An Underactuated Control System Design for Adaptive Autopilot of Fixed-Wing Drones

Baldi, Simone; Roy, Spandan; Yang, Kang; Liu, Di

Published in:
IEEE/ASME Transactions on Mechatronics

DOI:
10.1109/TMECH.2022.3144459

Publication date:
2022

Citation for published version (APA):
Baldi, S., Roy, S., Yang, K., & Liu, D. (2022). An Underactuated Control System Design for Adaptive Autopilot of Fixed-Wing Drones. *IEEE/ASME Transactions on Mechatronics*, *27*(5), 4045 - 4056. https://doi.org/10.1109/TMECH.2022.3144459

An Underactuated Control System Design for Adaptive Autopilot of Fixed-Wing Drones

Simone Baldi, *Senior Member, IEEE*, Spandan Roy, Kang Yang, and Di Liu, *Member, IEEE*

***Abstract*—Effective design of autopilots for fixed-wing unmanned aerial vehicles (UAVs) is still a great challenge, due to unmodeled effects and uncertainties that these vehicles exhibit during flight. Unmodeled effects and uncertainties comprise longitudinal/lateral cross-couplings, as well as poor knowledge of equilibrium points (trimming points) of the UAV dynamics. The main contribution of this article is a new adaptive autopilot design, based on uncertain Euler–Lagrange dynamics of the UAV and where the control can explicitly take into account under-actuation in the dynamics, reduced structural knowledge of cross-couplings and trimming points. This system uncertainty is handled via appropriately designed adaptive laws: stability of the controlled UAV is analyzed. Hardware-in-the-loop tests, comparisons with an Ardupilot autopilot and with a robustified autopilot validate the effectiveness of the control design, even in the presence of strong saturation of the UAV actuators.**

***Index Terms*—Adaptive control, autopilot, fixed-wing unmanned aerial vehical (UAV).**

I. Introduction

FIXED-WING unmanned aerial vehicles (UAVs) are becoming more and more popular as the ease of implementation of embedded control platforms increases [1], [2]. However, effective control of fixed-wing UAVs is still a great challenge, due to unmodeled effects or uncertainties in their dynamics [3], [4]. Control of fixed-wing UAVs is typically organized according to a low-level layer (*autopilot*), and a high-level layer (*guidance*) [5]. This two-layered architecture makes it possible to simplify Euler–Lagrange equations of motion into unicycle-type models used for guidance [5]–[7]. Then, the guidance inputs (e.g., course angles and velocities) can be passed to the autopilot, which uses measured/observed UAV states (position, velocity, attitude, gyro, etc. [8], [9]) to control the flying surfaces (aileron, rudder, and elevator) and the thrust. Off-the-shelf autopilots (Ardupilot, PX4, NAVIO2, etc.) have become widely available: since many of these autopilots have open-source code, we know that their architectures rely on cascaded proportional-integral-derivative (PID) loops for roll, pitch, and yaw dynamics (cf. the documentation [10], [11]).

The autopilot layer should compensate for uncertain aerodynamics unaccounted in guidance models, but this is challenging in general [12]–[15]. For example, when roll, pitch, and yaw dynamics are linearized and simplified for cascaded loops, cross-couplings are neglected and the knowledge of equilibrium points of the Euler–Lagrange equations (trimming points) is required [5]. This means that underactuation effects and longitudinal/lateral cross-couplings are not taken into account in an appropriate way. As a result, the ideal performance of any intelligent guidance law developed to adapt to uncertain winds [16]–[18], uncertain course dynamics [19], [20], or uncertain interactions with neighbors [21]–[25], may be degraded if the autopilot layer is not able to deal with uncertainty. Studies have shown that the autopilot layer ultimately affects the performance of the UAV, and can lead to degraded tracking and increased power consumption [26], [27].

Beyond off-the-shelf autopilots, linearization of the UAV dynamics around one or multiple trimming points [28], [29] has been adopted for multivariable robust [30], [31] or optimal [32], [33] control. Alternatively, nonlinear (e.g., Euler–Lagrange) UAV dynamics have been used for reference generation, feedforward control [34]–[36], or feedback linearization. The main advantage of the aforementioned architectures is that the design takes into account the underactuation of the UAV dynamics. However, this is done at the price of requiring accurate parametric and structural knowledge of the UAV dynamics. To relax the requirement of an accurate model of the UAV, sliding-mode control [37] or ad-hoc robust cascaded design [38], [39] have been studied to robustify the feedback linearization design. Robustness in sliding-mode control of similar designs is ensured by fixed gains designed based on a worst-case uncertainty bound:

Manuscript received 17 April 2021; revised 12 July 2021, 22 September 2021, and 23 November 2021; accepted 16 January 2022. Date of publication 3 February 2022; date of current version 17 October 2022. Recommended by Technical Editor S. G. Loizou and Senior Editor M. BASIN. This work was supported in part by the Natural Science Foundation of China under Grant 62073074, in part by the Double Innovation Plan under Grant 4207012004, in part by the Special Funding for Overseas under Grant 6207011901, and in part by the Drone-based Aerial Manipulation with Human-in-the-loop, IHFC, India. *(Corresponding authors: Simone Baldi; Di Liu.)*

Simone Baldi is with the School of Cyber Science and Engineering, Southeast University, 210096 Nanjing, China, and also with the Delft Center for Systems and Control, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: s.baldi@tudelft.nl).

Spandan Roy is with the Robotics Research Center, International Institute of Information Technology Hyderabad, Hyderabad 500032, India (e-mail: spandan.roy@iiit.ac.in).

Kang Yang is with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: 220194629@seu.edu.cn).

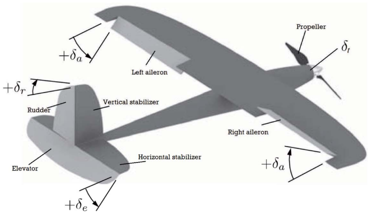
Di Liu is with the Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, 9712 Groningen, The Netherlands, and also with the School of Cyber Science and Engineering, Southeast University, 210096 Nanjing, China (e-mail: di.liu@rug.nl).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2022.3144459.

Digital Object Identifier 10.1109/TMECH.2022.3144459

therefore, these approaches require *a priori* bounded uncertainty with known bounds, implying that some parametric a structural knowledge is still crucial in the autopilot design. This overview shows that there is no definite answer to the design of autopilots with significant uncertainty in the UAV dynamics. This significant uncertainty may require to estimate the unmodeled terms online by making use of adaptive gains.

TABLE I
FIXED-WING UAV CONTROLS, AND STATE VARIABLES IN EQUATIONS OF MOTION



x	north position, inertial
y	east position, inertial
z	down position, inertial
u	surge, body frame
v	sway, body frame
w	heave, body frame
ϕ	Roll Euler angle
θ	Pitch Euler angle
ψ	Yaw Euler angle
p	Roll rate
q	Pitch rate
r	Yaw rate

The main contribution of this article is a new adaptive autopilot framework based on uncertain Euler–Lagrange dynamics, where the resulting control can explicitly take into account under-actuation in the dynamics, as well as uncertainty via adaptive laws. This article significantly improves our preliminary study [40], which assumes full-actuation. On the other hand, the framework proposed in this work exploits a realistic description of the underactuated fixed-wing UAV dynamics, with cross-couplings and without linearization around trimming points. Hardware-in-the-loop tests, comparisons with an Ardupilot-based autopilot and with a robustified autopilot validate the effectiveness of the control design, even in the presence of strong saturation of the UAV actuators.

The rest of this article is organized as follows: Section II recalls the equations of motions of fixed-wing UAVs; Section III presents the autopilot design with stability analysis in Appendix; tests in Section IV validate the approach.

II. FIXED-WING UAV MODELING

A fixed-wing UAV can be modeled using a 6 degrees-of-freedom rigid body equations of motions. Let us start by introducing some coordinate frames of interest:

1) The Vehicle Frame $\mathscr{F}^v$: The inertial frame whose unit vectors $\mathbf{i}^i$, $\mathbf{j}^i$, and $\mathbf{k}^i$ are directed north, east, and down, respectively (NED convention), and translated onto the center of mass of the vehicle.

2) The Body Frame $\mathscr{F}^b$: Defined via the Euler angles ψ, θ and ϕ (yaw, pitch, and roll) about the axis $\mathbf{k}^i$, $\mathbf{j}^{v_1}$, and $\mathbf{i}^{v_2}$ ($\mathscr{F}^{v1}$ and $\mathscr{F}^{v2}$ are the two intermediate vehicle frames when Euler angle rotations occurs from vehicle to body frame). The rotation matrix from the vehicle to the body frame is

$$\mathscr{R}_v^b(\phi,\theta,\psi) = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\psi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & s_\phi c_\theta \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\psi s_\theta s_\psi - s_\psi c_\psi & c_\phi c_\theta \end{bmatrix}$$

where the short notations $c_\phi = \cos\phi$ and $s_\phi = \sin\phi$ (and similar for θ, ψ) have been used.

Remark 1 (Role of frames): Forces and moments acting on the UAV are in the body frame $\mathscr{F}^b$. Other important frames are the stability and wind frames $\mathscr{F}^s$, $\mathscr{F}^w$ [5], not presented here for lack of space.

A. Equations of Motion

Twelve state variables, summarized in Table I, are introduced to derive the equations of motion. The resulting 6 degrees-of-freedom nonlinear UAV model is

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = (\mathscr{R}_v^b(\phi,\theta,\psi))^{-1} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \frac{1}{m} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_\phi s_\theta / c_\theta & c_\phi s_\theta / c_\theta \\ 0 & c_\phi & s_\phi \\ 0 & s_\phi / c_\theta & c_\psi / c_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \Gamma_1 pq - \Gamma_2 qr \\ \Gamma_5 pr - \Gamma_6 (p^2 - r^2) \\ \Gamma_7 pq - \Gamma_1 qr \end{bmatrix} + \begin{bmatrix} \Gamma_3 \mathscr{L} + \Gamma_4 \mathscr{N} \\ \frac{1}{J_y} \mathscr{M} \\ \Gamma_4 \mathscr{L} + \Gamma_8 \mathscr{N} \end{bmatrix} \tag{1}$$

where m is the mass of the UAV; f_x, f_y, and f_z are the forces acting on $\mathbf{i}^b$, $\mathbf{j}^b$, and $\mathbf{k}^b$, respectively; $\mathscr{L}$, $\mathscr{M}$, $\mathscr{N}$ are the rolling, pitching, and yawing moments, and

$$\Gamma_1 = \frac{J_{xz}(J_x - J_y + J_z)}{J_x J_z - J_{xz}^2}, \quad \Gamma_2 = \frac{J_z(J_z - J_y) + J_{xz}^2}{J_x J_z - J_{xz}^2}$$

$$\Gamma_3 = \frac{J_z}{J_x J_z - J_{xz}^2}, \quad \Gamma_4 = \frac{J_{xz}}{J_x J_z - J_{xz}^2},$$

$$\Gamma_5 = \frac{J_z - J_x}{J_y}$$

$$\Gamma_7 = \frac{(J_x - J_y)J_x + J_{xy}^2}{J_x J_z - J_{xz}^2}, \quad \Gamma_8 = \frac{J_x}{J_x J_z - J_{xz}^2}, \quad \Gamma_6 = \frac{J_{xy}}{J_y}$$

where the J-terms are components of the inertia tensor J.

