

Diagnosis of icing and actuator faults in UAVs using LPV unknown input observers

Damiano Rotondo · Andrea Cristofaro · Tor Arne Johansen · Fatiha Nejjari · Vicenç Puig

Received: date / Accepted: date

Abstract This paper proposes a discrete-time linear parameter varying (LPV) unknown input observer (UIO) for the diagnosis of actuator faults and ice accretion in unmanned aerial vehicles (UAVs). The proposed approach, which is suited to an implementation on-board, exploits a complete 6-degrees of freedom (DOF) UAV model, which includes the coupled longitudinal/lateral dynamics and the impact of icing. The LPV formulation has the advantage of allowing the icing diagnosis scheme to be consistent with a wide range of operating conditions. The developed theory is supported by simulations illustrating the diagnosis of actuator faults and icing in a small UAV. The obtained results validate the effectiveness of the proposed approach.

Keywords Unknown input observers (UIOs) · Linear parameter varying (LPV) systems · Icing diagnosis · Fault diagnosis · Unmanned aerial vehicles (UAVs)

This work has been supported by the Research Council of Norway through the Centres of Excellence funding scheme (ref. 223254 - AMOS). Damiano Rotondo is also supported by the ERCIM Alain Bensoussan Fellowship programme. This work has also been partially funded by the Spanish Government (MINECO) and FEDER through the project CICYT HARCRCIS (ref. DPI2014-58104-R).

Damiano Rotondo, Tor Arne Johansen
Centre for Autonomous Marine Operations and Systems (AMOS), Department of Engineering Cybernetics (ITK), Norwegian University of Science and Technology (NTNU), Trondheim, Norway
Phone number: +4746591822
E-mail: damiano.rotondo@yahoo.it

Andrea Cristofaro
Scuola di Scienze e Tecnologie, Università di Camerino, Camerino (MC), Italy

Damiano Rotondo, Fatiha Nejjari, Vicenç Puig
Advanced Control Systems (SAC), Universitat Politècnica de Catalunya (UPC), Terrassa, Spain

Vicenç Puig
Institut de Robòtica i Informàtica Industrial (IRI), UPC-CSIC, Barcelona, Spain

1 Introduction

Due to the increasing demand of reliability and safety, fault tolerant control techniques have been investigated widely in the last decades, with the aim of maintaining stability and acceptable performances in the event of faults [1]. As a consequence, the problem of detecting and identifying any kind of potential abnormalities and faults has been a hot topic of research, which has led to the development of several fault diagnosis techniques [2]. Fault diagnosis includes three tasks, namely fault detection (determining whether there is a fault in the system, as well as the time at which the fault occurs), fault isolation (determining the location of the faulty component), and fault estimation (providing information about the type, shape, and size of the fault). During the last decades, several approaches have been proposed as possible solutions, e.g. the geometric [3] and the observer-based [4, 5] ones.

Among the most significant faults that affect aviation safety, there is icing, i.e. the accretion of ice on aircraft wings, control surfaces and other critical locations [6]. Ice accretion decreases the lift and the static longitudinal stability, while at the same time increasing the drag and the mass of the vehicle. Consequently, it can have a profound impact on the aircraft's performance, inducing a safety risk that can potentially lead to the crash [7]. The type and severity of icing are determined by several factors, such as velocity of the airplane, exposure time, atmospheric air temperature, and liquid water content [8].

For manned aircraft, the mitigation of in-flight icing can be done by manually activating an ice protection system, which is usually either chemical, thermal or pneumatic. However, these systems are usually heavy, expensive, and structurally intricate, hence they cannot be applied to small unmanned aerial vehicles (UAVs). For this latter class of vehicles, alternative solutions have been proposed recently, such as heat conducting tapes [9] and electrically conductive carbon nanomaterial based coating for temperature control of UAV airfoil surfaces [10, 11]. However, due to power consumption, fault/icing detection schemes with fast and accurate responses are needed to guarantee the efficiency of these systems [12].

Several approaches have been studied recently for the icing detection in aircrafts and UAVs, including observer-based [13] and statistical [14] methods. An approach that has shown to be promising in the last years involves the use of unknown input observers (UIOs), a special class of observers which has found several applications for UAVs, e.g. fault detection and isolation [15]. UIOs are observers that allow estimating the state of a given system, independently of some unknown inputs [16]. First introduced in [17] and [18], this technique has been further developed using different techniques, such as the geometric [19] and the algebraic [20] approaches. UIOs are a very useful tool for achieving a successful fault detection and isolation [21], because they can be made insensitive to certain input space directions if some structural algebraic conditions on the system are fulfilled [22].

The application of UIOs to the problem of icing detection started with [23], where it was addressed considering the linearized longitudinal model of the vehicle. Further improvements have been obtained using multiple models [24] and linear parameter varying (LPV) methods [25, 26]. LPV techniques have attracted a lot of attention in the last decades, as testified by the increasing number of publications dealing with this

topic [27,28], some of which are related with controller design for aircrafts/UAVs [5]. The LPV paradigm has provided an elegant way to apply linear-like techniques to nonlinear systems, with theoretical guarantees of stability and performance [29]. Unlike linearization techniques, LPV methods do not involve any approximation, since they rely on exact transformation of the original nonlinear system into a quasi-linear one, by embedding all the original nonlinearities within some varying parameters that schedule the state space matrices [30]. As a consequence, an LPV UIO-based icing detection scheme has the advantage of being consistent with a wide range of operating conditions.

The main and novel contribution of this paper is the generalization of the fault and icing diagnosis method introduced in [25], where it was developed for the longitudinal dynamics, to the complete 6-DOF UAV model, which includes the coupled longitudinal/lateral dynamics. Another contribution of this work is the adaptation of the continuous-time formulation of the UIO proposed in [25] to deal with discrete-time models, which makes the methodology more suitable for a practical implementation in the UAV on-board computer. The developed theory is supported by simulations illustrating diagnosis of actuator faults and icing in a small UAV, with results that are thoroughly discussed.

The paper is structured as follows. Section 2 presents the UAV 6-DOF model, and shows how a *quasi-LPV* model can be obtained using a nonlinear embedding in the parameters approach. Following the established terminology, a *quasi-LPV* model has the same structure of a *pure* LPV model, but the varying parameters depend on endogenous signals, such as state and input variables [31]. Also, the modeling of undesired effects, such as wind disturbance, actuator faults, and icing, are presented. Section 3 is dedicated to the design of a discrete-time LPV UIO for performing fault and icing diagnosis in UAVs. In addition, a metric that can be used for analysing the robustness of the LPV UIO is proposed, along with a Monte Carlo strategy for obtaining an estimate with some desired bound on the standard deviation. Simulation results are presented in Section 4 to demonstrate the effectiveness of the proposed approach, and Section 5 outlines the main conclusions.

Notation

The notation is listed in Table 1 using alphabetical order, where the romanized names of Greek letters have been used. For example, α is listed as if it were spelled *alpha*.

