated in Bernuau et al[4] . Dozens of articles have applied SMC to UAVs in order to solve the position and the attitude tracking problems ensuring robustness against external disturbances. Just to mention some of them, a chattering-free sliding mode controller was proposed in Gonzalez at al[5] , where the sign function is replaced with a high-slope saturation. A sliding mode controller based on the backstepping approach was introduced in Madani et al[6] , and an enhanced version was proposed in Zeghlache et al[7] using fuzzy logic. An algorithm based on a higher order sliding mode control (HOSMC) technique, known as super-twisting algorithm, was used in Derafa et al[8] and in Rajappa et al[9] . Second order sliding mode controller was also proposed in Zheng et al[10] , and Benallegue et al[11] used a HOSM observer as an estimator of the effect of the external disturbances such as wind and noise. Integral sliding mode controller was proposed in Zhang et al[12] , and an adaptive sliding mode controller was developed in Bouadi et al[13] .
In this paper a robust nonlinear HOSMC law design is described, which considers realistic assumptions on external disturbances of quadrotors. In the considered case the upper bound of matched disturbances depends nonlinearly on the control itself, the system state vector and wind disturbances. The closed-loop system stability is ensured for a selected maximum admissible value of the wind speed. The control strategy proposed in this article can be equipped with an additional wind estimator algorithm reducing automatically the control effort on the rotors when it is not necessary. The paper outline is as follows. In Section 2 the considered UAV is described and the flight dynamics model is derived. The control and disturbance bounds are calculated in Section 3. The control design is presented in Section 4. The results of numeric experiments are shown in Section 5. The experimental setup and the drone are described in Section 6. The remarks and discussion conclude the paper in Section 7.

2. Quadrotor dynamical model

This section presents the model of the UAV dynamics, which has configuration as it is shown in Fig. 1, and which is used for the Onera project to estimate the wind velocity, making the drone as a wind sensor.

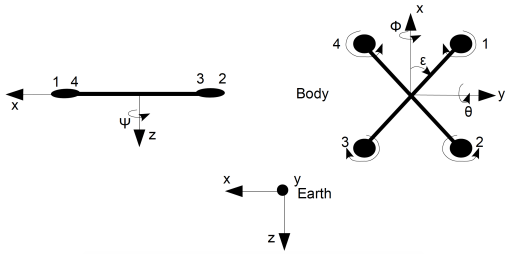

Figure 1: Drone configuration

According to the identification work at low speed in Planckaert et al[14] , rotors gyroscopic effects and inertial counter torques are neglected since they are rather small. The dynamics of the drone in the body frame yield

$$\begin{aligned} m\dot{\mathbf{u}} &= -m\boldsymbol{\varpi} \times \mathbf{u} + \mathbf{F}_{\mathbf{aero}} + m\mathbf{g}^T\mathbf{R}, \\ \mathbf{I}\dot{\boldsymbol{\varpi}} &= -\boldsymbol{\varpi} \times \mathbf{I}\boldsymbol{\varpi} + \tau_{\mathbf{aero}}, \end{aligned} \tag{1}$$

where m is the mass of the UAV, $\mathbf{u} = [u\ v\ w]^T$ is its linear velocity expressed in body frame, $\boldsymbol{\varpi} = [p\ q\ r]^T$ is its angular velocity in body frame, $\mathbf{F}_{\mathbf{aero}} = [F_{Xaero}\ F_{Yaero}\ F_{Zaero}]^T$ is the vector of the external aerodynamic forces in body frame, $\mathbf{g} = [0\ 0\ g]^T$ is the gravity acceleration in inertial frame, $\mathbf{I}$ is the inertia matrix of the UAV, $\tau_{\mathbf{aero}} = [L_{aero}\ M_{aero}\ N_{aero}]^T$ is the external aerodynamic moments in the body frame, $\mathbf{R}$ is the rotational matrix defined as

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

where $c_\psi = \cos(\psi)$, $s_\psi = \sin(\psi)$ and similarly. The relationship equations between angular velocities and Euler angles

$$\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\phi),\ \dot{\theta} = q\cos\phi - r\sin\phi,\ \dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta}, \tag{2}$$

are considered avoiding the singularities $\theta \neq \frac{\pi}{2}$, which is a reasonable assumption in our case since the topic of this article is not to achieve aggressive maneuvers.

2.1 Aerodynamics

The aerodynamic forces, moments, and coefficients are derived using a combination of momentum and blade element theory in helicopters, well explained by Johnson[15] , Bramwell et al[16] , and Leishman[17] . Aerodynamic forces and moments for each rotor are derived as

$$\begin{aligned}
F_{Xj} &= -\rho A R^2 \frac{u_j - u_w}{\sqrt{(u_j - u_w)^2 + (v_j - v_w)^2}} C_{Hj} \omega_j^2, \\
F_{Yj} &= -\rho A R^2 \frac{v_j - v_w}{\sqrt{(u_j - u_w)^2 + (v_j - v_w)^2}} C_{Hj} \omega_j^2, \\
F_{Zj} &= -\rho A R^2 C_{Tj} \omega_j^2, \\
L_j &= -\operatorname{sign}\omega_j \rho A R^3 \frac{u_j - u_w}{\sqrt{(u_j - u_w)^2 + (v_j - v_w)^2}} C_{Rmj} \omega_j^2, \\
M_j &= -\operatorname{sign}\omega_j \rho A R^3 \frac{v_j - v_w}{\sqrt{(u_j - u_w)^2 + (v_j - v_w)^2}} C_{Rmj} \omega_j^2, \\
N_j &= -\operatorname{sign}\omega_j \rho A R^3 C_{Qj} \omega_j^2,
\end{aligned} \tag{3}$$

where subscript j indicates the j^{th} rotor, ρ is the air density, A is the rotor area, R is the rotor radius, $[u_w\ v_w\ w_w]$ is the wind velocity with respect to the earth in body frame, C_H is the hub force coefficient, C_T is rotor thrust coefficient, ω is the rotor angular speed, C_Q is the rotor drag moment coefficient, C_{Rm} is the rotor rolling moment coefficient. Total aerodynamic forces are

$$F_{Xaero} = \sum_{j=1}^{4} F_{Xj},\ F_{Yaero} = \sum_{j=1}^{4} F_{Yj},\ F_{Zaero} = \sum_{j=1}^{4} F_{Zj}.$$

Total aerodynamic moments are

$$L_{aero} = \sum_{j=1}^{4}(L_j + F_{Zj} l \mathrm{s}_j - h F_{Yj}),\ M_{aero} = \sum_{j=1}^{4}(M_j - F_{Zj} l \mathrm{c}_j + h F_{Xj}),\ N_{aero} = \sum_{j=1}^{4}(N_j + F_{Yj} l \mathrm{c}_j - F_{Xj} l \mathrm{s}_j),$$

where h is the distance between rotors plane and the center of gravity of the UAV, l is the arm lenght, and with $\mathrm{c}_j = \cos\left(\frac{\pi}{2}(j-1) + \epsilon\right),\ \mathrm{s}_j = \sin\left(\frac{\pi}{2}(j-1) + \epsilon\right)$. In our UAV configuration we have $\epsilon = \frac{\pi}{4}$, thus for vectors c_j and s_j we have cosines and sinus of the angles $[\frac{\pi}{4}, \frac{3}{4}\pi, \frac{5}{4}\pi, \frac{7}{4}\pi]$. According to Planckaert et al[14] and thanks to direct experiments on the drone at low speed, the aerodynamic coefficients are identified as follows

$$\begin{aligned}
C_{Tj} &= C_{T\,stat} + K_z \frac{w_j - w_w}{R|\omega_j|}, \quad C_{T\,stat} > 0, \\
\lambda_j &= \lambda_{stat} - \frac{4}{\sigma a} K_z \frac{w_j - w_w}{R|\omega_j|}, \quad K_z \geq 0,\ \lambda_{stat} > 0, \\
\mu_j &= \frac{1}{|\omega_j| R} \sqrt{(u_j - u_w)^2 + (v_j - v_w)^2}, \\
C_{Rmj} &= \sigma a \frac{\mu_j}{8}(\lambda_j - \frac{4}{3}), \\
C_{Qj} &= \frac{\sigma C_{D0}}{8}(1 + \mu_j^2) + \sigma a \lambda_j (\frac{\theta_0}{6} - \frac{\lambda_j}{4}), \\
C_{Hj} &= K_D \mu_j. \quad K_D \geq 0
\end{aligned} \tag{4}$$

where σ is the rotor solidity ratio, a is the lift curve slope of the blade section, C_{D0} is the drag coefficient of the blade section, θ_0 is the angle of attack of the root profile, λ is the inflow ratio, μ is the advance ratio, subscript *stat* indicates the value in stationary phase. The shape of above UAV coefficients can be explained recalling aerodynamic science: the thrust is the resultant of the vertical forces acting on all the blade elements. The inflow ratio is the ratio between the component of UAV velocity perpendicolar to the rotor disk with respect to the blade tip speed. The advance ratio indicates the component of the UAV velocity parallel to the rotor disk with respect to the blade tip speed. The rolling moment of a propeller exists in forward flight when the advancing blade is producing more lift than the retreating one and it is the integration over the entire rotor of the lift of each section acting at a given radius. The hub forces is the resultant of the horizontal forces acting on all the blade elements. The drag moment about the rotor shaft is caused by the aerodynamic forces acting on the blade elements, the horizontal forces acting on the rotor are multiplied by the moment arm and integrated over the rotor.