B. External Forces and Moments

The external forces and moments, as expressed in body frame components, are

$$\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \mathscr{R}_v^b(\phi,\theta,\psi) \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + F_a + F_p, \quad \begin{bmatrix} \mathscr{L} \\ \mathscr{M} \\ \mathscr{N} \end{bmatrix} = M_a$$

where the first term in the force is the gravitational force, while the subscripts a and p stand for the aerodynamic and propulsion contribution.

1) Propulsion Force: A simple propulsion model for UAVs uses the Bernoulli principle to compute the thrust

$$F_p = C_{\text{prop}} \begin{bmatrix} S_p(P_{\text{out}} - P_{\text{in}}) \\ 0 \\ 0 \end{bmatrix} \tag{2}$$

where C_{prop} is the rotor thrust efficiency, S_p is the area swept out by the propeller, and P_{in} and P_{out} are the pressures before and after the propeller, expressed as

$$P_{\text{in}} = P_0 + \frac{1}{2}\rho V_a^2, \quad P_{\text{out}} = P_0 + \frac{1}{2}\rho V_{\text{out}}^2 \tag{3}$$

where V_{out} is the air velocity at the exit of the propeller and P_0 the atmospheric pressure. In [41] and [42], it is shown that V_{out} can be approximated as $R(k_{\text{mot}}\delta_t + q_{\text{mot}})$, where R is the radius of the propeller, and the other term is its angular velocity, approximated as a linear function of the throttle δ_t (in %).

2) Aerodynamic Forces and Moments: Consider the UAV control surfaces δ_a (aileron deflection), δ_e (elevator deflection), and δ_r (rudder deflection). The lift force, the drag force, and the pitch moment can be expressed as

$$F_{\text{lift}} = \frac{1}{2}\rho V_a^2 S_{\text{wing}} C_L(\alpha, q, \delta_e) \tag{4}$$

$$F_{\text{drag}} = \frac{1}{2}\rho V_a^2 S_{\text{wing}} C_D(\alpha, q, \delta_e) \tag{5}$$

$$\mathscr{M} = \frac{1}{2}\rho V_a^2 S_{\text{wing}} c_{\text{wing}} C_m(\alpha, q, \delta_e) \tag{6}$$

where ρ is the air density, S_{wing} the wing planform area, c_{wing} is the wing main chord, C_L, C_D, and C_m are nondimensional coefficients depending on angle of attack α, pitch rate q, and elevator deflection δ_e. As F_{lift} and F_{drag} are expressed in the stability frame, they have to the rotated onto the body frame

$$\begin{bmatrix} f_x \\ 0 \\ f_z \end{bmatrix} = \mathscr{R}_w^b(\alpha, \beta) \begin{bmatrix} -F_{\text{drag}} \\ 0 \\ -F_{\text{lift}} \end{bmatrix} \tag{7}$$

where β the side-slip angle, and the negative signs result from the NED convention.

The lateral aerodynamics depend on the lift force, the roll and yaw moments, expressed as

$$f_y = \frac{1}{2}\rho V_a^2 S_{\text{wing}} C_Y(\beta, p, r, \delta_a, \delta_r) \tag{8}$$

$$\mathscr{L} = \frac{1}{2}\rho V_a^2 S_{\text{wing}} b_{\text{wing}} C_l(\beta, p, r, \delta_a, \delta_r) \tag{9}$$

$$\mathscr{N} = \frac{1}{2}\rho V_a^2 S_{\text{wing}} b_{\text{wing}} C_n(\beta, p, r, \delta_a, \delta_r) \tag{10}$$

where b_{wing} is the wingspan, and C_Y, C_l, and C_n are nondimensional coefficients depending on side-slip angle β, roll rate p, yaw rate r, aileron deflection δ_a, and rudder deflection δ_r. Estimates of C_Y, C_l, C_n, C_L, C_D, and C_m can obtained for a fixed-wing UAV by means of USAF DATCOM [42], [43] or similar software, after inputting all the geometric characteristics of the UAV.

The UAV model described in this section is in line with the literature (cf. [5]). In the next section, such model is manipulated for autopilot design, retaining its main dynamical features, and addressing unmodeled dynamics.

III. Autopilot Design

Based on the system dynamics (1), the UAV position and attitude dynamics can be written as

$$\begin{bmatrix} mI_3 & 0 \\ 0 & \mathscr{R}_j(\chi_2)J \end{bmatrix} \begin{bmatrix} \ddot{\chi}_1 \\ \ddot{\chi}_2 \end{bmatrix} + \begin{bmatrix} G + d_1 \\ C_2(\chi_2, \dot{\chi}_2)\dot{\chi}_2 + d_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{11}$$

$$\tau_1 = \mathscr{R}_v^b(\chi_2)\bar{\tau}_1 \tag{12}$$

where $m \in \mathbb{R}^+$ and $J \in \mathbb{R}^{3\times3}$ are mass and inertia matrix, $\mathscr{R}_j \in \mathbb{R}^{3\times3}$ is the transformation in the second equation of (1); $\chi_1 \triangleq \begin{bmatrix} x & y & z \end{bmatrix}^T \in \mathbb{R}^3$; $\chi_2 \triangleq \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T \in \mathbb{R}^3$; $G \triangleq \begin{bmatrix} 0 & 0 & mg \end{bmatrix}^T \in \mathbb{R}^3$; $C_2(\chi_2, \dot{\chi}_2) \in \mathbb{R}^{3\times3}$ is the Coriolis matrix; d_1 and d_2 denote disturbances in position and attitude; $\tau_2 \triangleq \begin{bmatrix} \mathscr{L} & \mathscr{M} & \mathscr{N} \end{bmatrix}^T \in \mathbb{R}^3, \tau_1 \in \mathbb{R}^3$ are forces in inertial frame, and $\bar{\tau}_1 \triangleq \begin{bmatrix} f_x & f_y & f_z \end{bmatrix}^T \in \mathbb{R}^3$ the forces in body frame.

To the purpose of control design, we further introduce the following control allocation transformation:

$$\begin{bmatrix} f_x \\ \mathscr{M} \\ f_z \\ f_y \\ \mathscr{L} \\ \mathscr{N} \end{bmatrix} = \underbrace{\begin{bmatrix} C_{D_t} & -C_{D_e}c_\alpha + C_{L_e}s_\alpha & 0 & 0 \\ C_{m_t} & C_{m_e} & 0 & 0 \\ C_{L_t} & -C_{D_e}s_\alpha - C_{L_e}c_\alpha & 0 & 0 \\ 0 & 0 & C_{Y_r} & C_{Y_a} \\ 0 & 0 & C_{l_r} & C_{l_a} \\ 0 & 0 & C_{n_r} & C_{n_a} \end{bmatrix}}_{B_\chi} \underbrace{\begin{bmatrix} \delta_t \\ \delta_e \\ \delta_r \\ \delta_a \end{bmatrix}}_{\delta} \tag{13}$$

where $c_\alpha = \cos\alpha$, $s_\alpha = \sin\alpha$, the terms C_{D_t}, C_{m_t}, C_{L_t} represent the linear contribution of the throttle to forces and moments, C_{D_e}, C_{m_e}, C_{L_e} represent the linear contribution of the elevator to forces and moments, C_{Y_r}, C_{l_r}, C_{n_r} represent the linear contribution of the rudder to forces and moments, and C_{Y_a}, C_{l_a}, C_{n_a} represent the linear contribution of the aileron to forces and moments. These terms come from Taylor expansion of the nonlinear functions in (6) and (10).

The following assumption highlights the *lack of knowledge* of various dynamic terms for control design:

Assumption 1: Define $\bar{\chi} \triangleq [\chi_1{}^T\, \chi_2{}^T\, \dot{\chi}_1^T\, \dot{\chi}_2^T]^T$. The system terms C_2, d_1, d_2 can be bounded as $\|C_2\| \le \bar{c}\|\dot{\chi}_2\|$, $\|d_1\| \le \bar{d}_0 + \bar{d}_1\|\bar{\chi}\| + \bar{d}_2\|\bar{\chi}\|^2$, $\|d_2\| \le \check{d}_0 + \check{d}_1\|\bar{\chi}\| + \check{d}_2\|\bar{\chi}\|^2$. All coefficients in the bounds are unknown for control design. Nominal values of m, J and of the terms in B_χ are available.

Remark 2 (Uncertainties): The bound $\|C_2\| \le \bar{c}\|\dot{\chi}_2\|$ is a standard property of Euler–Lagrange dynamics (cf. [44]). In UAVs, this bound holds since $C_2(\chi_2, \dot{\chi}_2)\dot{\chi}_2$ in (11) is quadratic in the angular velocities and bounded with respect to Euler angles. Nominal payload and geometry provide nominal values of mass and inertia, whereas software tools such as USAF DATCOM [43] provide nominal linearized values of B_χ in (13). Considering only the linear terms of the aerodynamics creates unmodeled terms, which can be addressed by the state-dependent form of disturbances d_1 and d_2. Uncertainty in B_χ is addressed later in the control design (cf. Assumption 2). Considering Assumption 1 with an uncertain B_χ requires a new formulation of the error dynamics, unavailable even in recent literature on underactuated Euler–Lagrange systems where B_χ is typically the identity matrix [45], [46].

Control Problem: Design an adaptive autopilot for (11)–(13) to track desired position and attitude trajectories, in the presence of uncertain system dynamics terms.

A. Under-Actuated Decomposition

To design the autopilot, we decompose the states as

$$\chi_a = \begin{pmatrix} x & \phi & y & \theta \end{pmatrix}^T, \quad \chi_u = \begin{pmatrix} z & \psi \end{pmatrix}^T \tag{14}$$

where the subscript a stands for actuated, while u stands for unactuated. Accordingly, the dynamics can be written as follows (dependency on χ_2 will be omitted for compactness):

$$\begin{bmatrix} M_{aa} & M_{au} \\ M_{ua} & M_{uu} \end{bmatrix} \begin{bmatrix} \ddot{\chi}_a \\ \ddot{\chi}_u \end{bmatrix} + \begin{bmatrix} h_a \\ h_u \end{bmatrix} = \begin{bmatrix} B_a \\ B_u \end{bmatrix} \delta \tag{15}$$

where the terms M_{aa}, M_{au}, M_{uu} are the rearranged mass/inertia terms in (11) and (12), h_a, h_u, are the rearranged gravity/Coriolis/disturbance terms in (11) and (12), and B_a, B_u are the rearranged control allocation terms in (13).