2 Model and setup

2.1 Nonlinear model

The UAV nonlinear model consists of an equation for the altitude h , three equations for the airspeed components (u, v, w) , three equations for the Euler angles (ϕ, θ, ψ) , and three equations for the angular rates (p, q, r) [32]:

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta \quad (1)$$

Table 1 Notation

Symbol	Description
A, A_d	continuous-time and discrete-time state matrix
a_p, a_q, a_r	angular acceleration along the body frame x-axis, y-axis, z-axis
a_x, a_y, a_z	linear acceleration along the body frame x-axis, y-axis, z-axis
α	angle of attack
B, B_d	continuous-time and discrete-time input matrix
B_{un}	unknown input matrix
b	wing span
β	side slip angle
C	output matrix
C_i, C_i^*	stability/control derivatives under nominal and under icing conditions
c	mean aerodynamic chord of the wing
χ^{ref}	desired course angle
d, d_d	continuous-time and discrete-time gravitational effect vector
$\delta_a, \delta_e, \delta_r, \delta_t$	control signal corresponding to aileron, elevator, rudder, throttle deflections
e_i	canonical basis vector
$\varepsilon, \varepsilon_i^{th}$	observation error vector and decision algorithm threshold
η	icing severity factor
F	one of the unknown input observer matrices
\mathcal{F}	additive unknown input term due to actuator faults
G, G_d	continuous-time and discrete-time icing input matrix
g	gravitational acceleration (9.81 m/s^2)
I_i	products of the inertia matrix
h, h^{ref}	real and desired altitude
I_n	identity matrix of order n
K_i	icing coefficients
k	sample (discrete-time)
k_m	efficiency of the motor
λ_i^F	eigenvalue of the matrix F
m	mass of the airframe
$O_{m \times n}$	$m \times n$ matrix with all its elements equal to zero
ω	additive disturbance vector
p	roll rate along the body frame x-axis
ϕ	roll angle
φ	effect of faults vector
$\varphi_a, \varphi_e, \varphi_r, \varphi_t$	effect of faults in aileron, elevator, rudder, throttle
ψ	heading angle
q	pitch rate along the body frame y-axis
R	one of the unknown input observer matrices
r	yaw rate along the body frame z-axis
ρ	density of air
S	surface area of the wing
S_{prop}	area of the propeller
$\Sigma, \Sigma_1, \Sigma_2$	some of the unknown input observer matrices
T	one of the unknown input observer matrices
T_s	sampling period
t	time variable (continuous-time)
θ	pitch angle
ϑ	varying parameter vector
u	relative airspeed projected onto the body frame x-axis
v, v_{un}	control and unknown input vector
V_a, V_a^{ref}	real and desired total airspeed (velocity of the airframe w.r.t. air mass)
v	relative airspeed projected onto the body frame y-axis
v_i	sensor measurement noise
v_w	gust component of the wind in the North-East-Down frame
W	rotation matrix from North-East-Down frame to body frame
W_d	discrete-time rotation matrix of the UAV
\mathcal{W}	additive wind disturbance term
w	relative airspeed projected onto the body frame z-axis
x	state vector
ξ_i, ξ_i^{th}	low-pass filtered residual and low-pass filtered decision algorithm threshold
\hat{x}	observed state vector
z	unknown input observer state vector

$$\dot{u} = rv - qw - g \sin \theta + a_x \quad (2)$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + a_y \quad (3)$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + a_z \quad (4)$$

$$\dot{\phi} = p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \quad (5)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (6)$$

$$\dot{\psi} = q \sin \phi \sec \theta + r \cos \phi \sec \theta \quad (7)$$

$$\dot{p} = \Gamma_1 pq - \Gamma_2 qr + a_p \quad (8)$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6 (p^2 - r^2) + a_q \quad (9)$$

$$\dot{r} = \Gamma_7 pq - \Gamma_1 qr + a_r \quad (10)$$

where the linear/angular accelerations $a_x, a_y, a_z, a_p, a_q, a_r$ are given by:

$$a_x = \frac{\rho V_a^2 S}{2m} \left(C_X(\alpha) + C_{X_q}(\alpha) \frac{cq}{2V_a} + C_{X_{\delta_e}}(\alpha) \delta_e \right) + \frac{\rho S_{prop} C_{prop}}{2m} (k_m^2 \delta_i^2 - V_a^2) \quad (11)$$

$$a_y = \frac{\rho V_a^2 S}{2m} \left(C_{Y_0} + C_{Y_\beta} \beta + C_{Y_p} \frac{bp}{2V_a} + C_{Y_r} \frac{br}{2V_a} + C_{Y_{\delta_a}} \delta_a + C_{Y_{\delta_r}} \delta_r \right) \quad (12)$$

$$a_z = \frac{\rho V_a^2 S}{2m} \left(C_Z(\alpha) + C_{Z_q}(\alpha) \frac{cq}{2V_a} + C_{Z_{\delta_e}}(\alpha) \delta_e \right) \quad (13)$$

$$a_p = \frac{1}{2} \rho V_a^2 S b \left(C_{p_0} + C_{p_\beta} \beta + C_{p_p} \frac{bp}{2V_a} + C_{p_r} \frac{br}{2V_a} + C_{p_{\delta_a}} \delta_a + C_{p_{\delta_r}} \delta_r \right) \quad (14)$$

$$a_q = \frac{\rho V_a^2 S c}{2J_y} \left(C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{cq}{2V_a} + C_{m_{\delta_e}} \delta_e \right) \quad (15)$$

$$a_r = \frac{1}{2} \rho V_a^2 S b \left(C_{r_0} + C_{r_\beta} \beta + C_{r_p} \frac{bp}{2V_a} + C_{r_r} \frac{br}{2V_a} + C_{r_{\delta_a}} \delta_a + C_{r_{\delta_r}} \delta_r \right) \quad (16)$$

The nondimensional coefficients C_i are usually referred to as *stability* and *control derivatives*. Some of them are nonlinear functions of:

$$\alpha = \arctan(u/w) \quad (17)$$

according to:

$$C_X(\alpha) = (C_{L_0} + C_{L_\alpha} \alpha) \sin \alpha - (C_{D_0} + C_{D_\alpha} \alpha) \cos \alpha \quad (18)$$

$$C_{X_q}(\alpha) = C_{L_q} \sin \alpha - C_{D_q} \cos \alpha \quad (19)$$

$$C_{X_{\delta_e}}(\alpha) = C_{L_{\delta_e}} \sin \alpha - C_{D_{\delta_e}} \cos \alpha \quad (20)$$

$$C_Z(\alpha) = -[(C_{D_0} + C_{D_\alpha} \alpha) \sin \alpha + (C_{L_0} + C_{L_\alpha} \alpha) \cos \alpha] \quad (21)$$

$$C_{Z_q}(\alpha) = -(C_{D_q} \sin \alpha + C_{L_q} \cos \alpha) \quad (22)$$

$$C_{Z_{\delta_e}}(\alpha) = -(C_{D_{\delta_e}} \sin \alpha + C_{L_{\delta_e}} \cos \alpha) \quad (23)$$