3. Control system equations

The model presented in previous section can be rewritten in the state-space form

$$\dot{X} = f(X,\, U,\, d),$$

where f is expressed in (1), (2) and the state vector X is chosen as

$$X = [x \quad y \quad z \quad \dot{x} \quad \dot{y} \quad \dot{z} \quad \phi \quad \theta \quad \psi \quad p \quad q \quad r]^{\,T},$$

$U = [U_z\ U_\theta\ U_\phi\ U_\psi]^T$ is the control input, and disturbances d are described in the next subsection. The relations between the control inputs and the rotor velocities are defined by an invertible matrix

$$\begin{bmatrix} U_z \\ U_\theta \\ U_\phi \\ U_\psi \end{bmatrix} = \begin{bmatrix} K_f & K_f & K_f & K_f \\ K_f l\mathrm{c}_j & K_f l\mathrm{c}_j & K_f l\mathrm{c}_j & K_f l\mathrm{c}_j \\ -K_f l\mathrm{s}_j & -K_f l\mathrm{s}_j & -K_f l\mathrm{s}_j & -K_f l\mathrm{s}_j \\ K_m & -K_m & K_m & -K_m \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix},$$

with $\omega_{min} \le \omega_j \le \omega_{max}$, where $K_f = \rho A R^2 C_{T\,stat}$ and $K_m = \rho A R^3 (\frac{\sigma C_{D0}}{8} + \lambda_{stat}\sigma a(\frac{\theta_0}{6} - \frac{\lambda_{stat}}{4}))$. In this work, the control inputs are selected to be proportional to the terms with ω_j^2. Thus, expanding (1), the other terms dependent linearly on ω_j and wind velocities are considered as disturbances. Since we do not know in advance the wind perturbations, then we cannot use these terms in controls. Such a decomposition of thrust (which is proportional to ω_j and ω_j^2) and selection of disturbances are almost exact in the hover flight, where we have $(p,\, q,\, r) \approx (\dot{\phi},\, \dot{\theta},\, \dot{\psi})$.
In the following, linear rotors speed in body frame are computed as a function of the state in body frame

$$\begin{bmatrix} u_j \\ v_j \\ w_j \end{bmatrix} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} lc_j \\ ls_j \\ h \end{bmatrix} + \begin{bmatrix} u \\ v \\ w \end{bmatrix}.$$

3.1 Definition of disturbances

In the literature, it is often assumed that each component of the disturbance input vector d admits a fixed upper bound, which means $|d| \le D$ for some known $D \ge 0$. Unfortunately, it is a rather conservative hypothesis, and that is why the varying state-dependent bounds will be considered in our case for d. However, we will assume boundedness of the wind velocities: $|u_w| \le D_x$, $|v_w| \le D_y$, $|w_w| \le D_z$, for some known $D_x \ge 0$, $D_y \ge 0$, $D_z \ge 0$, which is a reasonable restriction. Substituting (4), (3) in (1) and performing calculations, we have the disturbance $d = [d_x, d_y, d_z, d_\phi, d_\theta, d_\psi]$ represented by the following terms

$$d_x = \sum_{j=1}^{4} -\rho A R K_D \left(u_j - u_w\right)|\omega_j|, \tag{5}$$

$$d_y = \sum_{j=1}^{4} -\rho A R K_D \left(v_j - v_w\right)|\omega_j|, \tag{6}$$

$$d_z = \sum_{j=1}^{4} -\rho A R K_z \left(w_j - w_w\right)|\omega_j|, \tag{7}$$

$$\begin{aligned} d_\phi = \sum_{j=1}^{4} \Big(&\omega_j \rho A R^2 (u_j - u_w)\frac{\sigma a}{2}(\frac{\theta_0}{3} - \frac{\lambda_{stat}}{4}) + |\omega_j|\rho A R(hK_D\left(v_j - v_w\right) - lK_z\left(w_j - w_w\right)\mathrm{s}_j) \\ &+ \mathrm{sign}\left(\omega_j\right)\frac{1}{2}\rho A R K_z\left(u_j - u_w\right)\left(w_j - w_w\right)\Big), \end{aligned} \tag{8}$$

$$\begin{aligned} d_\theta = \sum_{j=1}^{4} \Big(&\omega_j \rho A R^2 \left(v_j - v_w\right)\frac{\sigma a}{2}(\frac{\theta_0}{3} - \frac{\lambda_{stat}}{4}) + |\omega_j|\rho A R(-hK_D\left(u_j - u_w\right) + lK_z\left(w_j - w_w\right)\mathrm{c}_j) \\ &+ \mathrm{sign}\left(\omega_j\right)\frac{1}{2}\rho A R K_z\left(v_j - v_w\right)\left(w_j - w_w\right)\Big), \end{aligned} \tag{9}$$

$$d_\psi = \sum_{j=1}^{4}\Big(\omega_j\rho AR^2K_z\left(w_j - w_w\right)(\frac{2\theta_0}{3} - 2\lambda_{stat}) - |\omega_j|\rho lARK_D(\left(v_j - v_w\right)\mathrm{c}_j - \left(u_j - u_w\right)\mathrm{s}_j)$$
$$- \operatorname{sign}\left(\omega_j\right)\rho AR\Big(\frac{\sigma C_{D0}}{8}(\left(u_j - u_w\right)^2 + (v_j - v_w)^2) - \frac{4}{\sigma a}K_z^2\left(w_j - w_w\right)^2\Big)\Big). \tag{10}$$

3.2 Disturbance upper bounds

To design a control, which is able to compensate the disturbances, we have to evaluate the upper bounds for them.

3.2.1 Disturbance in x dynamics

From the equation (5)

$$|d_x| = |\tilde{K}_D\sum_{j=1}^{4}\left(u_j - u_w\right)\omega_j| \le \tilde{K}_D\sum_{j=1}^{4}|u_j - u_w||\omega_j| \le \tilde{K}_D\sum_{j=1}^{4}\left(|u_j| + |u_w|\right)|\omega_j| \le \tilde{K}_D(\max_j|u_j| + |u_w|)\sum_{j=1}^{4}|\omega_j|,$$

where $\tilde{K}_D = \rho ARK_D$. Using the control equation

$$U_z = K_f\sum_{j=1}^{4}\omega_j^2$$

and applying the Jensen's inequality, an upper estimate can be obtained

$$\sum_{j=1}^{4}|\omega_j| \le K\sqrt{|U_z|}, \qquad K = \frac{2}{\sqrt{K_f}}. \tag{11}$$

An upper bound of the disturbance becomes

$$|d_x| \le \bar{K}_D\left(|X| + D_x\right)\sqrt{|U_z|} = d_{xx}, \tag{12}$$

where $\bar{K}_D = K\tilde{K}_D$. In the earth frame using the rotation matrix and the upper bound for the disturbances (12), (13), (14) it becomes

$$d_{xe} = c_\psi c_\theta d_x + (c_\psi c_\theta c_\phi - s_\psi c_\phi)d_y + (s_\phi s_\psi + c_\psi s_\theta c_\phi)d_z,\ |d_{xe}| \le d_{xx} + 2d_{yy} + 2d_{zz}$$

For the disturbances d_{ye} and d_{ze} the computations are similar to the previous ones, therefore only final expressions are given below.