Dynamics (15) can be rearranged as

$$\bar{M}\ddot{\chi}_a = -\bar{h}_a + \bar{B}_a\delta \tag{16a}$$

$$M_{uu}\ddot{\chi}_u = -\bar{h}_u + \bar{B}_u\delta \tag{16b}$$

where

$$\begin{aligned} &\bar{M} \triangleq M_{aa} - M_{au}M_{uu}^{-1}M_{ua} && \bar{B}_a \triangleq B_a + M_{au}M_{uu}^{-1}B_u \\ &\bar{h}_a \triangleq h_a + M_{au}M_{uu}^{-1}h_u && \bar{B}_u \triangleq B_u + M_{ua}\bar{M}^{-1}\bar{B}_a \\ &\bar{h}_u \triangleq h_u - M_{ua}\bar{M}^{-1}\bar{h}_a. \end{aligned} \tag{17}$$

Let $\chi^d \triangleq [{\chi_a^d}^T \ {\chi_u^d}^T]^T$ be the desired trajectory and let $e_a \triangleq \chi_a - \chi_a^d$, $e_u \triangleq \chi_u - \chi_u^d$ be the tracking error in actuated and nonactuated states. Define an auxiliary error

$$\eta \triangleq \dot{e}_a + \Phi_a e_a + \Upsilon_u \dot{e}_u + \Phi_u e_u \tag{18}$$

with user-defined matrices $\Phi_a > 0$ and $\Upsilon_u, \Phi_u \in \mathbb{R}^{4\times 2}$ with (full) rank 2.

Remark 3 (Desired trajectories): A nontrivial problem for under-actuated systems is how to define a feasible desired trajectory. This problem is well known for UAVs and standard solutions exist. In this work, we adopt the approach of [5, Chap. 9], which gives geometric conditions for desired positions and angles along primitive paths (straight lines and orbits): more complex paths can be defined as the combination of such primitive paths.

Define $\xi \triangleq [{e_a}^T \ {e_u}^T \ \dot{e}_a^T \ \dot{e}_u^T]^T$ and $x \triangleq [{e_u}^T \ \dot{e}_u^T]^T$. The control law is designed as

$$\delta = \hat{M}_s^{-1}(-\Lambda\eta - \Phi_a\dot{e}_a - \Phi_u\dot{e}_u - \bar{\delta})$$

$$\bar{\delta} = \begin{cases} \mu\frac{\eta}{||\eta||} & \text{if } ||\eta|| \geq \epsilon \\ \mu\frac{\eta}{\epsilon} & \text{if } ||\eta|| < \epsilon \end{cases} \tag{19}$$

where $\Lambda \in \mathbb{R}^{4\times 4}$ satisfies $\Lambda > 0$, $\epsilon > 0$ defines the saturation region, $\bar{\delta}$ tackles uncertainties via μ, which is designed as

$$\mu = \frac{1}{1-E}(\hat{\vartheta}_0 + \hat{\vartheta}_1||\xi|| + \hat{\vartheta}_2||\xi||^2 + \gamma) \tag{20}$$

with adaptive laws ($i = 0, 1, 2$)

$$\dot{\hat{\vartheta}}_i = \kappa_i(||\eta|| + ||x||)||\xi||^i - \sigma_i\hat{\vartheta}_i\bar{\epsilon}||x||||\xi||^i \tag{21a}$$

$$\begin{aligned} \dot{\gamma} = & -\gamma\left\{\gamma_0 + \gamma_1(||\xi||^5 - ||\xi||^4) + \gamma_2(||x|| + ||\xi||)\right\} \\ & + \gamma_0(||\eta|| + ||x||) + \gamma_0\nu \end{aligned} \tag{21b}$$

$$\text{initial conditions } \hat{\vartheta}_i(0) > 0,\ \gamma(0) > \nu \tag{21c}$$

and $\kappa_i, \sigma_i, \bar{\epsilon}, \gamma_0, \gamma_1, \gamma_2, \nu \in \mathbb{R}^+$ satisfying

$$\gamma_2 \geq \gamma_1,\ \bar{\epsilon} > 1 + \frac{E_1}{1-E} \tag{21d}$$

with E_1 being a constant satisfying $||PB_m\bar{B}|| \leq E_1$, where these three matrices are defined after (27) in Appendix.

Finally, the structure of μ in (19) is motivated by the upper bounds of $||\Psi||$ and $||\bar{\Psi}||$ in (26) and (29) (see analysis in Appendix), while $\hat{M}_s$ is the nominal value of

$$M_s \triangleq \bar{M}^{-1}\bar{B}_a - \Upsilon_u M_{uu}^{-1}\bar{B}_u$$

which satisfies

Assumption 2: A scalar E is known such that

$$||M_s\hat{M}_s^{-1} - I_4|| \leq E < 1. \tag{22}$$

The uncertainty in the mass matrix in the form (22) was originally proposed for fully actuated Euler–Lagrange systems [44], and its meaning is that the estimated mass matrix $\hat{M}_s$ should not deviate too much from the actual mass matrix M_s. Therefore, the value of E can be calculated based on an uncertainty set around the nominal M_s: the larger the uncertainty set, the larger E, being $E < 1$ required for stability analysis.

Remark 4 (Meaning of adaptive laws): In (21), $\hat{\vartheta}_i$ is the estimate of $\bar{\vartheta}_i^* \triangleq \max\{\vartheta_i^*, \vartheta_i^{**}\}$, $i = 0, 1, 2$ which are positive gains describing the state-dependent bound of the uncertainty term [cf. (26), (29) in Appendix]; γ is an auxiliary gain. This auxiliary gain is also positive, since $\gamma_2 \geq \gamma_1$ makes the term '$\gamma_0 + \gamma_1(||\xi||^5 - ||\xi||^4) + \gamma_2(||x|| + ||\xi||)$' in (21b) positive for all x, ξ. In other words, the first-order filter (21b) generating γ is a stable filter with positive exogenous input.

Remark 5 (Approximating uncertainty): The term μ in (20) represents a parametric model for the uncertainty terms $||\Psi||$ and $||\bar{\Psi}||$ in (26) and (29). This means that the uncertainty can be simply approximated by adapting three gains, which reduces the complexity of the approximation as compared to complex neural networks or fuzzy logic approximators.

B. Stability Result

Theorem 1: Under Assumptions 1–2, the trajectories of system (11)–(12) using the control laws (19), (20) and adaptive laws (21) are uniformly ultimately bounded (UUB).

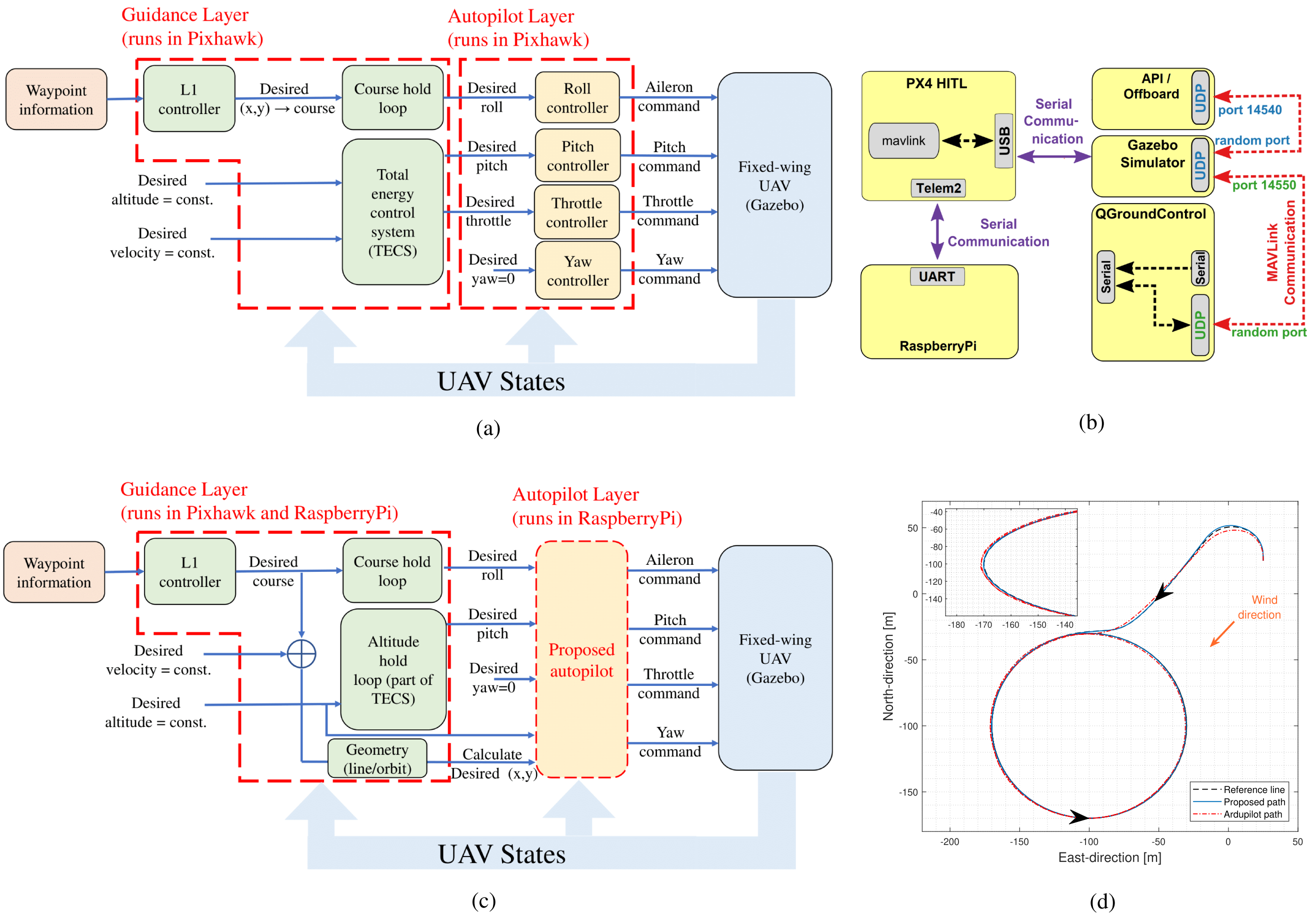


Fig. 1. Structure of the Ardupilot control suite and of the hardware-in-the-loop setup. (a) Control architecture with standard Ardupilot autopilot code. (b) Hardware interacting with Gazebo. (c) Modified architecture for the proposed autopilot. (d) x–y plane (black UAV is out of scale).