2.2 Quasi-LPV model

The nonlinear model (1)-(10) can be brought to a quasi-LPV form using the nonlinear embedding in the parameters approach [33, 34]:

$$\dot{x}(t) = A(x(t))x(t) + B(x(t))v(t) + d(x(t)) \quad (24)$$

where $x = (h, u, v, w, \phi, \theta, \psi, p, q, r)^T$, $v = (\delta_l^2, \delta_e, \delta_a, \delta_r)^T$, and the matrix functions $A(x(t))$, $B(x(t))$, $d(x(t))$ are given by:

$$A(x(t)) = \begin{pmatrix} 0 & a_{12}(\cdot) & a_{13}(\cdot) & a_{14}(\cdot) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{22}(\cdot) & a_{23}(\cdot) & a_{24}(\cdot) & 0 & 0 & 0 & 0 & a_{29}(\cdot) & a_{210}(\cdot) \\ 0 & a_{32}(\cdot) & a_{33}(\cdot) & a_{34}(\cdot) & 0 & 0 & 0 & a_{38}(\cdot) & 0 & a_{310}(\cdot) \\ 0 & a_{42}(\cdot) & a_{43}(\cdot) & a_{44}(\cdot) & 0 & 0 & 0 & a_{48}(\cdot) & a_{49}(\cdot) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & a_{59}(\cdot) & a_{510}(\cdot) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{69}(\cdot) & a_{610}(\cdot) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{79}(\cdot) & a_{710}(\cdot) \\ 0 & a_{82}(\cdot) & a_{83}(\cdot) & a_{84}(\cdot) & 0 & 0 & 0 & a_{88}(\cdot) & a_{89}(\cdot) & a_{810}(\cdot) \\ 0 & a_{92}(\cdot) & a_{93}(\cdot) & a_{94}(\cdot) & 0 & 0 & 0 & a_{98}(\cdot) & a_{99}(\cdot) & a_{910}(\cdot) \\ 0 & a_{102}(\cdot) & a_{103}(\cdot) & a_{104}(\cdot) & 0 & 0 & 0 & a_{108}(\cdot) & a_{109}(\cdot) & a_{1010}(\cdot) \end{pmatrix} \quad (25)$$

$$B(x(t)) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ b_{21} & b_{22}(\cdot) & 0 & 0 \\ 0 & 0 & b_{33}(\cdot) & b_{34}(\cdot) \\ 0 & b_{42}(\cdot) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_{83}(\cdot) & b_{84}(\cdot) \\ 0 & b_{92}(\cdot) & 0 & 0 \\ 0 & 0 & b_{103}(\cdot) & b_{104}(\cdot) \end{pmatrix} \quad (26)$$

$$d(x(t)) = \begin{pmatrix} 0 \\ -g \sin \theta \\ g \cos \theta \sin \phi \\ g \cos \theta \cos \phi \\ \mathcal{O}_{6 \times 1} \end{pmatrix} \quad (27)$$

The expression of the coefficients appearing in the matrices are detailed in the Appendix.

2.3 Wind disturbance

The system dynamics is affected by the wind velocity, which can be expressed by the additional input:

$$\begin{pmatrix} \mathcal{W} \\ \mathcal{O}_{6 \times 1} \end{pmatrix}, \quad \mathcal{W} = -W(x(t))\dot{v}_w(t) \quad (28)$$

where $v_w = (v_N, v_E, v_D)^T$ is the gust component of the wind speed expressed in the North-East-Down (NED) coordinate frame and W is the rotation matrix from NED to body:

$$W(x(t)) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In order to model the wind gusts in a realistic way, the widely accepted Dryden wind turbulence model, also known as Dryden gusts, is used [35]. The Dryden model uses spatially varying stochastic processes to represent the components of the gusts, specifying their power spectral density.

2.4 Measured outputs

The UAV is assumed to be equipped with a sensor suite which includes a pitot static tube aligned with the longitudinal body axis, a GPS, an altimeter, gyroscopes and accelerometers. In addition, it is assumed that an accurate wind speed estimator is available [36], such that the interpolation of the estimated wind speed with the average UAV speed, that can be computed through the GPS data, provides also a measurement of the velocities v, w . Hence, the output matrix of the system verifies $C = I_{10}$.

2.5 Actuator faults

Actuator faults may affect the system and can be represented as an additive unknown input term $\mathcal{F}(t)$ given by:

$$\mathcal{F}(t) = B(x(t)) \begin{pmatrix} \varphi_t(t) \\ \varphi_e(t) \\ \varphi_a(t) \\ \varphi_r(t) \end{pmatrix} = B(x(t)) \varphi(t) \quad (29)$$

where $\varphi_t, \varphi_e, \varphi_a, \varphi_r$ represent the effects of faults in propellers/engines, elevator, aileron and rudder, respectively.

2.6 Icing effect model

The accretion of ice on the UAV surfaces modifies the stability and control derivatives according to the following linear model [37]:

$$C_i^* = (1 + \eta K_i) \quad (30)$$

where η denotes the icing severity factor and the coefficients K_i depend on the UAV design, the atmospheric conditions and the icing location. The clean condition corresponds to $\eta = 0$, while the worst icing condition occurs for $\eta = 0.2$.

As a consequence, the overall effect of icing can be regarded as an additive input described by:

$$G(x(t))\eta(t) \quad (31)$$

with:

$$G(x(t)) = (0 \ g_2(\cdot) \ g_3(\cdot) \ g_4(\cdot) \ 0 \ 0 \ 0 \ g_8(\cdot) \ g_9(\cdot) \ g_{10}(\cdot))^T \quad (32)$$

The expressions of the coefficients g_i are detailed in the Appendix.

Remark 1: As a matter of fact, icing may also affect the airspeed measurements, since the pitot tube may be clogged by the ice [38, 39]. A blocked pitot tube will cause typically the airspeed indicator to register an increase in airspeed when the UAV climbs, even though actual airspeed is constant. This is caused by the pressure in the pitot system remaining constant when the atmospheric pressure is decreasing. In reverse, the airspeed indicator will show a decrease in airspeed when the UAV descends. However, many pitot tubes are equipped with a heating element, which is required in all UAVs certified for instrument flight, such that a straightforward accommodation of icing effects on sensors can be obtained.

3 Actuator faults and icing diagnosis

This section describes the LPV UIO-based strategy for performing fault and icing diagnosis in UAVs. The overall conceptual scheme of the proposed approach is given in Fig. 1. The LPV UIO uses the state measurements coming from the UAV's sensors and the control input, along with the UAV's model, to compute an estimate \hat{x} , which is later compared to the measurements. The observation error ε is fed to the decision algorithm, which decides about the occurrence of faults/icing.

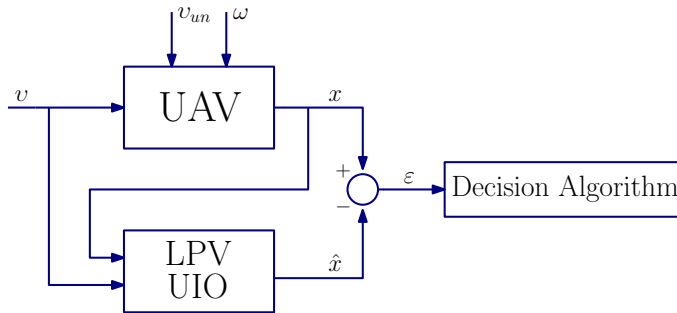


Fig. 1 LPV UIO-based fault and icing diagnoser for UAVs.