3.2.2 Disturbance in y dynamics

From equation (6), we have the upper bounds in body and earth frames respectively

$$|d_y| \le \bar{K}_D\left(|X| + D_y\right)\sqrt{|U_z|} = d_{yy},\ |d_{ye}| \le d_{xx} + 2d_{yy} + 2d_{zz}. \tag{13}$$

3.2.3 Disturbance in z dynamics

From equation (7), we have the upper bounds in body and earth frames respectively

$$|d_z| \le \bar{K}_z\left(|X| + D_z\right)\sqrt{|U_z|} = d_{zz},\ |d_{ze}| \le \left(f_{ze}\left(|X|\right) + D_{ze}\right)\sqrt{|U_z|}. \tag{14}$$

where $\bar{K}_z = K\rho ARK_z$, $f_{ze}(|X|) = \bar{K}_z\max_j|w_j| + \bar{K}_D\max_j|u_j| + \bar{K}_D\max_j|v_j|$, $D_{ze} = \bar{K}_z|w_w| + \bar{K}_D|u_w| + \bar{K}_D|v_w|$.

3.2.4 Disturbance in roll dynamics

From the equations (8) and (11), the following upper estimate can be computed

$$|d_\phi| \le \tilde{K}_\phi\left(f_{\phi 1}\left(X\right) + D_{\phi 1}\right)\sqrt{|U_z|} + \bar{K}_\phi\left(f_{\phi 2}\left(X\right) + D_{\phi 2}\right)$$

where $K_{\phi 1} = \rho AR^2\frac{\sigma a}{2}(\frac{\theta_0}{3} - \frac{\lambda_{stat}}{4})$, $K_{\phi 2} = \rho ARhK_D$, $K_{\phi 3} = lK_z\rho AR$, $\tilde{K}_\phi = K$, $f_{\phi 1}(X) = K_{\phi 1}\max_j|u_j| + K_{\phi 2}\max_j|v_j| + K_{\phi 3}\max_j|w_j\mathrm{s}_j|$, $D_{\phi 1} = K_{\phi 1}D_x + K_{\phi 2}D_y + K_{\phi 3}D_z\max_j|\mathrm{s}_j|$, $\bar{K}_\phi = \frac{1}{2}\rho ARK_z$, $f_{\phi 2}(X) = \max_j|u_j|^2 + \max_j|w_j|^2$, $D_{\phi 2} = D_x^2 + D_z^2$.

3.2.5 Disturbance in pitch dynamics

From the equations (9) and (11), the following upper estimate can be computed

$$|d_\theta| \leq \tilde{K}_\theta \left(f_{\theta 1}(X) + D_{\theta 1}\right) \sqrt{|U_z|} + \bar{K}_\theta \left(f_{\theta 2}(X) + D_{\theta 2}\right)$$

where $K_{\theta 1} = \rho A R^2 \frac{\sigma a}{2}(\frac{\theta_0}{3} - \frac{\lambda_{stat}}{4})$, $K_{\theta 2} = \rho A R h K_D$, $K_{\theta 3} = l K_z \rho A R$, $\tilde{K}_\theta = K$, $f_{\theta 1}(X) = K_{\theta 1} \max_j |v_j| + K_{\theta 2} \max_j |u_j| + K_{\theta 3} \max_j |w_j \mathrm{c}_j|$, $D_{\theta 1} = K_{\theta 1} D_y + K_{\theta 2} D_x + K_{\theta 3} D_z \max_j |\mathrm{c}_j|$, $\bar{K}_\theta = \frac{1}{2}\rho A R K_z$, $f_{\theta 2}(X) = \max_j |v_j|^2 + \max_j |w_j|^2$, $D_{\theta 2} = D_y^2 + D_z^2$.

3.2.6 Disturbance in yaw dynamics

From the equations (10) and (11), the following upper estimate can be computed

$$|d_\psi| \leq \tilde{K}_\psi \left(f_{\psi 1}(X) + D_{\psi 1}\right) \sqrt{|U_z|} + \bar{K}_\psi \left(f_{\psi 2}(X) + D_{\psi 2}\right)$$

where $K_{\psi 1} = \rho A R^2 K_z(\frac{2\theta_0}{3} - 2\lambda_{stat})$, $K_{\psi 2} = \rho A R l K_D$, $K_{\psi 3} = \rho A R l K_D$, $\tilde{K}_\psi = K$, $f_{\psi 1}(X) = K_{\psi 1} \max_j |w_j| + K_{\psi 2} \max_j |v_j \mathrm{c}_j| + K_{\psi 3} \max_j |u_j \mathrm{s}_j|$, $D_{\psi 1} = K_{\psi 1} D_z + K_{\psi 2} D_y \max_j |\mathrm{c}_j| + K_{\psi 3} D_x \max_j |\mathrm{s}_j|$, $\bar{K}_\psi = \rho A R$, $f_{\psi 2}(X) = \frac{\sigma C_{D0}}{8}(\max_j |u_j|^2 + \max_j |v_j|^2) + \frac{4}{\sigma a} \max_j |w_j|^2$, $D_{\psi 2} = \frac{\sigma C_{D0}}{8}(D_x^2 + D_y^2) + \frac{4}{\sigma a} D_z^2$.

4. Control design

For synthesis of the control law, the quasi-continuous control will be applied which can be considered as an approximation of the sign on the plane, using a recent tool by Ding et al[18] . This control methodology takes into account and compensates the matched disturbances. The big issue for the considered problem is that the disturbance d depends on wind signals, the control itself, and state of the system, as shown above. Thus, a mild development of SMC approach is needed (see the Appendix for the explanation of the idea of HOSMCs). To this end, the sliding surfaces in this work are selected proportional to the regulation errors e_i, in this way we are going to control the dynamics proportional to position and velocity

$$S_i = \dot{e}_i + \alpha_i e_i, \quad \alpha_i > 0,$$

where $i \in (x, y, z, \phi, \theta, \psi)$. Usually in the control design it is assumed that the rotors possess an immediate response on the desired values ω_j assigned to them by control law. In reality they admit some dynamics, and for design of the proposed control, the transfer functions for the rotors are introduced in a generic and realistic form

$$\frac{L_j}{U_j} = \frac{1}{1 + b\,s} \rightarrow b\,\dot{L}_j = U_j - L_j, \tag{15}$$

where $b = 0.1$, $j \in (z, \phi, \theta, \psi)$ corresponding to the controls, L_j are the controls subjected to the delay caused by rotors dynamics.

4.1 Position control

4.1.1 z control

Recall that for z dynamics, the second derivative of the position error, $e_z = z - z_{des}$, has the form

$$\ddot{e}_z = g - \frac{\cos\theta \cos\phi}{m}(L_z + d_{ze}) - \ddot{z}_{des},$$

where the control U_z is substituted by rotor dynamic outputs from (15). The second derivative of the position error can be rewritten

$$\ddot{e}_z = -L_z \delta_z + \Delta_z,$$

where $\delta_z = (\cos\theta\cos\phi)/m$, $\Delta_z = g - \ddot{z}_{des} - (\cos\theta\cos\phi\, d_{ze})/m$. The first derivative of the sliding surface is obtained

$$\dot{S}_z = \ddot{e}_z + \alpha_z \dot{e}_z = -L_z \delta_z + \Delta_z + \alpha_z \dot{e}_z.$$

Using (15), its second derivative is computed as

$$\ddot{S}_z = -\frac{U_z - L_z}{b}\delta_z + \dot{\delta}_z L_z + \dot{\Delta}_z + \alpha_z(-\delta_z L_z + \Delta_z) = -U_z \frac{\delta_z}{b} + \tilde{d}_z,$$

where $\tilde{d}_z = \dot{\Delta}_z + \alpha_z(-\delta_z L_z + \Delta_z) + (L_z\delta_z)/b + \dot{\delta}_z L_z$.
In Ding et al[18] it is stated that if the second order control is selected as

$$U_z = \frac{b}{\delta_z}\tilde{D}_z(t,X)\frac{\lceil \dot{S}_z \rfloor^2 + S_z}{|\dot{S}_z|^2 + |S_z|},$$

then $S_z = \dot{S}_z = 0$ is reached in a finite time, with $\tilde{D}_z(t,X) > |\tilde{d}_z|$. See the Appendix for the explanation of its idea.