Proof: See Appendix. Tuning considerations for the control and adaptive laws are given at the end of Appendix. ■

Theoretically, error bounds can be reduced for $\sigma_i, \epsilon \to 0$ in (19)–(21), i.e., for an adaptive sliding mode controller with monotonically increasing $\hat{\vartheta}_i$. However, high gains $\hat{\vartheta}_i$ are to be avoided in practice as the control might hit saturation constraints or excite complex uncertainty. Our tests in Section IV will show practical aspects of the gains $\hat{\vartheta}_i$, by comparing the proposed approach against various $\hat{\vartheta}_i$. The tests show that increasing $\hat{\vartheta}_i$ does not monotonically improves performance: the performance can degrade for large $\hat{\vartheta}_i$. Tests with tighter saturation bounds of the UAV actuators are also presented. In the future, it is of interest to address the joint presence of state-dependent uncertainty and saturation, e.g., by extending recently proposed Lyapunov redesigns [47], [48] to an underactuated setting.

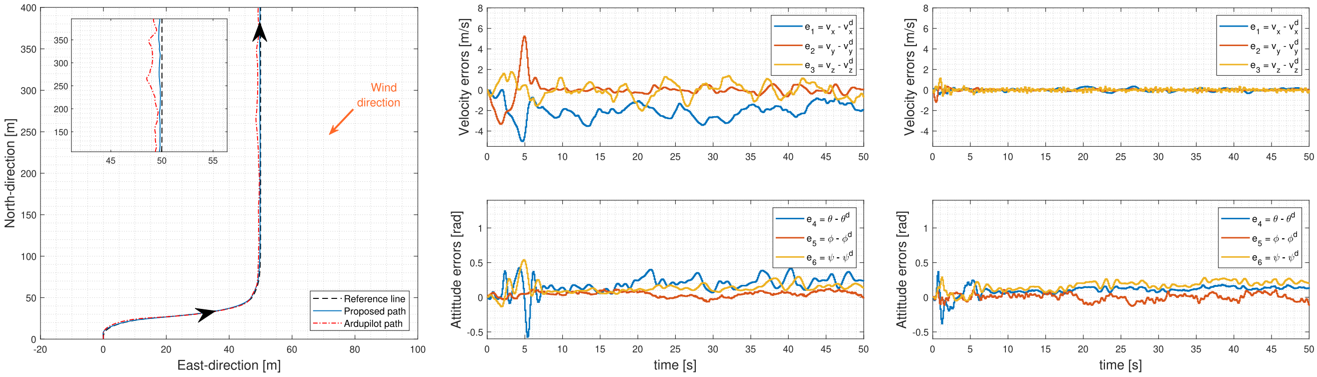
IV. Tests

The performance of the proposed controller is tested on a hardware-in-the-loop set up with Ardupilot autopilot software suite and all controllers (including the proposed one) running on the same Ardupilot-compatible hardware, i.e., a PixHawk microcontroller coupled with RaspberryPi as companion computer [cf. Fig. 1(b)]. The original Ardupilot suite, whose architecture can be seen in Fig. 1(a), is used for comparisons and the interested reader can refer to the Ardupilot documentation [10] for the control diagrams (see also the following Section IV-A). Therefore, a comparison with Ardupilot provides an interesting benchmark with an open-source software suite developed and maintained by a large UAV community. In our hardware-in-the-loop UAV platform, the Ardupilot hardware is connected to a Gazebo/ROS environment. The experiments include the inner and outer loops embedded in Ardupilot, and allow to capture realistic effects of the inner layer on the outer guidance layer, similar to real flight experiments.

A. Hardware-in-The-Loop Set up With Ardupilot

Ardupilot is organized via the aforementioned two-layered architecture: autopilot and guidance, cf. Fig. 1(a). The guidance determines the commands via an L_1 position error controller for the course angle, and a total energy control system (TECS) for simultaneous control of airspeed and altitude.

The autopilot layer in Ardupilot controls lateral (ailerons and rudder) and longitudinal (throttle and elevator) dynamics via cascaded PID loops. For the lateral dynamics, the variables of interest are the roll rate p and yaw rate r (PID inner loops), and the roll angle ϕ and yaw angle ψ (PID outer loops). The control surfaces are the aileron deflection δ_a and the rudder deflection δ_r. The roll angle is used as a control variable for tracking the course angle coming from the L_1 position error controller.



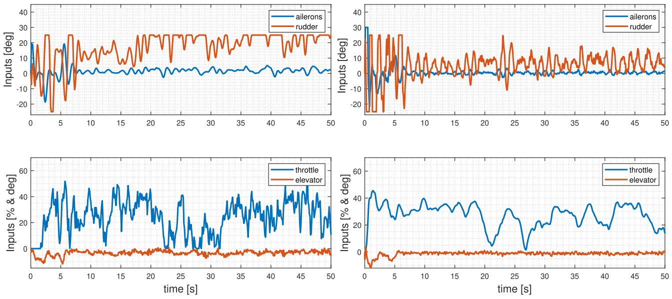
(a) (b) (c)

Fig. 2. Line path experiments: path following and tracking errors. (a) x-y plane (black UAV is out of scale). (b) Ardupilot tracking errors. (c) Proposed tracking errors.

For the longitudinal dynamics, the variables of interest are the pitch rate q (PID inner loop), and the pitch angle θ (PID outer loop). The control signals are the elevator deflection δ_e, and the throttle δ_t. The pitch angle is used as a control variable for tracking the altitude h and airspeed V_a determined by TECS. The elevator affects θ, which can be used to affect altitude and airspeed. The throttle also affects airspeed and altitude (another PID outer loop). It is not straightforward to provide all the details of Ardupilot: the interested reader is referred to the documentation and code of standard open-source autopilot suites [10], [11]. A complete academic reference is the book [5, Chap. 6] describing the successive loop closure via PID control and methods for tuning the various gains. In our experiments, the gains of the controllers in Ardupilot are the result of the Autotune procedure[1] of Ardupilot itself. In other words, Ardupilot provides an in-built routine for automatically tuning the PID gains of the control loops.

To test the proposed controller, the original Ardupilot architecture is modified accordingly as in Fig. 1(c): in particular, we make use of a RaspberryPi that can communicate with Pixhawk as in Fig. 1(b). The RasperryPi acts as a companion computer and the proposed strategy runs there. The L_1 position controller, course hold loop are the same as the original Ardupilot, and the pitch reference is also provided by the corresponding code in the Ardupilot. The main difference with the original Ardupilot is that a geometric approach is used to calculate the desired position based on the desired velocity and the desired course from the L_1 position controller.

Gazebo is used as a physical simulator of the UAV, which is a fixed-wing UAV with dynamics in 6 degrees of freedom (also refer to the template [49]). The dynamics in Gazebo essentially are analogous to the dynamics described in Section II, as it can be found in most flight control books. The UAV model generated in Gazebo is a 1.0 kg standard structure fixed-wing UAV with aileron, rudder, and elevator, and where the rotor can generate airspeed in the range [10–25]m/s. Primitive paths are used during testing, i.e., straight-line paths and orbit paths [and their combination, cf. Fig. 1(d)]. The use of such primitives is standard in UAV literature [5], [6], [50]. Gazebo runs in parallel, using MAVLink, with QGroundControl a suite for flight control and mission planning.



(a) (b)

Fig. 3. Line path experiments: inputs. (a) Ardupilot actuation. (b) Proposed actuation.

The main parameters of the UAV are: $m = 1.0$, $J_{xx} = 0.02$, $J_{yy} = 0.025$, $J_{xx} = 0.05$ (off-diagonal entries of the inertia tensor are taken as 0), $S = 0.23$, $b = 1.3$, $c = 0.18$ and

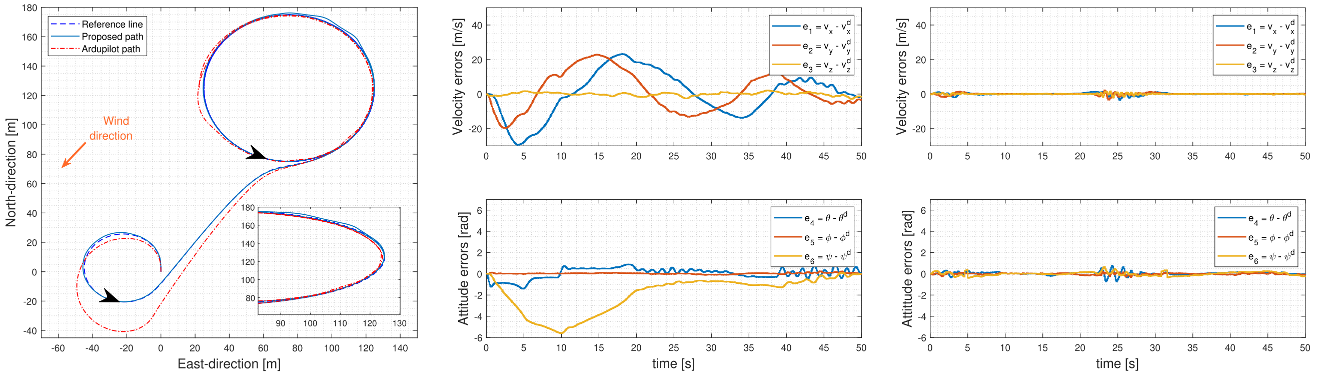
$$\hat{M}_s^{-1} = \begin{bmatrix} 4.13 \cdot 10^2 & 0 & 0 & 0 \\ 0 & 0.464 & 2.366 & 0 \\ 0 & -5.13 & 24.7 & -23.3 \\ 0 & 0 & 11.7 & 0 \end{bmatrix} \tag{23}$$

as estimated from aerodynamic coefficients through USAF DATCOM. The control parameters of the proposed autopilot are the same for all tests: $\Lambda = 2.5I_4$, $\Phi_a = 1.5I_4$, $\Phi_u = 1.5[e_1\ e_3]$, $\Upsilon_u = [e_1\ e_3]$, $e_1 = [1\ 0\ 0\ 0]^T$, $e_3 = [0\ 0\ 1\ 0]^T$, $E = 0.5$, $\epsilon = 1$, $\bar{\epsilon} = 3$, $\sigma_0 = 2$, $\sigma_1 = \sigma_2 = 50$, $\gamma_0 = 3$, $\gamma_1 = \gamma_2 = 0.5$, $\nu = 0.01$.

B. Results and Discussions

Fig. 1(d) shows three paths. The reference path is generated by the standard Ardupilot guidance in ideal no-wind conditions. Then, we have the actual Ardupilot path and the proposed path in the presence of wind. All the tests have an average wind disturbance of 6 m/s pointing in South–West direction and perturbed by a Dryden turbulence. Therefore, in the plane of Fig. 1(d), the UAV goes "with the wind" along the straight line, but it can be seen that the Ardupilot performance degrades when the UAV goes "against the wind."