3.1 Discrete-time LPV unknown input observer

Taking into account the wind, the actuator faults and the icing, the quasi-LPV model (24) is slightly modified, as follows:

$$\begin{aligned} \dot{x}(t) = & A(x(t))x(t) + B(x(t))v(t) + d(x(t)) \\ & - \begin{pmatrix} O_{1 \times 3} \\ W(x(t)) \\ O_{6 \times 3} \end{pmatrix} \dot{v}_w(t) + B(x(t))\varphi(t) + G(x(t))\eta(t) \end{aligned} \quad (33)$$

which can be rewritten in a more general form for further reasoning:

$$\dot{x}(t) = A(\vartheta(t))x(t) + B(\vartheta(t))v(t) + B_{um}(\vartheta(t))v_{um}(t) + d(\vartheta(t)) + \omega(t) \quad (34)$$

where ϑ is some varying parameter vector, containing exogenous variables, endogenous variables (e.g. states and/or inputs), or a combination of them, $v_{um}(t)$ and $\omega(t)$ are unknown vectors containing the fault effects $\varphi(t)$ and the wind/icing effects given by terms $\dot{v}_w(t)$, $\eta(t)$, respectively, and $B_{um}(\vartheta(t))$ is the matrix that describes how $v_{um}(t)$ affects the system's state. The continuous-time LPV system (34) can be brought to the discrete-time form:

$$x(k+1) = A(\vartheta(k))x(k) + B(\vartheta(k))v(k) + B_{um}(\vartheta(k))v_{um}(k) + d(\vartheta(k)) + \omega(k) \quad (35)$$

using discretization methods, such as Euler, Runge-Kutta or other approaches [40].

In the following, the continuous-time LPV unknown input observer introduced in [25] is adapted in order to cope with the discrete-time formulation (35). As remarked by [21], the main advantage of such observers is that, if some structural conditions are met, the parameters can be designed such that the resulting observation error is independent of some inputs of the system, even if these are not measured directly.

More specifically, let $R(\vartheta(k))$ and $F(\vartheta(k))$ be some given matrix functions, calculate the following matrix functions:

$$T(\vartheta(k)) = I - R(\vartheta(k)) \quad (36)$$

$$\Sigma_1(\vartheta(k)) = R(\vartheta(k))A(\vartheta(k)) - F(\vartheta(k)) \quad (37)$$

$$\Sigma_2(\vartheta(k-1), \vartheta(k)) = F(\vartheta(k))T(\vartheta(k-1)) \quad (38)$$

$$\Sigma(\vartheta(k)) = \Sigma_1(\vartheta(k)) + \Sigma_2(\vartheta(k-1), \vartheta(k)) \quad (39)$$

and let the unknown input observer for the system (34) be given by:

$$\begin{aligned} z(k+1) = & F(\vartheta(k))z(k) + R(\vartheta(k))B(\vartheta(k))v(k) \\ & + \Sigma(\vartheta(k-1), \vartheta(k))x(k) + d(\vartheta(k)) - T(\vartheta(k))d(\vartheta(k)) \end{aligned} \quad (40)$$

$$\hat{x}(k) = z(k) + T(\vartheta(k-1))x(k) \quad (41)$$

Then, the observation error $\epsilon(k) = x(k) - \hat{x}(k)$ is given by:

$$\begin{aligned} \epsilon(k+1) = & [A(\vartheta(k)) - T(\vartheta(k))A(\vartheta(k)) - \Sigma(\vartheta(k-1), \vartheta(k))]x(k) \\ & - F(\vartheta(k))z(k) + [I - R(\vartheta(k)) - T(\vartheta(k))]B(\vartheta(k))v(k) \\ & + [I - T(\vartheta(k))] [B_{um}(\vartheta(k))v_{um}(k) + \omega(k)] \end{aligned} \quad (42)$$

which, taking into account (36), can be rewritten as:

$$\begin{aligned} \varepsilon(k+1) = & [R(\vartheta(k))A(\vartheta(k)) - \Sigma(\vartheta(k-1), \vartheta(k))]x(k) \\ & - F(\vartheta(k))z(k) + R(\vartheta(k))B_{un}(\vartheta(k))\mathbf{v}_{un}(k) + R(\vartheta(k))\omega(k) \end{aligned} \quad (43)$$

Then, combining (37)-(38) with (39), it can be shown that (43) is equivalent to:

$$\begin{aligned} \varepsilon(k+1) = & [F(\vartheta(k)) - F(\vartheta(k))T(\vartheta(k-1))]x(k) \\ & - F(\vartheta(k))z(k) + R(\vartheta(k))B_{un}(\vartheta(k))\mathbf{v}_{un}(k) + R(\vartheta(k))\omega(k) \end{aligned} \quad (44)$$

It is easy to see that the following comes from (44), after taking into account (41):

$$\varepsilon(k+1) = F(\vartheta(k))\varepsilon(k) + R(\vartheta(k))B_{un}(\vartheta(k))\mathbf{v}_{un}(k) + R(\vartheta(k))\omega(k) \quad (45)$$

Notice that $F(\vartheta(k))$ can be chosen as a constant matrix F through an appropriate choice of the matrix function $\Sigma_1(\vartheta(k))$, which allows assigning some desired eigenvalues of F and assure convergence of the observation error ε when $\mathbf{v}_{un} = 0$ and $\omega = 0$. On the other hand, the matrix function $R(\vartheta(k))$ can be chosen to constrain the range of $R(\vartheta(k))B_{un}(\vartheta(k))$, in such a way that different output directions of the residuals are assigned for the unknown inputs acting on the system, with the aim of identifying the cause for some detected system malfunctions.

3.2 Application to the UAV fault/icing diagnosis

Due to the superposition of effects and the lack of degrees of freedom in the unknown input observer design, it is not possible to decouple completely the wind disturbance and icing effects from the actuator faults. However, it is still possible to design the unknown input observer matrices such that a successful fault/icing diagnosis can be achieved.

Let us consider the discrete-time model of the UAV obtained from (33) using the Euler approach with sampling period T_s :

$$\begin{aligned} x(k+1) = & A_d(x(k))x(k) + B_d(x(k))\mathbf{v}(k) + d_d(x(k)) \\ & - \begin{pmatrix} O_{1 \times 3} \\ W_d(x(k)) \\ O_{6 \times 3} \end{pmatrix} \dot{\mathbf{v}}_w(k) + B_d(x(k))\varphi(k) + G_d(x(k))\eta(k) \end{aligned} \quad (46)$$

with $A_d(x(k)) = I + T_s A(x(k))$, $B_d(x(k)) = T_s B(x(k))$, $W_d(x(k)) = T_s W(x(k))$, $G_d(x(k)) = T_s G(x(k))$ and $d_d(x(k)) = T_s d(x(k))$, and let us denote by e_1, \dots, e_{10} the canonical basis vectors of \mathcal{R}^{10} . Then, the following condition holds:

$$G(x(k))\eta(k) \in \text{span} [e_3 \ e_4 \ B_d(x(k))] \quad \forall k \geq 0 \quad (47)$$

which allows defining our target as designing the unknown input observer matrices with the following properties:

$$R(x(k)) [e_3 \ e_4 \ B_d(x(k)) \ e_1 \ e_5 \ e_6 \ e_7] = I_{10} \quad (48)$$

$$F e_i = \lambda_i^F e_i \quad \forall i = 1, \dots, 10 \quad (49)$$

where $\lambda_i^F, i = 1, \dots, 10$ are the desired eigenvalues of the matrix F (λ_i^F inside the unit circle, in order to assure the stability of the observer).