4.1.2 x, y controls

Since x, y dynamics cannot be directly controlled then the following virtual controls

$$U_x = \sin\phi\sin\psi + \cos\psi\sin\theta\cos\phi,\ U_y = \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi, \tag{16}$$

are introduced, which will be used to find the desired Euler angles ϕ_{des}, θ_{des} that will be the inputs of attitude controller next. The position errors, $e_x = x - x_{des}$, $e_y = y - y_{des}$, have the dynamics

$$\ddot{e}_x = U_x\frac{U_z}{m} - d_{xe} - \ddot{x}_{des},\ \ddot{e}_y = U_y\frac{U_z}{m} - d_{ye} - \ddot{y}_{des}.$$

In Ding et al[18] it is stated that if

$$\ddot{e}_x = -\tilde{D}_x(t,X)\frac{\lceil \dot{e}_x \rfloor^2 + \alpha_x e_x}{|\dot{e}_x|^2 + \alpha_x|e_y|},\ \ddot{e}_y = -\tilde{D}_y(t,X)\frac{\lceil \dot{e}_y \rfloor^2 + \alpha_y e_y}{|\dot{e}_y|^2 + \alpha_y|e_y|},$$

then $S_i = \dot{S}_i = 0$, $i = x, y$ is reached in a finite time provided that $\tilde{D}_x(t,X) > |d_{xe} + \ddot{x}_{des}|$, $\tilde{D}_y(t,X) > |d_{ye} + \ddot{y}_{des}|$. x and y dynamics are not influenced directly by the rotors, hence their stability doesn't need the introduction of an auxiliary sliding surface, like we did for the other dynamics. The respective controls for x, y positions are

$$U_x = -\frac{m\,\tilde{D}_x(t,X)}{U_z}\frac{\lceil \dot{e}_x \rfloor^2 + \alpha_x e_x}{|\dot{e}_x|^2 + \alpha_x|e_x|},\ U_y = -\frac{m\,\tilde{D}_y(t,X)}{U_z}\frac{\lceil \dot{e}_y \rfloor^2 + \alpha_y e_y}{|\dot{e}_y|^2 + \alpha_y|e_y|}.$$

4.2 Attitude control

The controls for other dynamics can be designed following similar computations as for z, so only final expressions are given for roll, pitch and yaw, respectively. The desired angles ϕ_{des}, θ_{des} are derived from the expressions for U_x and U_y (16), using the desired value of ψ:

$$\phi_{des} = \arcsin\left(U_x\sin\psi_{des} - U_y\cos\psi_{des}\right),\ \theta_{des} = \arcsin\left(\frac{U_x\cos\psi_{des} + U_y\sin\psi_{des}}{\cos\phi_{des}}\right).$$

4.2.1 Roll control

The second derivative of the position error, $e_\phi = \phi - \phi_{des}$, taking into account (15), has the form

$$\ddot{e}_\phi = qr\frac{I_{yy} - I_{zz}}{I_{xx}} + \frac{1}{I_{xx}}\left(L_\phi + d_\phi\right) - \ddot{\phi}_{des},$$

The roll control is

$$U_\phi = -\frac{b}{\delta_\phi}\tilde{D}_\phi(t,X)\frac{\lceil \dot{S}_\phi \rfloor^2 + S_\phi}{|\dot{S}_\phi|^2 + |S_\phi|},$$

where $\tilde{d}_\phi = \dot{\Delta}_\phi + \alpha_\phi(\delta_\phi L_\phi + \Delta_\phi) - (L_\phi\delta_\phi)/b - \dot{\delta}_\phi L_\phi$, $\delta_\phi = 1/I_{xx}$, $\Delta_\phi = qr(I_{yy} - I_{zz})/I_{xx} + d_\phi/I_{xx} - \ddot{\phi}_{des}$, with $\tilde{D}_\phi(t,X) > |\tilde{d}_\phi + \ddot{\phi}_{des}|$.

4.2.2 Pitch control

The second derivative of the position error, $e_\theta = \theta - \theta_{des}$, taking into account (15), has the form

$$\ddot{e}_\theta = pr\frac{I_{zz} - I_{xx}}{I_{yy}} + \frac{1}{I_{yy}}(L_\theta + d_\theta) - \ddot{\theta}_{des},$$

The pitch control is

$$U_\theta = -\frac{b}{\delta_\theta}\tilde{D}_\theta(t,X)\frac{\lceil \dot{S}_\theta \rfloor^2 + S_\theta}{|\dot{S}_\theta|^2 + |S_\theta|},$$

where $\tilde{d}_\theta = \dot{\Delta}_\theta + \alpha_\theta(\delta_\theta L_\theta + \Delta_\theta) - (L_\theta\delta_\theta)/b$, $\delta_\theta = 1/I_{yy}$, $\Delta_\theta = pr(I_{zz} - I_{xx})/I_{yy} + d_\phi/I_{yy} - \ddot{\theta}_{des}$, with $\tilde{D}_\theta(t,X) > |\tilde{d}_\theta + \ddot{\theta}_{des}|$.

4.2.3 Yaw control

Recall that for ψ dynamics, the second derivative of the position error, $e_\psi = \psi - \psi_{des}$, has the form

$$\ddot{e}_\psi = pq\frac{I_{xx} - I_{yy}}{I_{zz}} + \frac{1}{I_{zz}}\left(L_\psi + d_\psi\right) - \ddot{\psi}_{des},$$

The yaw control is

$$U_\psi = -\frac{b}{\delta_\psi}\tilde{D}_\psi(t, X)\frac{\lceil \dot{S}_\psi \rfloor^2 + S_\psi}{|\dot{S}_\psi|^2 + |S_\psi|},$$

where $\tilde{d}_\psi = \dot{\Delta}_\psi + \alpha_\psi(\delta_\psi L_\psi + \Delta_\psi) - (L_\psi \delta_\psi)/b,\ \delta_\psi = 1/I_{zz},\ \Delta_\psi = pq(I_{xx} - I_{yy})/I_{zz} + d_\psi/I_{zz} - \ddot{\psi}_{des}$, with $\tilde{D}_\psi(t, X) > |\tilde{d}_\psi + \ddot{\psi}_{des}|$.

4.3 Chattering issue

A big shortage of SMC is the chattering (high frequency oscillations of the control signal in the steady-state mode caused by discontinuity of control function and digital/measurement noises), which can ruin the rotors. In the literature the problem of chattering reduction is a well-known issue discussed in many articles, see for an example Gonzalez et al[5] . Saturation functions are a standard tool for chattering reduction in SMC that leads to a practical stability in the closed-loop system. According to Khalil[19] , if for a sign function all trajectories converge to an equilibrium, then with a saturation all trajectories converge to a compact set around that equilibrium. The proposed 2-SMC (second order sliding mode control) can also be modified to counteract the chattering avoiding the saturation functions, and using the quasi-continuous function as an approximation of the sign on the plane, with a mild modification by adding a small constant $\varrho_i > 0$ in the denominator:

$$\frac{\lceil \dot{S}_i \rfloor^2 + S_i}{\varrho_i + |\dot{S}_i|^2 + |S_i|}, \quad \frac{\lceil \dot{e}_i \rfloor^2 + \alpha_i e_i}{\varrho_i + |\dot{e}_i|^2 + \alpha_i |e_i|}.$$

where ϱ_i is strictly related with accuracy. The smaller is ϱ_i and the higher is the effort on the rotors, which result in more accentuated oscillation of the controls but with smaller convergence error, and vice versa. According to Ding et al[18] , the finite-time convergence of the system can be achieved in the ideal case, when $\varrho_i = 0$ (implies the presence of quasi-chattering) and there is no external disturbance. In our case, since these restrictions are not satisfied, the convergence is assured with respect to a compact set around the equilibrium. Then ϱ_i are tuned accordingly to a trade-off between control oscillations and convergence error.