[1] https://ardupilot.org/plane/docs/automatic-tuning-with-autotune.html

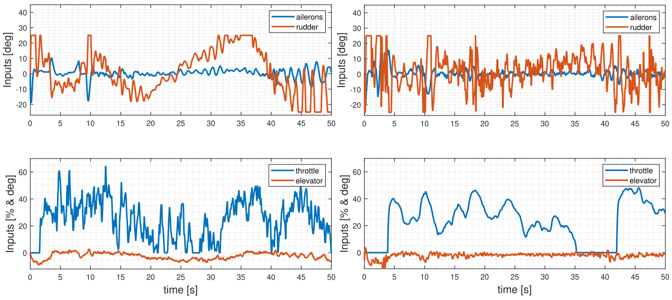


(a) (b) (c)

Fig. 4. Orbit path experiments: path following and tracking errors. (a) x–y plane (black UAV is out of scale). (b) Ardupilot tracking errors. (c) Proposed tracking errors.

TABLE II
TRACKING ERROR COSTS FOR AUTOPILOTS WITH ADAPTIVE GAINS (PROPOSED) AND WITH FIXED SLIDING MODE CONTROL GAINS (SMC) OF INCREASING MAGNITUDE

Proposed	SMC	SMC	SMC	SMC	SMC	SMC
adaptive	$\hat{\vartheta}_0 = 1$	$\hat{\vartheta}_0 = 2$	$\hat{\vartheta}_0 = 5$	$\hat{\vartheta}_0 = 10$	$\hat{\vartheta}_0 = 15$	$\hat{\vartheta}_0 = 20$
2.15	4.50	4.13	3.04	2.73	6.40	8.16
	(+109%)	(+92%)	(+41%)	(+27%)	(+198%)	(+280%)



(a) (b)

Fig. 5. Orbit path experiments: inputs. (a) Ardupilot actuation. (b) Proposed actuation.

A complete comparison is reported in Figs. 2 and 3 for a straight line at constant altitude. In the first phase, the UAV approaches the desired line by traveling "orthogonal" to it, and then the course becomes "parallel" to the line as the UAV approaches it. Again, there is a wind disturbance of 6 m/s (on average) pointing in South–West direction. From the plane in Fig. 2(a), it is clear that Ardupilot is strongly affected by the wind and cannot counteract it effectively. From the control inputs in Fig. 3(a) (for Ardupilot) and Fig. 3(b) (for the proposed approach), it can be seen that the Ardupilot saturates the rudder actuator without managing to counteract the wind.

The tracking errors in Fig. 2(b) (Ardupilot) and Fig. 2(c) (proposed approach) are calculated with respect to the reference path. In particular, the tracking errors are calculated with respect to linear velocities and attitude angles, because guidance algorithms provide velocities and attitudes as commands to the low-level autopilot. These figures confirm that, for the proposed approach, the errors are small despite the presence of wind (only the roll error is comparable in size).

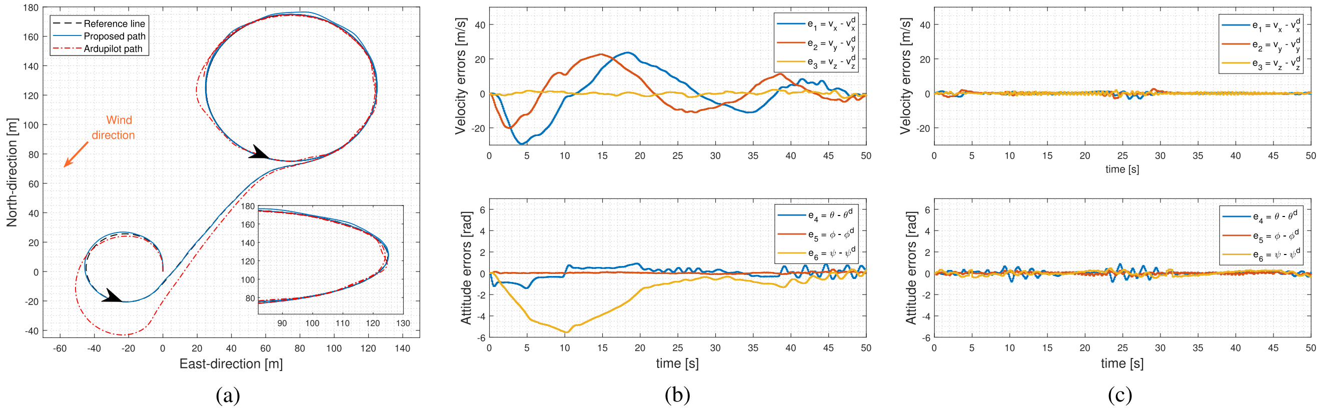
Another evaluation is reported in Figs. 4 and 5 for a combination of orbits and line paths. Differently from Fig. 1(d), the UAV travels "against the wind" along the straight line.

A large deviation from the reference path is reported for Ardupilot in Fig. 4(a), due to the wind. The transition from "with the wind" to "against the wind" is difficult for Ardupilot, while the proposed approach can successfully manage this transition. This is further confirmed in the similar transition that occurs along the large orbit and causing another deviation from the reference path. The control inputs are reported in Fig. 5(a) (Ardupilot) and Fig. 5(b) (proposed approach).

The tracking errors in Fig. 4(b) (Ardupilot) and Fig. 4(c) (proposed approach) further confirm the effectiveness of the proposed approach. In particular, it can be seen that Ardupilot exhibits a large oscillation in the velocities, as well as the aforementioned transient in the yaw error. The experiments confirm that the proposed design is able to maintain performance in the presence of significant wind disturbances. While standard autopilot suites can be affected by such disturbances, the trajectory of the proposed method turns out to be close to the reference path calculated in the absence of winds.

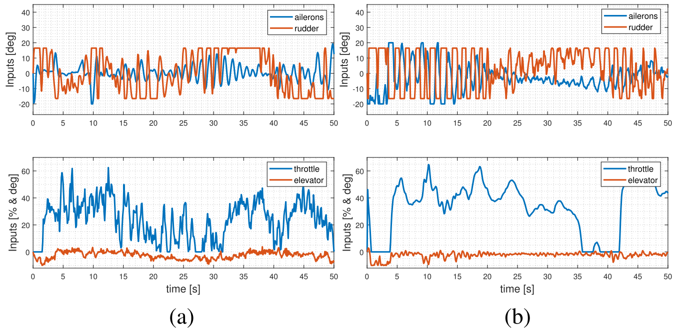
C. Comparisons With a Robust Approach

In order to validate the benefits of the proposed adaptation, we also run a standard sliding mode version of the proposed approach, which amounts to a control action similar to (19)–(20), but where the adaptation laws are switched OFF. In particular, we take $\hat{\vartheta}_1 = \hat{\vartheta}_2 = 0$ (i.e., as in standard sliding mode literature), and with $\hat{\vartheta}_0 = \text{const}$, for various constants $= (1, 2, 5, 10, 15, 20)$. The performance of the different autopilots (called SMC autopilots) is compared with the proposed idea in Table II, in terms of tracking error with the desired path. Two aspects are worth remarking. First, increasing $\hat{\vartheta}_0$ from 1 to 10 initially leads improving the performance, but increasing $\hat{\vartheta}_0$ beyond 10 leads to performance degradation: this aspect was anticipated in Remark 6, i.e., larger gains are not always beneficial. The second point is that the proposed "adaptive gain" method works better than any "fixed gain," irrespective of the choice of the fixed gain: this further validates the intelligence of the proposed law in adapting to the uncertainty.



(a) (b) (c)

Fig. 6. Orbit path experiments with tighter saturation: path following and tracking errors. (a) x-y plane (black UAV is out of scale). (b) Ardupilot tracking errors. (c) Proposed tracking errors.



(a) (b)

Fig. 7. Orbit path experiments with tighter saturation: inputs. (a) Ardupilot actuation. (b) Proposed actuation.

D. Experiments With Tighter Saturation

For the fixed-wing UAV under consideration, the aileron angles are constrained in ±30 degrees, the elevator angles are constrained in ±15 degrees, the rudder angles are constrained in ±25 degrees. One can notice that some saturation arises (especially in the rudder angle) and this is due to the fact that the the UAV flies against a quite strong wind of 4 m/s, which is 27% of the nominal UAV speed (15 m/s): when the wind hits the UAV diagonally, the rudder might require a large angle to counteract the effect of the wind. It is worth noticing that the rudder of the proposed strategy [Figs. 3(b) and 5(b)] is less affected by saturation than the original Ardupilot [Figs. 3(a) and 5(a)], which happens because a better control of throttle and aileron can counteract the wind and require less effort from the rudder.

To further investigate the performance of the proposed controller, we perform additional tests where we tighten *on purpose* the saturation limits. Now, the aileron angles are constrained in ±20 degrees, the elevator angles are constrained in ±10 degrees, the rudder angles are constrained in ±16.5 degrees. This amounts to introducing 33.3% more saturation in the UAV actuators. We compare the proposed autopilot with the original Ardupilot. The comparisons are shown in Figs. 6 and 7. In particular, Fig. 6(a) shows that the proposed method again outperforms the original ArduPilot. Although both methods degrade their original performance as compared to the nominal saturation case, the comparisons between the proposed method in Fig. 6(b) and the Ardupilot method in Fig. 6(c) show that the proposed method has still much smaller error than the Ardupilot autopilot. Comparisons in terms of inputs are in Fig. 7(a) and (b). These comparisons suggest that the proposed method can handle the presence of tighter saturation constraints, which may stimulate future research on tackling the effect of saturation in the stability analysis.

V. Conclusion

No matter how intelligent the design of a guidance law for fixed-wing UAVs, the guidance inputs must be passed to the low-level autopilot to control the flying surfaces and the thrust. As a result, the final performance of the UAV is ultimately affected by the autopilot, which is usually not effective in dealing with uncertainty. As most autopilots rely on the knowledge of the UAV trimming points or on linearized dynamics, in this work, we have explored the design of autopilots where such knowledge is not required. The proposed autopilot was designed based on uncertain Euler–Lagrange dynamics of the UAV and tested in hardware-in-the-loop experiments.

Appendix
Error Dynamics

Using (16a) and (16b), the time derivative of (18) yields

$$\begin{aligned}\dot{\eta} &= (I_4 - \Upsilon_u M_{uu}^{-1} M_{au}^T)(M_s^{-1}\delta + h_a)\\ &\quad + \Upsilon_u h_u + \Phi_a \dot{e}_a + \Phi_u \dot{e}_u\\ &= M_s \delta + h_s + \Phi_a \dot{e}_a + \Phi_u \dot{e}_u \qquad (24)\end{aligned}$$

with $M_s \triangleq \bar{M}^{-1}\bar{B}_a - \Upsilon_u M_{uu}^{-1}\bar{B}_u$ and $h_s \triangleq -\bar{M}^{-1}\bar{h}_a - \Upsilon_u M_{uu}^{-1}\bar{h}_u$.