It is easy to check that the matrix satisfying (48)-(49) has the following structure:

$$R(x(k)) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & r_{18}(\cdot) & 0 & r_{110}(\cdot) \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & r_{29}(\cdot) & 0 \\ 0 & -1/b_{21}T_s & 0 & 0 & 0 & 0 & 0 & 0 & r_{39}(\cdot) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/b_{92}T_s & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & r_{58}(\cdot) & 0 & r_{510}(\cdot) \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & r_{68}(\cdot) & 0 & r_{610}(\cdot) \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (50)$$

where the elements $r_{18}(\cdot), r_{110}(\cdot), r_{29}(\cdot), r_{39}(\cdot), r_{58}(\cdot), r_{510}(\cdot), r_{68}(\cdot), r_{610}(\cdot)$ are functions of the elements of the matrix $B_d(x(k))$.

Under the assumption that at a given instant either a single fault or icing could act on the system (no simultaneous multiple faults and icing), (48)-(49) lead to the following algorithm for deciding about the occurrence of faults/icing.

Decision Algorithm.

```

if { |ε1(k)| ≤ ε1th, |ε2(k)| ≤ ε2th, |ε3(k)| ≤ ε3th
    { |ε4(k)| ≤ ε4th, |ε5(k)| ≤ ε5th, |ε6(k)| ≤ ε6th
    then "no faults/no icing"
if { |ε1(k)| ≤ ε1th, |ε2(k)| ≤ ε2th, |ε3(k)| > ε3th
    { |ε4(k)| ≤ ε4th, |ε5(k)| ≤ ε5th, |ε6(k)| ≤ ε6th
    then "fault in thrust"
if { |ε1(k)| ≤ ε1th, |ε2(k)| ≤ ε2th, |ε3(k)| ≤ ε3th
    { |ε4(k)| > ε4th, |ε5(k)| ≤ ε5th, |ε6(k)| ≤ ε6th
    then "fault in elevator"
if { |ε1(k)| ≤ ε1th, |ε2(k)| ≤ ε2th, |ε3(k)| ≤ ε3th
    { |ε4(k)| ≤ ε4th, |ε5(k)| > ε5th, |ε6(k)| ≤ ε6th
    then "fault in ailerons"
if { |ε1(k)| ≤ ε1th, |ε2(k)| ≤ ε2th, |ε3(k)| ≤ ε3th
    { |ε4(k)| ≤ ε4th, |ε5(k)| ≤ ε5th, |ε6(k)| > ε6th
    then "fault in rudder"
else "icing"

```

The thresholds $\epsilon_i^{th}, i = 1, \dots, 6$ should be calculated in such a way that the residuals never exceed them due to the wind turbulence. It is worth noticing that:

$$R(x(k)) \begin{pmatrix} O_{1 \times 3} \\ W_d(x(k)) \\ O_{6 \times 3} \end{pmatrix} = \begin{pmatrix} W_R(x(k)) \\ O_{7 \times 3} \end{pmatrix}$$

for some $W_R(x(k))$, which means that (at least in theory) the wind turbulence should affect only the residuals $\epsilon_1(k), \epsilon_2(k), \epsilon_3(k)$. However, due to undesired effects such as

discretization errors, presence of sensor noise and parametric uncertainties, in practice the wind turbulence would affect the other residuals as well (although at a much smaller extent than the first three residuals).

3.3 Robustness analysis

Hereafter, a metric based on the idea of stochastic robustness [41, 42] is proposed in order to analyse the robustness of the LPV UIO-based faults and icing diagnoser. This metric is given by the probability that the LPV UIO will exhibit an unacceptable behaviour. More specifically, let us denote the LPV UIO as \mathcal{O} , while the set of possible scenarios is denoted by $\mathcal{S}(\mu)$, where $\mu \in \mathbb{M}$ denotes possible variations due to model uncertainty, noise, etc. within a bounded set \mathbb{M} , which can be described by a probability density function $pr(\mu)$. Then, the performance metric can be defined as the integral of an indicator function over the space of expected variations:

$$\Psi(\mathcal{O}) = \int_{\mathbb{M}} I[\mathcal{S}(\mu), \mathcal{O}] pr(\mu) d\mu \quad (51)$$

where I is a binary function which describes if the behaviour of the fault/icing diagnoser for a given realization of μ is acceptable ($I = 1$) or not ($I = 0$).

Unfortunately, (51) cannot be integrated analytically. A practical alternative is to use Monte Carlo methods [43] with $pr(\mu)$ shaping random values of μ that will be denoted by μ_i . When M random μ_i , $i = 1, \dots, M$ are generated, then an estimate of Ψ is given by:

$$\hat{\Psi}(\mathcal{O}) = \frac{1}{M} \sum_{i=1}^M I[\mathcal{S}(\mu_i), \mathcal{O}] \quad (52)$$

where $\hat{\Psi}$ approaches Ψ in the limit as $M \rightarrow \infty$. However, it is impossible to set $M = \infty$, thus it is interesting to choose M in such a way that $\hat{\Psi}$ has standard deviation less than a desired value $\sigma_{\hat{\Psi}}$. Since I is binary, $\hat{\Psi}$ has a binomial distribution, such that M can be chosen as [41]:

$$M \geq \left\lceil \frac{1}{4\sigma_{\hat{\Psi}}^2} \right\rceil \quad (53)$$

4 Results

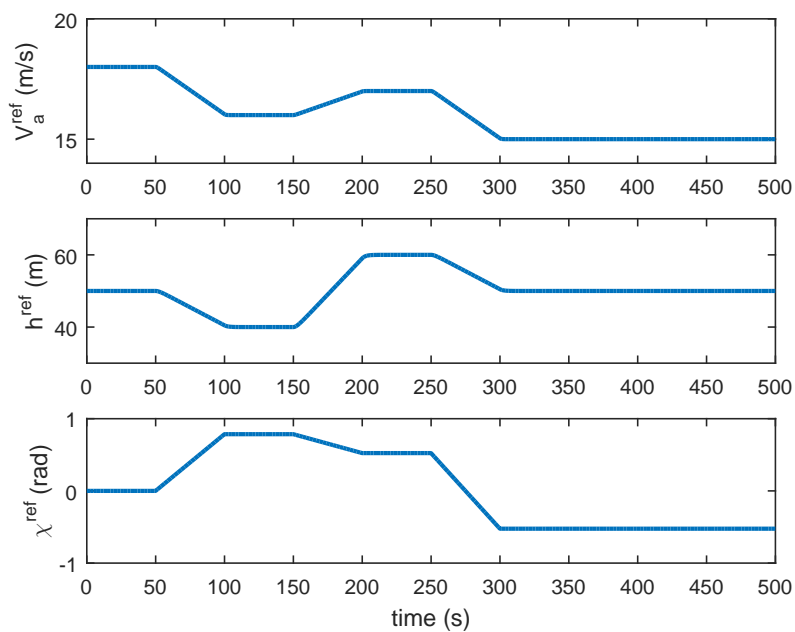
In this section, the proposed actuator faults and icing diagnosis through discrete-time LPV unknown input observer is validated through the application to the case study of a typical small UAV, which is supposed to encounter icing conditions. The parameters appearing in the nonlinear model detailed in Section 2.1 are provided in Table 2 [32].