4.4 Control implementation

As conclusion from previous control design process the quasi-continuous control is suitable to be applied to quadrotor problems with the introduction of rotors dynamics and ensuring the homogeneity property of the system. It means that the function sign can be approximated efficacely with quasi-continuous function in SMC for quadrotors, and its finite time stability is proved in Ding et al[18] , considering $\varrho_i = 0$. Design of the implemented control laws is explained extensively in Perozzi et al[20] , hence only final expression will be given below with the considered approximations.
The control for z is given by

$$U_z = \frac{m}{\cos\theta\cos\phi}\left(g + \left(\frac{\nu(X)^2 + 2\varrho(X) + \nu(X)\sqrt{\nu^2(X) + 4\varrho(X)}}{2} + \frac{L_z}{b} - L_z\alpha_z + \varpi_z\right)\frac{\lceil \dot{S}_z \rfloor^2 + S_z}{\varrho_z + |\dot{S}_z|^2 + |S_z|} - \ddot{z}_{des} + \alpha_z\dot{e}_z\right),$$

with $\varpi_z > |\dot{\Delta}_z + \dot{\delta}_z L_z|$ tuned parameter and

$$\varrho(X, \ddot{z}_{des}, \dot{z}_{des}) = \frac{1}{\sqrt{m\gamma_z}}\left(f_{ze}\left(|X|\right) + D_{ze}\right)\sqrt{|g + \alpha_z\dot{e}_z| - \ddot{z}_{des}}, \quad \nu(X) = \frac{1}{\sqrt{m\gamma_z}}\left(f_{ze}\left(|X|\right) + D_{ze}\right).$$

where γ_z is our operating point limit such that $|\cos\phi\cos\theta| \geq \gamma_z > 0$.
The controls for $i = x, y$ is given by

$$U_i = \frac{m}{U_z}\left(-\left(d_{xx} + 2d_{yy} + 2d_{zz} + \varpi_i\right)\frac{\lceil \dot{S}_i \rfloor^2 + S_i}{\varrho_i + |\dot{S}_i|^2 + |S_i|} + \ddot{i}_{des} - \alpha_i\dot{e}_i\right),$$

with $\varpi_i > 0$ tuned parameter ensuring $\tilde{D}_i(t, X) > |d_{ie} + \ddot{i}_{des}|$.

The controls for roll, pitch and yaw respectively are given by

$$
\begin{aligned}
U_\phi &= I_{xx}\left(-qr\frac{I_{yy}-I_{zz}}{I_{xx}} + \tilde{u}_\phi - \alpha_\phi \dot{e}_\phi + \ddot{\phi}_{des}\right),\\
U_\theta &= I_{yy}\left(-pr\frac{I_{zz}-I_{xx}}{I_{yy}} + \tilde{u}_\theta - \alpha_\theta \dot{e}_\theta + \ddot{\theta}_{des}\right),\\
U_\psi &= I_{zz}\left(-pq\frac{I_{xx}-I_{yy}}{I_{zz}} + \tilde{u}_\psi - \alpha_\psi \dot{e}_\psi + \ddot{\psi}_{des}\right),
\end{aligned}
$$

where the auxiliary controls are defined as

$$
\begin{aligned}
\tilde{u}_\phi &= -\frac{1}{I_{xx}}\frac{\lceil \dot{S}_\phi \rfloor^2 + S_\phi}{\varrho_\phi + |\dot{S}_\phi|^2 + |S_\phi|}\Big(\tilde{K}_\phi(f_{\phi 1}(X) + D_{\phi 1})\sqrt{|U_z|} + \bar{K}_\phi(f_{\phi 2}(X) + D_{\phi 2}) + \varpi_\phi + \alpha_\phi \delta_\phi L_\phi - \frac{\delta_\phi}{b}L_\phi\Big),\\
\tilde{u}_\theta &= -\frac{1}{I_{yy}}\frac{\lceil \dot{S}_\theta \rfloor^2 + S_\theta}{\varrho_\theta + |\dot{S}_\theta|^2 + |S_\theta|}\Big(\tilde{K}_\theta(f_{\theta 1}(X) + D_{\theta 1})\sqrt{|U_z|} + \bar{K}_\theta(f_{\theta 2}(X) + D_{\theta 2}) + \varpi_\theta + \alpha_\theta \delta_\theta L_\theta - \frac{\delta_\theta}{b}L_\theta\Big),\\
\tilde{u}_\psi &= -\frac{1}{I_{zz}}\frac{\lceil \dot{S}_\psi \rfloor^2 + S_\psi}{\varrho_\psi + |\dot{S}_\psi|^2 + |S_\psi|}\Big(\tilde{K}_\psi(f_{\psi 1}(X) + D_{\psi 1})\sqrt{|U_z|} + \bar{K}_\psi(f_{\psi 2}(X) + D_{\psi 2}) + \varpi_\psi + \alpha_\psi \delta_\psi L_\psi - \frac{\delta_\psi}{b}L_\psi\Big),
\end{aligned}
$$

with $\varpi_i > |\dot{\Delta}_i|,\ i = \phi, \theta, \psi$ tuned parameters.

4.5 Summary

The generic scheme of hierarchical control algorithm presented above is given in Fig. 2. The desired step references for x, y, z, ψ are filtered (using a third order filter) to obtain realistic smooth velocity and acceleration trajectories respecting physical constraints. Homogeneous differentiator in Perruquetti et al[21] is used to estimate the first and second derivatives of angles and the first derivative of sliding surfaces. The stability of the system can always be proved respecting the maximal value of ϕ, θ, D_x, D_y, D_z. In the proposed algorithms, the control gains are functions of the state and wind velocity, then using a wind estimator as in Perozzi et al[1] the wind velocity estimates can be substituted on-line in the control law. In this way, by adapting the control amplitude, the regulator effort on the rotors can be reduced when it is possible.

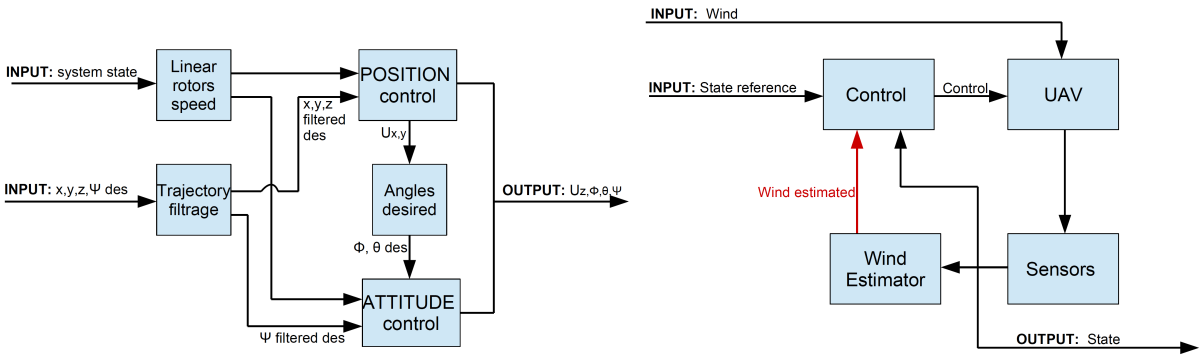

Figure 2: Control scheme and its coupling with a wind estimator. The red arrow indicates how varying wind estimation can be merged with the control. However, in this paper constant maximal absolute values of the wind are considered.

5. Numerical Experiments

Numeric experiments are performed in the simulator using UAV Simulink library, built with nonlinear aerodynamic coefficient, before to test the control in real experiments. It was developed using blade momentum theory in aerodynamic science and than validated through indoor experiments with the commercial Parrot Ar Drone 2.0 at the Onera Applied Aerodynamics Department. Hence, we consider the values listed in Tables 1. Experimental tests will be carried out in future for the X4-MaG drone to identify the aerodynamic coefficients and to test the control on-board. Wind perturbations are simulated as sinusoids, which is a reasonable assumption since the gust generator at Onera aerodynamics lab allows

to create a sinusoidal wind gust profile, hence the sinusoid wind effect over time is correlated to the motion of the drone through the turbolent area. For simplicity of demonstration the wind maximal velocities are selected constant.