Substituting the control (19) gives

$$\dot{\eta} = -\Lambda\eta - \bar{\delta} + \Psi - (M_s \hat{M}_s^{-1} - I_m)\bar{\delta} \qquad (25)$$

where $\Psi \triangleq h_s - (M_s \hat{M}_s^{-1} - I_4)(\Lambda\eta + \Phi_a \dot{e}_a + \Phi_u \dot{e}_u)$. From Assumptions 1 and 2, and along similar lines as [46], one can find the following upper bound:

$$\begin{aligned}||\Psi|| &= ||h_s - (M_s \hat{M}_s^{-1} - I_4)(\Lambda\eta + \Phi_a \dot{e}_a + \Phi_u \dot{e}_u)||\\ &\leq ||h_s|| + E(||\Lambda|| ||\eta|| + ||\Phi_a \dot{e}_a + \Phi_u \dot{e}_u||)\end{aligned}$$

$$\leq \vartheta_0^* + \vartheta_1^*||\xi|| + \vartheta_2^*||\xi||^2 \tag{26}$$

with $\vartheta_i^* \in \mathbb{R}^+$, $i = 0, 1, 2$ being *unknown constant* due uncertainty (cf. Assumption 1).

It is possible to see that the dynamics of $x = [e_u{}^T\ \dot{e}_u^T]^T$ are

$$\dot{x} = \underbrace{\begin{bmatrix} 0 & I_2 \\ -\bar{\Lambda}\Lambda\Phi_u & -\bar{\Lambda}\Lambda\Upsilon_u \end{bmatrix}}_{A_m} x + \underbrace{\begin{bmatrix} 0 \\ I_m \end{bmatrix}}_{B_m} \bar{B}\bar{\delta} + \bar{\Psi} \tag{27}$$

where $\bar{B} \triangleq M_{uu}^{-1}\bar{B}_u\hat{M}_s^{-1}$, $\bar{\Psi} \triangleq \bar{B}(\Phi_a\dot{e}_a + \Phi_u\dot{e}_u) + (H + \bar{B})\Lambda\eta - M_{uu}^{-1}\bar{h}_u - H\Lambda(\dot{e}_a + \Phi_a e_a)$, and the gains in A_m are such that

$$\bar{\Lambda}\Lambda\Phi_u > 0,\ \bar{\Lambda}\Lambda\Upsilon_u > 0 \tag{28}$$

for a full-rank matrix $\bar{\Lambda} \in \mathbb{R}^{2\times 4}$.

The following upper bound:

$$||\bar{\Psi}||||PB_m|| \leq \left(\vartheta_0^{**} + \vartheta_1^{**}||\xi|| + \vartheta_2^{**}||\xi||^2\right) \tag{29}$$

where $\vartheta_i^{**} \in \mathbb{R}^+$, $i = 0, 1, 2$ are unknown scalars holds from similar lines as (26); $P > 0$ is the solution to the Lyapunov equation $A_m^T P + PA_m = -Q$ for some $Q > 0$ [A_m is Hurwitz since the gains in (28) are positive definite]. The vectors Ψ and $\bar{\Psi}$ act as the uncertainty in the closed-loop dynamics.

Remark 6 (State-dependent disturbances): Note that the quadratic state-dependent bound of the disturbances d_1 and d_2 (cf. Assumption 1) is eventually reflected as a quadratic error-dependent bound in $\bar{h}_a$ and $\bar{h}_u$, which finally appears in the quadratic forms (26) and (29). This implies that higher-order Taylor terms neglected in the aerodynamic coefficients in (13) can be captured by such state dependency.

A. Proof of Theorem 1

Stability is analyzed via the Lyapunov function

$$V = \frac{1}{2}\left\{\eta^T\eta + x^T Px + \sum_{i=0}^{2}\frac{1}{\kappa_i}(\hat{\vartheta}_i - \bar{\vartheta}_i^*)^2 + \frac{\gamma^2}{\gamma_0}\right\} \tag{30}$$

where $\bar{\vartheta}_i^* = \max\{\vartheta_i^*, \vartheta_i^{**}\}$. Analysis considers the two cases (i) $||\eta|| \geq \epsilon$ and (ii) $||\eta|| < \epsilon$.

Case (i) $||\eta|| \geq \epsilon$

Using (19), (26), and (20), we have

$$\begin{aligned} \eta^T\dot{\eta} &\leq -\eta^T\Lambda\eta - (1-E)\mu||\eta|| + \sum_{i=0}^{2}\vartheta_i^*||\xi||^i||\eta|| \\ &\leq -\eta^T\Lambda\eta - \sum_{i=0}^{2}(\hat{\vartheta}_i||\xi||^i + \gamma)||\eta|| + \bar{\vartheta}_i^*||\xi||^i||\eta||. \end{aligned} \tag{31}$$

and

$$\begin{aligned} \frac{1}{2}(\dot{x}^T Px + x^T P\dot{x}) &= -\frac{1}{2}x^T Qx + x^T PB_m(\bar{B}\bar{\delta} + \bar{\Psi}) \\ &\leq -\frac{1}{2}x^T Qx + \mu E_1||x|| + ||\bar{\Psi}||||PB_m||||x||. \end{aligned} \tag{32}$$

The upper bound (29) further gives

$$\begin{aligned} \frac{1}{2}(\dot{x}^T Px + x^T P\dot{x}) &\leq -\frac{1}{2}x^T Qx \\ &+ \sum_{i=0}^{2}\bar{\vartheta}_i^*||\xi||^i||x|| + \frac{E_1}{1-E}\sum_{i=0}^{2}(\hat{\vartheta}_i||\xi||^i + \gamma)||x||. \end{aligned} \tag{33}$$

The adaptive laws (21a) and (21b) give, $i = 0, 1, 2$

$$\begin{aligned} \frac{1}{\kappa_i}(\hat{\vartheta}_i - \bar{\vartheta}_i^*)\dot{\hat{\vartheta}}_i &= \hat{\vartheta}_i(||\eta|| + ||x||)||\xi||^i - c_i\hat{\vartheta}_i^2||x||||\xi||^i \\ &- \bar{\vartheta}_i^*(||\eta|| + ||x||)||\xi||^i + c_i\hat{\vartheta}_i\bar{\vartheta}_i^*||x||||\xi||^i \end{aligned} \tag{34}$$

$$\begin{aligned} \frac{\gamma\dot{\gamma}}{\gamma_0} &= \gamma(||\eta|| + ||x||) - \gamma^2\{1 + \bar{\gamma}(||\xi||^5 - ||\xi||^4) \\ &+ c_3(||x|| + ||\xi||)\} + \gamma\nu \end{aligned} \tag{35}$$

for positive $c_i \triangleq \frac{\sigma_i}{\kappa_i}\bar{\epsilon}$, $c_3 \triangleq \frac{\gamma_2}{\gamma_0}$ and $\bar{\gamma} \triangleq \frac{\gamma_1}{\gamma_0}$. Steps (34) and (35) give

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\left(\sum_{i=0}^{2}\frac{(\hat{\vartheta}_i - \bar{\vartheta}_i^*)^2}{2\kappa_i} + \frac{\gamma^2}{2\gamma_0}\right) &= \sum_{i=0}^{2}\hat{\vartheta}_i(||\eta|| + ||x||)||\xi||^i \\ &- c_i\hat{\vartheta}_i^2||x||||\xi||^i - \bar{\vartheta}_i^*(||\eta|| + ||x||)||\xi||^i + c_i\hat{\vartheta}_i\bar{\vartheta}_i^*||x||||\xi||^i \\ &- \gamma^2\{1 + \bar{\gamma}(||\xi||^5 - ||\xi||^4) + c_3(||x|| + ||\xi||))\} \\ &+ \gamma(||\eta|| + ||x||) + \gamma\nu. \end{aligned} \tag{36}$$

Using (31), (33), and (36), the time derivative of (30) becomes

$$\begin{aligned} \dot{V} \leq\ & -\omega_m(||\eta||^2 + ||x||^2) + \gamma\nu + c\gamma||x|| \\ & - \gamma^2\{1 + \bar{\gamma}(||\xi||^5 - ||\xi||^4) + c_3||x||\} \\ & + \sum_{i=0}^{2}(c\hat{\vartheta}_i - c_i\hat{\vartheta}_i^2 + c_i\hat{\vartheta}_i\bar{\vartheta}_i^*)||\xi||^i||x|| \end{aligned} \tag{37}$$

where $\omega_m \triangleq \min\{\lambda_{\min}(\Lambda), (1/2)\lambda_{\min}(Q)\}$ and $\bar{E} \triangleq 1 + \frac{E_1}{1-E}$. Because $\hat{\vartheta}_i(t) \geq 0$, the following bound holds:

$$V \leq \omega_M(||\eta||^2 + ||x||^2) + \sum_{i=0}^{2}\frac{(\hat{\vartheta}_i^2 + \bar{\vartheta}_i^{*^2})}{\kappa_i} + \frac{\gamma^2}{\gamma_0} \tag{38}$$

where $\omega_M \triangleq \max\{1, ||P||\}$. Defining $\Omega \triangleq (\omega_m/\omega_M)$ and using (38), the time derivative (37) further becomes

$$\begin{aligned} \dot{V} \leq\ & -\Omega V + \sum_{i=0}^{2}\frac{\Omega}{\kappa_i}(\hat{\vartheta}_i^2 + \bar{\vartheta}_i^*)^2 + \frac{\Omega}{\gamma_0}\gamma^2 \\ & + \sum_{i=0}^{2}(\bar{E}\hat{\vartheta}_i - c_i\hat{\vartheta}_i^2 + c_i\hat{\vartheta}_i\bar{\vartheta}_i^*)||\xi||^i||x|| + \gamma\nu + \bar{E}\gamma||x|| \\ & - \gamma^2(1 + \bar{\gamma}(||\xi||^5 - ||\xi||^4) + c_3||x||). \end{aligned} \tag{39}$$