The UAV is controlled by an autopilot, working at a sampling period of $T_s = 0.05$ s, which is responsible for tracking some desired total airspeed V_a^{ref} , altitude h^{ref} and course angle χ^{ref} . In particular, the desired trajectory depicted in Fig. 2 is considered for the subsequent simulations.

Four different scenarios have been analysed, namely:

Table 2 System parameters values

Param.	Value	Param.	Value	Param.	Value
m	13.5 kg	C_{L_0}	0.28	C_{Y_0}	0
J_x	0.8244 kg m ²	C_{D_0}	0.03	C_{l_0}	0
J_y	1.135 kg m ²	C_{m_0}	-0.02338	C_{n_0}	0
J_z	1.759 kg m ²	C_{L_α}	3.45	C_{Y_β}	-0.98
J_{xz}	0.1204 kg m ²	C_{D_α}	0.30	C_{l_β}	-0.12
S	0.55 m ²	C_{m_α}	-0.38	C_{n_β}	0.25
b	2.8956 m	C_{L_q}	0	C_{Y_p}	0
c	0.18994 m	C_{D_q}	0	C_{l_p}	-0.26
S_{prop}	0.2027 m ²	C_{m_q}	-3.6	C_{n_p}	0.022
ρ	1.2682 kg/m ³	$C_{L_{\delta_e}}$	-0.36	C_{Y_r}	0
k_m	80	$C_{D_{\delta_e}}$	0	C_{l_r}	0.14
C_{prop}	1.0	$C_{m_{\delta_e}}$	-0.5	C_{n_r}	-0.35
M	50	$C_{Y_{\delta_r}}$	-0.17	$C_{Y_{\delta_a}}$	0
		$C_{l_{\delta_r}}$	0.105	$C_{l_{\delta_a}}$	0.08
		$C_{n_{\delta_r}}$	-0.032	$C_{n_{\delta_a}}$	0.06

**Fig. 2** Desired trajectories V_a^{ref} , h^{ref} and χ^{ref} .

- 1) Fault in thrust
- 2) Fault in elevator
- 3) Fault in ailerons
- 4) Icing

Table 3 Coefficients K_i for an all iced configuration

Coeff.	Value	Coeff.	Value	Coeff.	Value
K_{L_0}	0	K_{L_α}	-0.5000	K_{L_q}	-0.0675
$K_{L_{\delta_e}}$	-0.4770	K_{D_0}	2.5610	K_{D_α}	0
K_{D_q}	0	$K_{D_{\delta_e}}$	0	K_{m_0}	0
K_{m_α}	-0.4960	K_{m_q}	-0.1755	$K_{m_{\delta_e}}$	-0.5000
K_{Y_β}	-1	K_{Y_p}	0	K_{Y_r}	0
$K_{Y_{\delta_r}}$	-0.4	K_{l_β}	-0.5	K_{l_p}	-0.5
K_{l_r}	0	$K_{l_{\delta_a}}$	-0.5	$K_{l_{\delta_r}}$	-0.4
K_{n_β}	-1	K_{n_p}	0	K_{n_r}	-0.3056
$K_{n_{\delta_r}}$	-0.4167	$K_{n_{\delta_a}}$	0		

Fault scenario 1 (FS1)

The effective thrust input is subject to a loss of efficiency with respect to its nominal values δ_i^* :

$$\delta_i(t) = 0.7\delta_i^*(t)$$

The fault is considered to be abrupt, starting from time $t = 135.s$.

Fault scenario 2 (FS2)

The effective elevator deflection δ_e is subject to a loss of efficiency with respect to its nominal value δ_e^* :

$$\delta_e(t) = 0.9\delta_e^*(t)$$

The fault is considered to be abrupt, starting from time $t = 135.s$.

Fault scenario 3 (FS3)

The effective aileron deflection δ_a is subject to a loss of efficiency with respect to its nominal value δ_a^* :

$$\delta_a(t) = 0.6\delta_a^*(t)$$

The fault is considered to be abrupt, starting from time $t = 135.s$.

Fault scenario 4 (FS4)

The UAV is subject to icing, i.e. the stability and control derivatives are modified according to (30), taking into account the coefficients K_i listed in Table 3. The icing starts at time $t = 135.s$, and slowly increases from $\eta = 0$ to $\eta = 0.2$, such that $\eta = 0.2$ starting from time $t = 235.s$.

Remark 2: The coefficients K_i used in this work have been computed mimicking the proportional variation of the stability and control derivatives for a Twin Otter aircraft subject to all iced conditions [37], and they could differ in the case of a real Aerosonde UAV. However, since the proposed LPV unknown input observer does not depend on the values of these coefficients, which are included in the simulator

only to provide realistic icing effects, it can be expected that similar results would be obtained with different values of the coefficients K_i .

4.1 Results without sensor noise/uncertainties

Hereafter, the results related to the application of the UIO-based fault/icing diagnosis strategy described in Section 3 are presented. The proposed UIO, with a choice of $F = 0.9I_{10}$, is tested against the nonlinear continuous-time UAV simulator, which is affected by the wind turbulence acting as an exogenous undesired unknown effect. Also, in order to show the advantages of using an LPV framework when dealing with highly nonlinear systems, the results obtained with the proposed LPV UIO (plotted as blue solid lines) are compared with the ones obtained using a linear time invariant (LTI) UIO (plotted as green dotted lines). The LTI UIO has been tuned on the LTI model of the UAV working at the operating point given by $V_a = 18\text{ m/s}$, $h = 50\text{ m}$, $\chi = 0\text{ rad}$, which corresponds to the desired operating point at the beginning of the simulations, i.e. from $t = 0\text{ s}$ to $t = 50\text{ s}$ (see Fig. 2).

It has been noticed that the frequency content of the residuals $\varepsilon_i(t)$ changes under faults/icing occurrence. In fact, in absence of faults/icing, the residuals are excited only by the wind acceleration, which is mostly a high frequency disturbance. On the other hand, the actuator faults and icing effects increase the low frequency content of the variables $\varepsilon_i(t)$. Thus, low-pass filtering the residuals indicates more clearly the presence of faults/icing. More specifically, the low-pass filtered residuals are obtained as follows:

$$\dot{\xi}_i(t) = -a_i\xi_i(t) + a_i\varepsilon_i(t) \quad i = 1, \dots, 10 \quad (54)$$

where a_i is a design parameter that should be chosen to guarantee that only the part of the frequency spectrum of $\varepsilon_i(t)$ affected by faults/icing is presented (hereafter, a value $a_i = 0.995$ is used). Then, the decision algorithm proposed in the previous section is slightly modified by replacing all instances of ε_i with ξ_i and all instances of ε_i^{th} with ξ_i^{th} , $i = 1, \dots, 10$.