A generic flight case is illustrated which couples together forward, lateral and vertical flights, moving the UAV in x, y, z positions. The corresponding external disturbances and wind are illustrated in Fig. 3, where wind in z is chosen small because of the small UAV mass and limited thrust. The controls together with state variables and reference states are illustrated in Figures 4, 5. The convergence of the system is affected by the introduction of ϱ_i in the denominator of quasi-continuous function and by the wind maximal values (the higher is the wind value and the smaller is the convergence error). Reducing the convergence error, increases the control oscillations, hence a good trade-off between effectiveness of the control and the effort on the rotors has to be found. Fig. 6 illustrates better this concept which is due to the approximation of the disturbance first derivative in the upper-bounds. However, thanks to the function gains built in Par. 3.2, a wind estimator can be coupled and wind velocity and its derivative estimates can be substituted on-line. In this way, by adapting the control amplitude, the regulator effort on the rotors can be reduced when it is possible, allowing to achieve better convergence. In this paper, even if the gains have to be fixed *a priori*, the constants D_x, D_y, D_z now have physical meanings. They can be easily tuned accordingly to a reasonable prevision about the environment. This novel approach makes much easier the tuning process of the control, in fact in the proposed stabilizing algorithms, the control gains are functions of the state, wind velocity, and rotors speed.

As a result, the presented simulations demonstrate the stability of the system affected by wind velocity and realistic external disturbances, respecting the limitations of the mini drone mass and thrust.

Table 1: Parrot Drone parameters

Parameter	Value	Unit
R	0.10	m
l	0.185	m
h	-0.025	m
g	9.81	m/s^2
θ_0	23.9	deg
m	0.472	Kg
I_{xx}	3.56e-3	$Kg.m^2$
I_{yy}	4.02e-3	$Kg.m^2$
I_{zz}	7.12e-3	$Kg.m^2$
ρ	1.25	Kg/m^3
σ	0.1114	
a	4.6542	
C_{D0}	2.15	
λ_{stat}	0.1056	
$C_{T\,stat}$	0.0223	
K_D	0.06	
K_z	0.09	

Table 2: Saturation parameters

Parameter	Value	Unit
ϕ_{max}, θ_{max}	40	deg
ψ_{max}	180	deg
ϕ_{min}, θ_{min}	-40	deg
ψ_{min}	-180	deg
$\dot{\phi}_{max}$, $\dot{\theta}_{max}$	30	deg $/s$
$\dot{\psi}_{max}$	10	deg $/s$
$\dot{\phi}_{min}$, $\dot{\theta}_{min}$	-30	deg $/s$
$\dot{\psi}_{min}$	-10	deg $/s$
x_{max}, y_{max}	100	m
z_{max}	-100	m
x_{min}, y_{min}, z_{min}	0	m
$\dot{x}_{max}$, $\dot{y}_{max}$	5	m/s
$\dot{z}_{max}$	-1	m/s
$\dot{x}_{min}$, $\dot{y}_{min}$	-5	m/s
$\dot{z}_{min}$	1	m/s
ω_{max}	400	rad/s
ω_{min}	200	rad/s

Table 3: Controls parameters

Parameter	Value
D_x, D_y	3
D_z	0.2
α_x, α_y, α_z	1
α_ϕ, α_θ, α_ψ	10
ϱ_x, ϱ_y, ϱ_z	1
ϱ_ϕ, ϱ_θ, ϱ_ψ	0.7
ϖ_z	1
ϖ_x, ϖ_y	3
ϖ_ϕ, ϖ_θ, ϖ_ψ	7

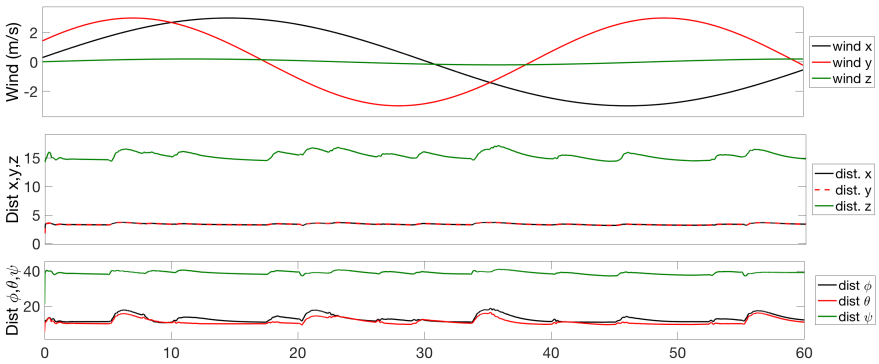

Figure 3: Wind velocities and the respective disturbances acting on x, y, z, ϕ, θ, ψ.

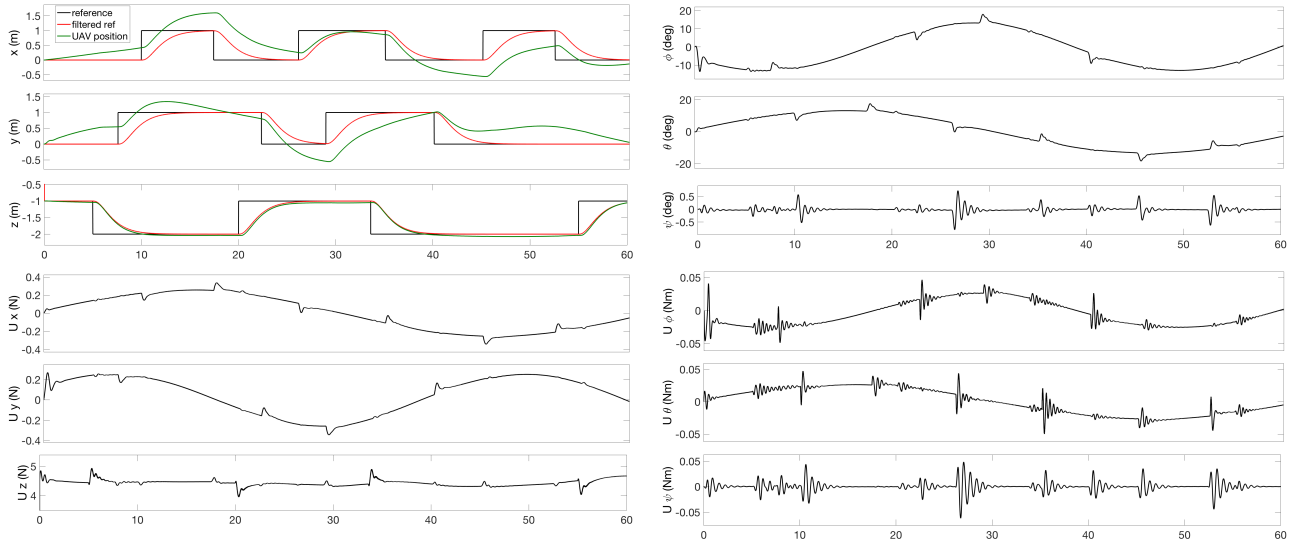

Figure 4: x, y, z positions and respective controls. ϕ, θ, ψ angles and respective controls.

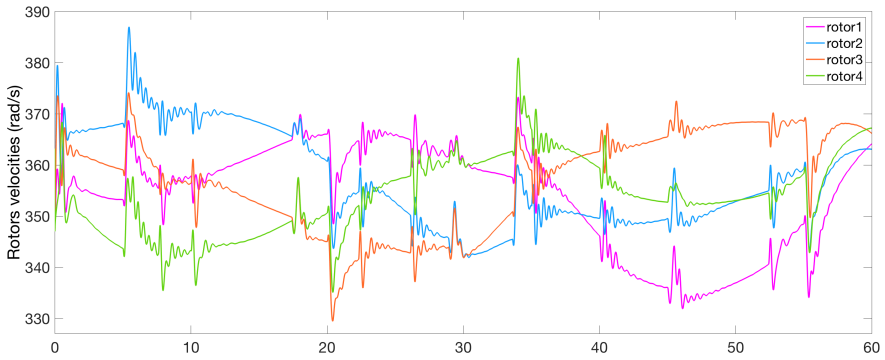

Figure 5: Rotors angular velocities.