Since c_i and c_3 are positive, they can be split as $c_i = \sum_{j=1}^{3} c_{ij}$, $c_3 = \sum_{k=1}^{2} c_{3\,k}$, with $c_{ij}, c_{3\,k} > 0$, leading to

$$\begin{aligned} & - c_i\hat{\vartheta}_i^2 + \bar{E}\hat{\vartheta}_i + c_i\hat{\vartheta}_i\bar{\vartheta}_i^* \\ & = -c_{i1}\hat{\vartheta}_i^2 - c_{i2}\left\{\left(\hat{\vartheta}_i - \frac{\bar{E}}{2c_{i2}}\right)^2 - \frac{\bar{E}^2}{4c_{i2}^2}\right\} \end{aligned}$$

$$
\begin{aligned}
&- c_{i3}\left\{\left(\hat{\vartheta}_i - \frac{c_i\bar{\vartheta}_i^*}{2c_{i3}}\right)^2 - \frac{(c_i\bar{\vartheta}_i^*)^2}{4c_{i3}^2}\right\} \\
&\le -c_{i1}\hat{\vartheta}_i^2 + \frac{\bar{E}^2}{4c_{i2}} + \frac{(c_i\bar{\vartheta}_i^*)^2}{4c_{i3}}. \qquad (40)
\end{aligned}
$$

Analogously

$$
\begin{aligned}
&- \gamma^2(1 + c_3||x||) \\
&+ \gamma\nu + \bar{E}\gamma||x|| \le \left(-c_{31}\gamma^2 + \frac{\bar{E}^2}{4c_{32}}\right)||x|| + \frac{\nu^2}{4}. \qquad (41)
\end{aligned}
$$

The adaptive law (21b) with initial conditions (21c) reveal that $\gamma(t) \ge \underline{\gamma} > 0\ \forall t \ge 0$ for some positive scalar $\underline{\gamma}$. Using this and (40) and (41), the time derivative (39) becomes

$$
\begin{aligned}
\dot{V} \le& -\Omega V + \sum_{i=0}^{2}\frac{\Omega}{\kappa_i}(\hat{\vartheta}_i^2 + (\bar{\vartheta}_i^*)^2) + \frac{\Omega\gamma^2}{\gamma_0} + \frac{\nu^2}{4} \\
&+ \sum_{i=0}^{2}\left(\frac{\bar{E}^2}{4c_{i2}} + \frac{\left(c_i\bar{\vartheta}_i^*\right)^2}{4c_{i3}}\right)||\xi||^{i+1} - c_{i1}\hat{\vartheta}_i^2||x||^{i+1} \\
&- \underline{\gamma}^2\bar{\gamma}(||\xi||^5 - ||\xi||^4) - c_{31}\gamma^2||x|| + \left(\bar{E}^2/4c_{32}\right)||x|| \\
=& -\Omega V - \hat{\vartheta}_0^2\left(c_{01}||x|| - (\Omega/\kappa_0)\right) + f(||\xi||) \\
&- \hat{\vartheta}_1^2\left(c_{11}||x||^2 - (\Omega/\kappa_1)\right) - \hat{\vartheta}_2^2\left(c_{21}||x||^3 - (\Omega/\kappa_2)\right) \\
&- \gamma^2\left(c_{31}||x|| - (\Omega/\gamma_0)\right) \qquad (42)
\end{aligned}
$$

$$
\begin{aligned}
\text{where } f(||\xi||) \triangleq& -\underline{\gamma}^2\bar{\gamma}||\xi||^5 + \omega_4||\xi||^4 + \omega_3||\xi||^3 \\
&+ \omega_2||\xi||^2 + \omega_1||\xi|| + \omega_0
\end{aligned}
$$

$$
\omega_0 \triangleq \sum_{i=0}^{2}\frac{\Omega}{\kappa_i}(\bar{\vartheta}_i^*)^2 + \frac{\nu^2}{4}, \omega_3 \triangleq \frac{\bar{E}^2}{4c_{22}} + \frac{\left(c_2\bar{\vartheta}_2^*\right)^2}{4c_{23}}, \omega_4 \triangleq \underline{\gamma}^2\bar{\gamma}
$$

$$
\omega_2 \triangleq \frac{\bar{E}^2}{4c_{12}} + \frac{\left(c_1\bar{\vartheta}_1^*\right)^2}{4c_{13}}, \omega_1 \triangleq \frac{\bar{E}^2}{4c_{02}} + \frac{\left(c_0\bar{\vartheta}_0^*\right)^2}{4c_{03}} + \frac{\bar{E}^2}{4c_{32}}.
$$

Using Descartes' rule of sign change and Bolzano's Theorem, it can be verified that the polynomial f has exactly one positive real root (call it $\iota \in \mathbb{R}^+$). The coefficient of the highest degree of f is negative as $\underline{\gamma}^2\bar{\gamma} \in \mathbb{R}^+$. Therefore, $f(||\xi||) \le 0$ when $||\xi|| \ge \iota$. Define $\iota_0 \triangleq \frac{\Omega}{\kappa_0 c_{01}}, \iota_1 \triangleq \sqrt{\frac{\Omega}{\kappa_1 c_{11}}}, \iota_2 \triangleq (\frac{\Omega}{\kappa_2 c_{21}})^{1/3}$ and $\iota_3 \triangleq \frac{\Omega}{\gamma_0 c_{31}}$. Hence, from (42), $\dot{V} \le -\Omega V$ when

$$
\begin{aligned}
\min\{||x||, ||\xi||\} &\ge \max\{\iota,\ \iota_0,\ \iota_1,\ \iota_2,\ \iota_3\} \\
&\Rightarrow ||x|| \ge \max\{\iota,\ \iota_0,\ \iota_1,\ \iota_2,\ \iota_3\}. \qquad (43)
\end{aligned}
$$

Case (ii) $||\eta|| < \epsilon$

Using a similar procedure as Case (i), the following time derivative holds for Case (ii):

$$
\begin{aligned}
\dot{V} \le& -\Omega V - \hat{\vartheta}_0^2\left(c_{01}||x|| - (\Omega/\kappa_0) - 1\right) + f_1(||\xi||) \\
&- \hat{\vartheta}_1^2\left(c_{11}||x||^2 - (\Omega/\kappa_1) - 1\right) - \hat{\vartheta}_2^2\left(c_{21}||x||^3\right. \\
&\left.- (\Omega/\kappa_2) - 1\right) - \gamma^2\left(c_{31}||x|| - (\Omega/\gamma_0) - 1\right) \qquad (44)
\end{aligned}
$$

$$
\begin{aligned}
\text{where } f_1(||\xi||) \triangleq& -\underline{\gamma}^2\bar{\gamma}||\xi||^5 + \omega_4'||\xi||^4 + \omega_3||\xi||^3 \\
&+ \omega_2'||\xi||^2 + \omega_1||\xi|| + \omega_0'
\end{aligned}
$$

$$
\omega_4' \triangleq \omega_4 + \frac{\epsilon^2}{4}, \omega_2' \triangleq \omega_2 + \frac{\epsilon^2}{4}
$$

$$
\omega_0' \triangleq \sum_{i=0}^{2}\frac{\Omega}{\kappa_i}(\bar{\vartheta}_i^*)^2 + \frac{(\nu+\epsilon)^2}{4} + \frac{\epsilon^2}{4}.
$$

Along similar lines as Case (i), $\dot{V} \le -\Omega V$ is guaranteed when

$$
||x|| \ge \max\{\iota,'\ \iota_0,'\ \iota_1,'\ \iota_2,'\ \iota_3'\} \qquad (45)
$$

where ι' is the only positive real root of the polynomial f_1 and $\iota_0' \triangleq (\frac{\Omega}{\kappa_0 c_{01}} + \frac{1}{c_{01}})$, $\iota_1' \triangleq \sqrt{\frac{\Omega}{\kappa_1 c_{11}} + \frac{1}{c_{11}}}$, $\iota_2' \triangleq (\frac{\Omega}{\kappa_2 c_{21}} + \frac{1}{c_{21}})^{1/3}$, and $\iota_3' \triangleq (\frac{\Omega}{\gamma_0 c_{31}} + \frac{1}{c_{31}})$.

Hence, investigating the results for Cases (i) and (ii), bounded Lyapunov function can be concluded, which implies the signals $\eta, e_u, \dot{e}_u, \hat{\vartheta}_i, \gamma \in \mathscr{L}_\infty$. Let us now write (18) as

$$
\dot{e}_a = -\Phi_a e_a - (\Upsilon_u\dot{e}_u + \Phi_u e_u) + \eta. \qquad (46)
$$

Using $\Phi_a > 0$ and $\eta, e_u, \dot{e}_u \in \mathscr{L}_\infty$, we conclude that $e_a, \dot{e}_a \in \mathscr{L}_\infty$, which makes all closed-loop signals bounded.

Remark 7 (Tuning considerations): Convergence of the non-actuated dynamics can be tuned via $\Lambda\Phi_u$ and $\Lambda\Upsilon_u$ in (27); convergence of the actuated dynamics can be tuned via Φ_a, as shown in (46). Large values of κ_i, γ_1, γ_2, and ν in the adaptive laws (21) increase the effect of the negative fifth-degree term of the polynomials $f(||\xi||)$ and $f_1(||\xi||)$, defined after (42) and (44): this means that the real root of these polynomials becomes smaller, thus decreasing the error bound. However, too large gains might lead to high control inputs (and possibly saturation in the case of the UAV). Therefore, these gains should be tuned according to performance requirements.

References

[1] F. S. Leira, K. Trnka, T. I. Fossen, and T. A. Johansen, "A ligth-weight thermal camera payload with georeferencing capabilities for small fixed-wing UAVs," in *Proc. Int. Conf. Unmanned Aircr. Syst.*, 2015, pp. 485–494.

[2] L. Furieri, T. Stastny, L. Marconi, R. Siegwart, and I. Gilitschenski, "Gone with the wind: Nonlinear guidance for small fixed-wing aircraft in arbitrarily strong windfields," in *Proc. Amer. Control Conf.*, 2017, pp. 4254–4261.

[3] Y. B. Shtessel, C. H. Tournes, and L. Fridman, "Advances in guidance and control of aerospace vehicles using sliding mode control and observation techniques," *J. Franklin Inst.*, vol. 349, no. 2, pp. 391–396, 2012.

[4] P. Sujit, S. Saripalli, and J. B. Sousa, "Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicless," *IEEE Control Syst. Mag.*, vol. 34, no. 1, pp. 42–59, Feb. 2014.

[5] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton, NJ, USA: Princeton Univ. Press, 2012.

[6] W. Yao, H. G. de Marina, B. Lin, and M. Cao, "Singularity-free guiding vector field for robot navigation," *IEEE Trans. Robot.*, vol. 37, no. 4, pp. 1206–1221, Aug. 2021.

[7] K. Tanaka, M. Tanaka, A. Iwase, and H. O. Wang, "A rational polynomial tracking control approach to a common system representation for unmanned aerial vehicles," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 2, pp. 919–930, Apr. 2020.

[8] L. Fusini, T. I. Fossen, and T. A. Johansen, "Nonlinear observers for GNSS- and camera-aided inertial navigation of a fixed-wing UAV," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 5, pp. 1884–1891, Sep. 2018.