Using a simulation scenario without faults/icing acting on the UAV, the values for the thresholds ξ_i^{th} , $i = 1, \dots, 6$ for the LPV UIO have been calculated as follows: $\xi_1^{th} = \xi_2^{th} = 0.06$, $\xi_3^{th} = 0.02$, $\xi_4^{th} = 0.002$ and $\xi_5^{th} = \xi_6^{th} = 0.001$ (red solid horizontal lines in the plots).

Figure 3 shows the evolution of the residuals ξ_i in fault scenario 1. The LPV UIO behaves as expected, since all the residuals are within the calculated thresholds, except for ξ_3 , which exceeds its threshold starting from time $t = 135.65\text{ s}$. According to the Decision Algorithm presented in the previous section, this allows determining correctly that a fault is acting on the thrust. On the other hand, note that the LTI UIO's residuals are sensitive to operating point changes and exogenous disturbances, such as the wind gusts, and hence they cannot be used effectively for performing the diagnosis of faults and icing.

Figure 4 shows the residuals ξ_i obtained in fault scenario 2. The presence of a fault in the elevator is clearly identified due to the response of the residual ξ_4 , which exceeds the threshold ξ_4^{th} (red line) starting from time $t = 135.2\text{ s}$.

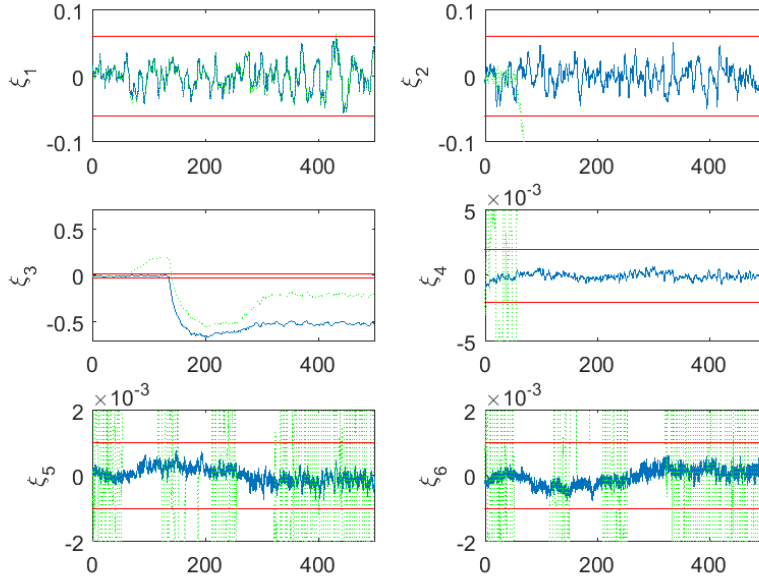


Fig. 3 Residuals $\xi_i(t)$, $i = 1, \dots, 6$, in fault scenario 1 (without sensor noise/uncertainties).

On the other hand, the response of the residual ξ_5 allows isolating a fault in the aileron at time $t = 135.65s$, as shown by Fig. 5, which presents the residuals' responses in fault scenario 3. It is worth noticing that the effect of φ_a on ξ_5 is weaker than the one of φ_e on ξ_4 . This is due to the operation conditions of the UAV, which works with a δ_a close to zero value, such that a loss of effectiveness of this actuator has less severe effects on the system (see Fig. 6, where both δ_a and δ_e for the nominal scenario are shown). In case of doubt about the actual presence of a fault in the aileron, the control system could force the UAV to perform manoeuvres requiring different values of δ_a , thus improving the effectiveness of the fault diagnosis algorithm.

Finally, Fig. 7 presents the residuals ξ_i obtained in fault scenario 4 (icing). In this case, all the residuals exceed the respective thresholds, which according to the Decision Algorithm presented in the previous section allows inferring correctly the presence of ice on the UAV.

4.2 Results with sensor noise/uncertainties

To conclude the analysis of the proposed method, let us assess its robustness against undesired effects, such as sensor noise and uncertainties. To this end, the approach described in Section 3.3 has been applied to choose appropriate thresholds ξ_i^{th} for the residuals. The Monte Carlo simulations used for calculating $\hat{\Psi}$, as in (52), con-

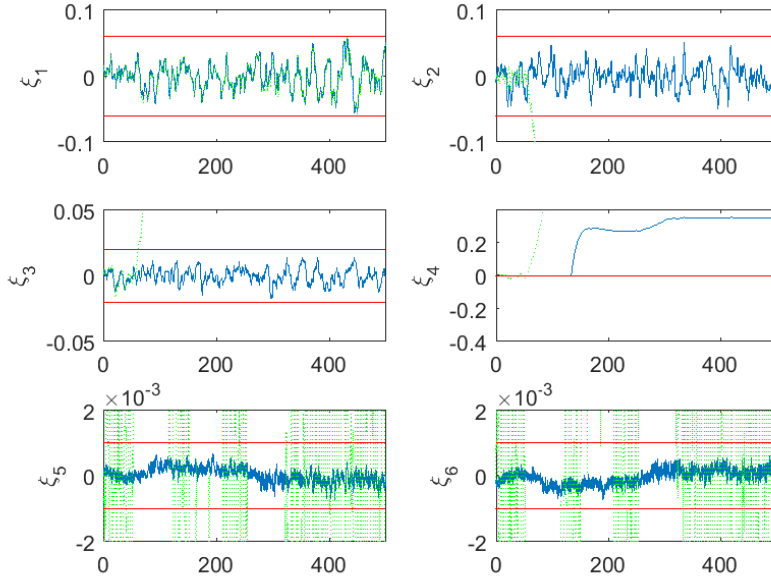


Fig. 4 Residuals $\xi_i(t)$, $i = 1, \dots, 6$, in fault scenario 2 (without sensor noise/uncertainties).

sider that all the stability/control derivatives are affected by normally distributed uncertainty with standard deviation equal to 3% of their nominal values, and that the noise affecting the sensor measurements is normally distributed around zero mean value, with the following standard deviations: $\sigma_1 = 0.25 m$, $\sigma_2 = \sigma_3 = \sigma_4 = 0.25 m/s$, $\sigma_5 = \sigma_6 = \sigma_7 = 0.002 rad$, $\sigma_8 = \sigma_9 = \sigma_{10} = 0.0005 rad/s$. For each residual ξ_i , $i = 1, \dots, 6$, the metric (52) has been calculated for different thresholds by using an indicator function I that takes into account whether the behaviour of the considered residual is compatible with a given fault scenario or not. For example, in the case of the residual ξ_3 , $I = 1$ in case the residual *does not* exceed its thresholds in scenarios with no faults/icing, fault in elevator, fault in aileron and fault in rudder, or in case the residual *does* exceed its thresholds in scenarios with faults in thrust or icing (see Decision Algorithm in Section 3.2), while $I = 0$ otherwise. Notice that by performing Monte Carlo simulations for different values of fault magnitudes, it is possible to select appropriately the thresholds, taking into account the existence of a trade-off between avoiding false alarms and making the LPV UIO-based scheme sensitive to faults/icing.

For each considered scenario, $M = 100$ Monte Carlo simulations have been performed which, given (53), corresponds to ensuring a standard deviation $\sigma_{\hat{\varphi}} = 0.05$. For each residual, a table has been obtained, providing the estimated metric for different scenarios and thresholds (see Tables 4-9), which is useful for choosing appropriate threshold values. For example, if a 95% success rate in avoiding false alarms is

Table 4 Estimated metrics $\hat{\Psi}$ for residual ξ_1 .