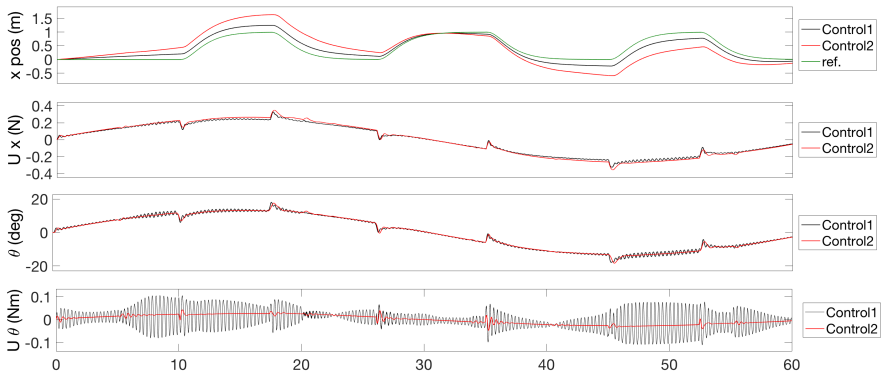

Figure 6: Effect of the constant gains in the proposed control design. In black, the response of the x, θ dynamics with $D_x = 7$, $D_y = 7$, $D_z = 1$. More robustness is obtained but with the presence of more accentuated oscillations in controls and more effort on the rotors.

6. Experimental setup

The working station for experiments is mainly composed by the Matlab/Simulink Toolbox RT-MaG (Real-Time Marseille and Grenoble toolbox) explained in Mancey et al[22] , the X4-MaG drone introduced in Manecy et al[23] , the wind gust generators at Onera-Lille, and the trajectography system at the applied aerodynamics lab inside the B20 facility, in Fig.7.

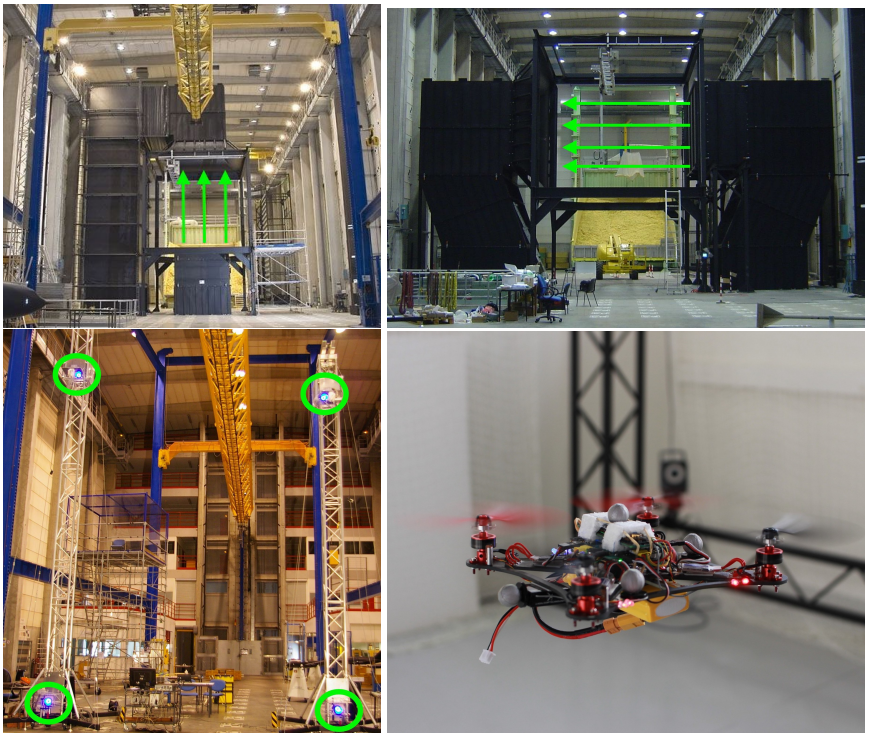

Figure 7: Vertical and lateral wind gust generators (green arrows reflect the air flow direction), trajectography system (green circles enlight the 4 cameras) inside the B20 facility, and the X4-MaG drone.

RT-MaG is an open-source toolbox, which can be used to directly design Linux-based real-time applications for Computer-On-Module (COM) using Matlab/Simulink software. Via Matlab Simulink, RT-MaG provides a high-level of abstraction user interface making it possible to design robotic applications and giving access to classical robotic communication interfaces. Since the development time is reduced, users can focus on the algorithms and the control systems. This toolbox is therefore attractive for academic and applied research purposes because it can be used to directly implement simulated controllers on embedded targets.
The drone is an open-hardware quadrotor platform called X4-MaG, developed jointly by the ISM laboratory (Marseille, France) and the Gipsa-Lab (Grenoble, France), for academic and research applications. It is a small and low-cost open quadrotor which offers two levels of controllers providing a manual mode and an automatic mode thanks to Linux-based controller embedded onboard. The robot is equipped with a low-level autopilot based on a microcontroller with its 6-axis IMU (NanoWii) and a high-level autopilot based on a Linux-based Computer-On-Module (Gumstix COM), which can be programmed directly via Simulink. The X4-MaG quadrotor is equipped with three different electronic boards: The NanoWii stabilizes the platform in the manual mode and sends sensors' values to the high-level controller in the automatic mode. A Rotor Controller Board (RCB) controls in closed-loop the rotational speed of each propeller. A Gumstix Overo AirSTORM COM is the high level controller programmed via the RT-MaG toolbox. Fig.8, taken directly from Mancey et al[23] , shows the links between the hardware components of the X4-MaG.
The B20 facility at Onera provides the wind gust generators. Gusts with sharp edges and with various amplitudes and shapes can be created. Taking advantage of the possibility of parameterizing the profile and the intensity of the gust of wind generated along the flight, as in Fig. 9, the facility is useful for modelling the real phenomenon and the representation of the spacetime nature of the effect of the turbulent wind on the aircraft. Generation of ascending or descending wind gusts is achieved by the vertical gust generator, which has the following characteristics: 3.5 m width of the airflow tube of (y axis). 4.8 m max length of the airflow tube (x axis). 5.3 m height of the airflow tube (z axis). 5 m/s maximal flow velocity. The lateral wind gusts generator has the following characteristics: 6 m width of the airflow tube (y axis). 4.8 m max length of the airflow tube (x axis). 3.5 m height of the airflow tube (z axis). 5 m/s max flow velocity.

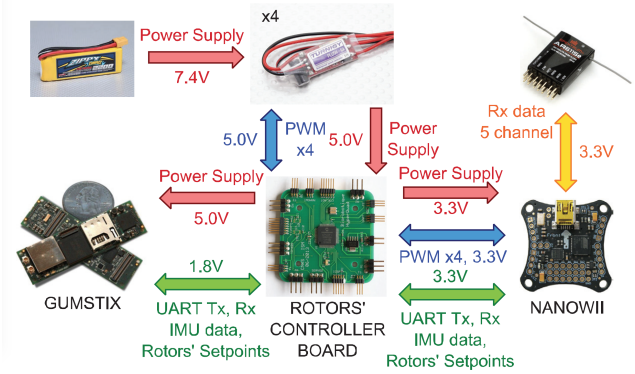

Figure 8: Hardware components of the drone.

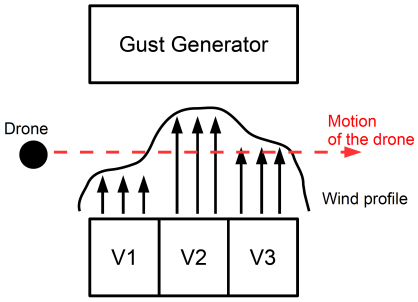

Figure 9: Lateral wind gust generator scheme. $V1$, $V2$, $V3$ are three different wind velocities that can be freely chosen accordingly to the desired wind profile.

7. Conclusion

In this work, a robust high order sliding mode control approach is introduced to stabilize a small quadrotor UAV under wind perturbations. It is shown that the disturbance bound for a UAV quadrotor at low speed depends on the control itself, the wind speed and the state. To this end, the constant gain, which is proportional to the sign functions in conventional sliding mode controls, is replaced with a time-varying function. In this way we aim to reduce as much as possible the control effort on the rotors, adapting in real-time the weighted function used in the control design. The presented results of numeric experiments confirm the effectiveness of the designed control to stabilize the drone under varying wind, and its stability is proved for a known maximal admissible wind value.
Constant maximal admissible wind values are considered as direct input to the control. However, since this approach is strictly related to the wind estimation, the coupling with the estimator, as in Fig. 2, is desired. Further works lead to a deeper study of their interaction, which must ensure the stability of the system in a compact set of the equilibrium depending by the estimation incertitude, and reduce even more the effort on the controls limiting the undesired oscillations. This coupling will ensure an optimal tuning process for the proposed control law that will be tested on the simulator. Real experiments in B20 facility and using wind gust generator will be carried out in future on the X4-MaG drone, and then the experimental results will be compared with simulations.