[9] D. Tang, Q. Fang, L. Shen, and T. Hu, “Onboard detection-tracking-localization,” *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1555–1565, Jun. 2020.
[10] ArduPilot, “Roll, pitch and yaw controller tuning,” 2021. [Online]. Available: http://ardupilot.org/plane/docs/roll-pitch-controller-tuning.html
[11] PX4 User Guide, “Controller diagrams,” 2021. [Online]. Available: https://docs.px4.io/master/en/flight-stack/controller-diagrams.html
[12] G. Ellingson and T. McLain, “ROSplane: Fixed-wing autopilot for education and research,” in *Proc. Int. Conf. Unmanned Aircr. Syst.*, 2017, pp. 1503–1507.
[13] C. B. Panathula, A. Rosales, Y. B. Shtessel, and L. M. Fridman, “Closing gaps for aircraft attitude higher order sliding mode control certification via practical stability margins identification,” *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 6, pp. 2020–2034, Nov. 2018.
[14] P. Poksawat, L. Wang, and A. Mohamed, “Gain scheduled attitude control of fixed-wing UAV with automatic controller tuning,” *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp. 1192–1203, Jul. 2018.
[15] D. Invernizzi, M. Lovera, and L. Zaccarian, “Integral ISS-based cascade stabilization for vectored-thrust UAVs,” *IEEE Contr. Syst. Lett.*, vol. 4, no. 1, pp. 43–48, Jan. 2020.
[16] B. Zhou, H. Satyavada, and S. Baldi, “Adaptive path following for unmanned aerial vehicles in time-varying unknown wind environments,” in *Proc. Amer. Control Conf.*, 2017, pp. 1127–1132.
[17] V. P. Tran, F. Santoso, M. Garratt, and I. R. Petersen, “Distributed formation control using fuzzy self-tuning of strictly negative imaginary consensus controllers in aerial robotics,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 5, pp. 2306–2315, Oct. 2021.
[18] N. Cho, S. Lee, J. Kim, Y. Kim, S. Park, and C. Song, “Wind compensation framework for unpowered-aircraft using online waypoint correction,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 1, pp. 698–710, Feb. 2020.
[19] S. Fari, X. Wang, S. Roy, and S. Baldi, “Addressing unmodelled path-following dynamics via adaptive vector field: A UAV test case,” *IEEE Trans, Aerosp. Electron. Syst.*, vol. 56, no. 2, pp. 1613–1622, Apr. 2019.
[20] S. Abdelmoeti and R. Carloni, “Robust control of UAVs using the parameter space approach,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 5632–5637.
[21] D. Galzi and Y. Shtessel, “Uav formations control using high order sliding modes,” in *Proc. Amer. Control Conf.*, 2006, pp. 1–6.
[22] Z. Sun, H. Garcia de Marina, B. D. O. Anderson, and C. Yu, “Collaborative target-tracking control using multiple fixed-wing unmanned aerial vehicles with constant speeds,” *J. Guid., Control, Dyn.*, vol. 44, no. 2, pp. 238–250, 2021.
[23] Y. Wang, Y. Yue, M. Shan, L. He, and D. Wang, “Formation reconstruction and trajectory replanning for multi-UAV patrol,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 2, pp. 719–729, Apr. 2021.
[24] N. Cho and Y. Kim, “Optimality of augmented ideal proportional navigation for maneuvering target interception,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 2, pp. 948–954, Apr. 2016.
[25] Y. Lin and S. Saripalli, “Sampling-based path planning for UAV collision avoidance,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 11, pp. 3179–3192, Nov. 2017.
[26] E. Bertran and A. Sànchez-Cerdà, “On the tradeoff between electrical power consumption and flight performance in fixed-wing UAV autopilots,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 8832–8840, Nov. 2016.
[27] A. Bosso, C. Conficoni, D. Raggini, and A. Tilli, “A computational-effective field-oriented control strategy for accurate and efficient electric propulsion of unmanned aerial vehicles,” *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 3, pp. 1501–1511, Jun. 2021.
[28] P. Hemakumara and S. Sukkarieh, “Learning UAV stability and control derivatives using Gaussian processes,” *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 813–824, Aug. 2013.
[29] B. Li, J. Sun, W. Zhou, C. Y. Wen, K. H. Low, and C. K. Chen, “Transition optimization for a VTOL tail-sitter UAV,” *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 5, pp. 2534–2545, Oct. 2020.
[30] M. Trittler, T. Rothermel, and W. Fichter, “Autopilot for landing small fixed-wing unmanned aerial vehicles with optical sensors,” *J. Guid., Control, Dyn.*, vol. 39, no. 9, pp. 2011–2021, 2016.
[31] R. Naldi, F. Forte, A. Serrani, and L. Marconi, “Modeling and control of a class of modular aerial robots combining under actuated and fully actuated behavior,” *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 1869–1885, Sep. 2015.
[32] M. Mammarella, E. Capello, F. Dabbene, and G. Guglieri, “Sample-based SMPC for tracking control of fixed-wing UAV,” *IEEE Control Syst. Lett.*, vol. 2, no. 4, pp. 611–616, Oct. 2018.
[33] Y. Yan, J. Yang, C. Liu, M. Coombes, S. Li, and W. Chen, “On the actuator dynamics of dynamic control allocation for a small fixed-wing UAV with direct lift control,” *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 3, pp. 984–991, May 2020.
[34] J. M. Levin, A. A. Paranjape, and M. Nahon, “Agile maneuvering with a small fixed-wing unmanned aerial vehicle,” *Robot. Auton. Syst.*, vol. 116, pp. 148–161, 2019.
[35] J. M. Beach, M. E. Argyle, T. W. McLain, R. W. Beard, and S. Morris, “Tailsitter attitude control using resolved tilt-twist,” in *Proc. Int. Conf. Unmanned Aircr. Syst.*, 2014, pp. 768–779.
[36] J. Stephan, O. Pfeifle, S. Notter, F. Pinchetti, and W. Fichter, “Precise tracking of extended three-dimensional dubins paths for fixed-wing aircraft,” *J. Guid., Control, Dyn.*, vol. 43, no. 12, pp. 2399–2405, 2020.
[37] T. Yamasaki, S. N. Balakrishnan, and H. Takano, “Separate-channel integrated guidance and autopilot for automatic path-following,” *J. Guid., Control, Dyn.*, vol. 36, no. 1, pp. 25–34, 2013.
[38] A. Galffy, M. Böck, and A. Kugi, “Nonlinear 3D path following control of a fixed-wing aircraft based on acceleration control,” *Control Eng. Pract.*, vol. 86, pp. 56–69, 2019.
[39] J.-M. Kai, T. Hamel, and C. Samson, “A unified approach to fixed-wing aircraft path following guidance and control,” *Automatica*, vol. 108, 2019, Art. no. 108491.
[40] S. Baldi, S. Roy, and K. Yang, “Towards adaptive autopilots for fixed-wing unmanned aerial vehicles,” in *Proc. 59th IEEE Conf. Decis. Control*, 2020, pp. 4724–4729.
[41] D. Invernizzi, M. Giurato, P. Gattazzo, and M. Lovera, “Comparison of control methods for trajectory tracking in fully actuated unmanned aerial vehicles,” *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 3, pp. 1147–1160, May 2020.
[42] S. Farí, “Guidance and control for a fixed-wing UAV,” M.S. thesis, Politecnico di Milano, 2017, doi: 10.13140/RG.2.2.24973.28641.
[43] “Description of digital DATCOM,” 2019. [Online]. Available: http://www.pdas.com/datcomDescription.html
[44] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*. Hoboken, NJ, USA: Wiley, 2008.
[45] S. Roy, S. Baldi, and P. A. Ioannou, “An adaptive control framework for underactuated switched euler-lagrange systems,” *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2021.3108507.
[46] S. Roy and S. Baldi, “Towards structure-independent stabilization for uncertain underactuated Euler–Lagrange systems,” *Automatica*, vol. 113, 2020, Art. no. 108775.
[47] C. D. Cruz-Ancona, M. A. Estrada, and L. Fridman, “Barrier function-based adaptive Lyapunov redesign for systems without a priori bounded perturbations,” *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2021.3107453.
[48] S. Roy, S. Baldi, and L. M. Fridman, “On adaptive sliding mode control without a priori bounded uncertainty,” *Automatica*, vol. 111, 2020, Art. no. 108650.
[49] SITL_gazebo, “Plane.sdf.jinja,” 2021. [Online]. Available: https://github.com/PX4/PX4-SITL-gazebo/blob/e580bbcd1eb6902c658ed3ece3b3b28dfd57eb17/models/plane
[50] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, “Vector field path following for miniature air vehicles,” *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 519–529, Jun. 2007.

Simone Baldi (Senior Member, IEEE) received the B.Sc. in electrical engineering, and the M.Sc. and Ph.D. degrees in automatic control from University of Florence, Florence, Italy, in 2005, 2007, and 2011.

He is currently a Professor with the School of Mathematics and School of Cyber Science and Engineering, Southeast University, with Guest Position with Delft Center for Systems and Control, TU Delft, . He was awarded outstanding reviewer for *Automatica* (2017). His research interests are adaptive and learning systems with applications in unmanned vehicles.

Prof. Baldi is a Subject Editor of *Int. Journal of Adaptive Control and Signal Processing*.

Spandan Roy received the B.Tech. degree in electronics and communication engineering from Techno India (Salt Lake), West Bengal University of Technology, West Bengal, India, in 2011, the M.Tech. degree in mechatronics from Academy of Scientific and Innovative Research, Tamil Nadu, India, in 2013, and the Ph.D. degree in control and automation from Indian Institute of Technology Delhi, Delhi, India, in 2018.

He is currently an Assistant Professor with Robotics Research Center, International Institute of Information Technology Hyderabad, India. Previously, he was Postdoc with Delft Center for System and Control, TU Delft, The Netherlands. His research interests include artificial delay based control, adaptive-robust control, and Euler–Lagrange systems.

Kang Yang received the B.Sc. degree in applied science, and the M.Sc. degree in cyber science and engineering both from Southeast University, Jiangsu, China, in 2018 and 2021, respectively. He is currently working toward the Ph.D. degree in cyber science and engineering with School of Cyber Science and Engineering, Southeast University, China.

His research interests include control of automated vehicles.

Di Liu (Member, IEEE) received the B.Sc. degree in information science from Hubei University of Science and Technology, Hebei, China, in 2014, and the M.Sc. degree in control science and engineering from Chongqing University of Posts and Telecommunications, Chongqing, China, in 2017, and the Ph.D. degree in cyber science and engineering from Southeast University, Nanjing, China, in 2021, and she is currently working toward the second Ph.D. degree with Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, Groningen, The Netherlands.

Her research interests are learning systems and control, with application in intelligent transportation, automated vehicles, unmanned aerial vehicles.