Appendix

According to Bernuau et al[4] , HOSMCs extend the properties of standard SMCs to systems with higher relative degree. They also help to reduce the chattering effect. In contrast to standard chattering reduction technique based on smoothing of the discontinuous control, the higher order sliding mode approach suggests to smooth the sliding motion providing the finite-time convergence to zero for the sliding variable S together with the derivatives $\dot{S}, \ldots, S^{r-1}$, where r is the relative degree of S with respect to the control input. In our case $r = 2$.
Following Ding et al[18] consider a double integrator system:

$$\ddot{x}(t) = g(t, X(t))\, u(t) + h(t, X(t)), \tag{17}$$

where $X(t) = [x(t), \dot{x}(t)]^T \in \mathbb{R}^2$ is the state vector, $u(t) \in \mathbb{R}$ is the control input, two functions $g : \mathbb{R}^3 \to \mathbb{R}$ and $h : \mathbb{R}^3 \to \mathbb{R}$ ensure forward existence and uniqueness of the system solutions at least locally. In addition, there are two known functions $\underline{g} : \mathbb{R}^3 \to \mathbb{R}$ and $\overline{h} : \mathbb{R}^3 \to \mathbb{R}$ such that for all $X \in \mathbb{R}^2$ and $t \geq 0$

$$g(t, X) \geq \underline{g}(t, X) > 0, \ |h(t, X)| \leq \overline{h}(t, X). \tag{18}$$

The following control can be proposed for (17) (a more generic case is studied in Ding et al[18]):

$$u(t, X) = -\frac{\overline{h}(t, X) + \alpha}{\underline{g}(t, X)} \frac{\left\lceil \dot{S} \right\rfloor^2 + \beta S}{|\dot{S}|^2 + \beta |S|}, \tag{19}$$

where $\alpha > 0$ and $\beta > 0$ are tuning parameters. In our case $\beta = 1$ and $\underline{g}$, $\overline{h}$ are respectively the computed lower and the upper-bounds. Note that the control (19) is continuous everywhere outside of the origin.

Theorem 1 *Consider the system in (17) and assume that the restrictions (18) are satisfied, then there exist $\alpha > 0$ sufficiently big and $\beta > 0$ such that the control (19) makes the system globally finite-time convergent.*

Aknowledgement

This work was possible thanks to the funds provided by Hauts-de-France region and ONERA: The French Aerospace Lab.

References

[1] G. Perozzi, D. Efimov, J.-M. Biannic, L. Planckaert, and P. Coton, "Wind estimation algorithm for UAV based on aerodynamic coefficients," Submitted.

[2] N. Sydney, B. Smyth, and D. A. Paley, "Dynamic control of autonomous quadrotor flight in an estimated wind field," in *52nd IEEE Conference on Decision and Control*, pp. 3609–3616, Dec. 2013. DOI: 10.1109/CDC.2013.6760438.

[3] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 153–158, Oct. 2007. DOI: 10.1109/IROS.2007.4399042.

[4] E. Bernuau, D. Efimov, W. Perruquetti, and A. Polyakov, "On homogeneity and its application in sliding mode control," *Journal of the Franklin Institute*, vol. 351, pp. 1866–1901, Apr. 2014. DOI: 10.1016/j.jfranklin.2014.01.007.

[5] I. González, S. Salazar, and R. Lozano, "Chattering-Free Sliding Mode Altitude Control for a Quad-Rotor Aircraft: Real-Time Application," *Journal of Intelligent & Robotic Systems*, vol. 73, pp. 137–155, Jan. 2014. DOI: 10.1007/s10846-013-9913-8.

[6] T. Madani and A. Benallegue, "Backstepping Sliding Mode Control Applied to a Miniature Quadrotor Flying Robot," in *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, pp. 700–705, Nov. 2006. DOI: 10.1109/IECON.2006.347236.

[7] S. Zeghlache, D. Saigaa, K. Kara, A. Harrag, and A. Bouguerra, "Backstepping sliding mode controller improved with fuzzy logic: Application to the quadrotor helicopter," *Archives of Control Sciences*, vol. 22, no. 3, pp. 315–342, 2012. DOI: 10.2478/v10170-011-0027-x.

[8] L. Derafa, A. Benallegue, and L. Fridman, "Super twisting control algorithm for the attitude tracking of a four rotors UAV," *Journal of the Franklin Institute*, vol. 349, pp. 685–699, Mar. 2012. DOI: 10.1016/j.jfranklin.2011.10.011.

[9] S. Rajappa, C. Masone, H. H. Bülthoff, and P. Stegagno, "Adaptive Super Twisting Controller for a quadrotor UAV," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2971–2977, May 2016. DOI: 10.1109/ICRA.2016.7487462.

[10] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, "Second order sliding mode control for a quadrotor UAV," *ISA Transactions*, vol. 53, pp. 1350–1356, July 2014. DOI: 10.1016/j.isatra.2014.03.010.

[11] A. Benallegue, A. Mokhtari, and L. Fridman, "High-order sliding-mode observer for a quadrotor UAV," *International Journal of Robust and Nonlinear Control*, vol. 18, pp. 427–440, Mar. 2008. DOI: 10.1002/rnc.1225.

[12] Y. Zhang and A. Chamseddine, "Fault Tolerant Flight Control Techniques with Application to a Quadrotor UAV Testbed," in *Automatic Flight Control Systems - Latest Developments*, Dr. Thomas Lombaerts (Ed.), 2012. ISBN: 978-953-307-816-8. InTech. Available from: http://www.intechopen.com/books/automatic-flight-control-systems-latest-developments/fault-tolerant-flight-control-techniques-with-application-to-a-quadrotor-uav-testbed.

[13] H. Bouadi, S. S. Cunha, A. Drouin, and F. Mora-Camino, "Adaptive sliding mode control for quadrotor attitude stabilization and altitude tracking," in *2011 IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI)*, pp. 449–455, Nov. 2011. DOI: 10.1109/CINTI.2011.6108547.

[14] L. Planckaert and P. Coton, "Quadrotor UAV aerodynamic model identification using indoor flight experiment and feasibility of UAV as wind gust sensor," in *International Micro Air Vehicles Conference and Flight Competition IMAV 2015*, (Aachen, Germany), Sept. 2015. Available from: https://hal.inria.fr/hal-01564692.

[15] W. Johnson, *Helicopter Theory*. Courier Corporation, Mar. 2012. ch 2-5,9-13,15.

[16] A. R. S. Bramwell, D. Balmford, and G. Done, *Bramwell's Helicopter Dynamics, second edition*. Butterworth-Heinemann, Apr. 2001. ch 2-7.

[17] G. J. Leishman, *Principles of Helicopter Aerodynamics with CD Extra*. Cambridge University Press, Apr. 2006. ch 2-5.

[18] S. Ding, A. Levant, and S. Li, "Simple homogeneous sliding-mode controller," *Automatica*, vol. 67, pp. 22–32, May 2016. DOI: 10.1016/j.automatica.2016.01.017.

[19] H. K. Khalil, *Nonlinear Systems*. Prentice Hall, 2002.

[20] G. Perozzi, D. Efimov, J.-M. Biannic, L. Planckaert, and P. Coton, "On sliding mode control design for UAV using realistic aerodynamic coefficients," in *56th IEEE Conference on Decision and Control*, (Melbourne, Australia), Dec. 2017.

[21] W. Perruquetti, T. Floquet, and E. Moulay, "Finite-Time Observers: Application to Secure Communication," *IEEE Transactions on Automatic Control*, vol. 53, pp. 356–360, Feb. 2008. DOI: 10.1109/TAC.2007.914264.

[22] A. Manecy, N. Marchand, and S. Viollet, "RT-MaG: An open-source SIMULINK toolbox for Linux-based real-time robotic applications," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pp. 173–180, Dec. 2014. DOI: 10.1109/ROBIO.2014.7090326.

[23] A. Manecy, N. Marchand, F. Ruffier, and S. Viollet, "X4-MaG: A Low-Cost Open-Source Micro-Quadrotor and its Linux-Based Controller," *International Journal of Micro Air Vehicles*, vol. 7, pp. 89–109, June 2015. DOI: 10.1260/1756-8293.7.2.89.