	$\xi_1^{th} = 0.10$	$\xi_1^{th} = 0.15$	$\xi_1^{th} = 0.20$	$\xi_1^{th} = 0.25$
No faults	0.77	0.90	0.97	0.99
$\delta_i(t) = 0.6\delta_i^*(t)$	0.70	0.91	1	1
$\delta_i(t) = 0.7\delta_i^*(t)$	0.76	0.95	0.99	1
$\delta_i(t) = 0.8\delta_i^*(t)$	0.84	0.95	0.99	1
$\delta_i(t) = 0.9\delta_i^*(t)$	0.74	0.92	0.97	0.99
$\delta_e(t) = 0.75\delta_e^*(t)$	0.68	0.95	0.99	1
$\delta_e(t) = 0.8\delta_e^*(t)$	0.69	0.88	0.97	1.0000
$\delta_e(t) = 0.85\delta_e^*(t)$	0.65	0.88	0.95	0.99
$\delta_e(t) = 0.9\delta_e^*(t)$	0.76	0.89	0.97	0.99
$\delta_a(t) = 0.5\delta_a^*(t)$	0.77	0.92	0.99	1
$\delta_a(t) = 0.6\delta_a^*(t)$	0.74	0.93	0.99	1
$\delta_a(t) = 0.7\delta_a^*(t)$	0.72	0.93	0.98	1
$\delta_a(t) = 0.8\delta_a^*(t)$	0.74	0.88	0.97	1
Icing, $\eta = 0.2$	0.25	0.09	0.03	0

Table 5 Estimated metrics $\hat{\Psi}$ for residual ξ_2 .

	$\xi_2^{th} = 0.20$	$\xi_2^{th} = 0.25$	$\xi_2^{th} = 0.30$	$\xi_2^{th} = 0.35$
No faults	0.80	0.92	0.98	0.99
$\delta_i(t) = 0.6\delta_i^*(t)$	0.89	0.95	0.97	1
$\delta_i(t) = 0.7\delta_i^*(t)$	0.85	0.96	0.98	1
$\delta_i(t) = 0.8\delta_i^*(t)$	0.81	0.92	0.97	0.98
$\delta_i(t) = 0.9\delta_i^*(t)$	0.88	0.96	0.99	1
$\delta_e(t) = 0.75\delta_e^*(t)$	0.67	0.91	0.99	1
$\delta_e(t) = 0.8\delta_e^*(t)$	0.73	0.90	0.99	1
$\delta_e(t) = 0.85\delta_e^*(t)$	0.85	0.98	1	1
$\delta_e(t) = 0.9\delta_e^*(t)$	0.74	0.86	0.93	1
$\delta_a(t) = 0.5\delta_a^*(t)$	0.70	0.88	0.97	0.99
$\delta_a(t) = 0.6\delta_a^*(t)$	0.81	0.92	0.98	0.99
$\delta_a(t) = 0.7\delta_a^*(t)$	0.82	0.94	1	1
$\delta_a(t) = 0.8\delta_a^*(t)$	0.92	0.99	1	1
Icing, $\eta = 0.2$	1	0.98	0.91	0.81

Table 6 Estimated metrics $\hat{\Psi}$ for residual ξ_3 .

	$\xi_3^{th} = 0.20$	$\xi_3^{th} = 0.25$	$\xi_3^{th} = 0.30$	$\xi_3^{th} = 0.35$
No faults	1	1	1	1
$\delta_i(t) = 0.6\delta_i^*(t)$	1	1	1	1
$\delta_i(t) = 0.7\delta_i^*(t)$	1	1	1	1
$\delta_i(t) = 0.8\delta_i^*(t)$	1	1	1	0.74
$\delta_i(t) = 0.9\delta_i^*(t)$	0	0	0	0
$\delta_e(t) = 0.75\delta_e^*(t)$	0.61	1	1	1
$\delta_e(t) = 0.8\delta_e^*(t)$	0.74	1	1	1
$\delta_e(t) = 0.85\delta_e^*(t)$	0.93	1	1	1
$\delta_e(t) = 0.9\delta_e^*(t)$	0.92	1	1	1
$\delta_a(t) = 0.5\delta_a^*(t)$	0.98	1	1	1
$\delta_a(t) = 0.6\delta_a^*(t)$	0.98	1	1	1
$\delta_a(t) = 0.7\delta_a^*(t)$	1	1	1	1
$\delta_a(t) = 0.8\delta_a^*(t)$	1	1	1	1
Icing, $\eta = 0.2$	0.80	0.77	0.01	0

Table 7 Estimated metrics $\hat{\Psi}$ for residual ξ_4 .

	$\xi_4^{th} = 0.25$	$\xi_4^{th} = 0.30$	$\xi_4^{th} = 0.35$	$\xi_4^{th} = 0.40$
No faults	0.92	0.95	0.98	0.99
$\delta_r(t) = 0.6\delta_r^*(t)$	0.95	0.98	1	1
$\delta_r(t) = 0.7\delta_r^*(t)$	0.93	0.99	1	1
$\delta_r(t) = 0.8\delta_r^*(t)$	0.90	0.93	0.97	1
$\delta_r(t) = 0.9\delta_r^*(t)$	0.96	1	1	1
$\delta_e(t) = 0.75\delta_e^*(t)$	1	1	1	1
$\delta_e(t) = 0.8\delta_e^*(t)$	1	1	1	1
$\delta_e(t) = 0.85\delta_e^*(t)$	1	0.96	0.92	0.87
$\delta_e(t) = 0.9\delta_e^*(t)$	0.89	0.78	0.63	0.54
$\delta_a(t) = 0.5\delta_a^*(t)$	0.94	1	1	1
$\delta_a(t) = 0.6\delta_a^*(t)$	0.94	0.98	0.99	1
$\delta_a(t) = 0.7\delta_a^*(t)$	0.91	0.98	0.99	1
$\delta_a(t) = 0.8\delta_a^*(t)$	0.95	0.98	1	1
Icing, $\eta = 0.2$	0.36	0.13	0.05	0.02

Table 8 Estimated metrics $\hat{\Psi}$ for residual ξ_5 .

	$\xi_5^{th} = 0.06$	$\xi_5^{th} = 0.07$	$\xi_5^{th} = 0.08$	$\xi_5^{th} = 0.09$
No faults	0.68	0.92	0.99	1
$\delta_r(t) = 0.6\delta_r^*(t)$	0.74	0.94	1	1
$\delta_r(t) = 0.7\delta_r^*(t)$	0.67	0.90	0.99	1
$\delta_r(t) = 0.8\delta_r^*(t)$	0.65	0.94	0.98	1
$\delta_r(t) = 0.9\delta_r^*(t)$	0.68	0.92	0.98	1
$\delta_e(t) = 0.75\delta_e^*(t)$	0.79	0.94	0.99	1
$\delta_e(t) = 0.8\delta_e^*(t)$	0.67	0.88	0.98	1
$\delta_e(t) = 0.85\delta_e^*(t)$	0.61	0.83	0.95	0.98
$\delta_e(t) = 0.9\delta_e^*(t)$	0.74	0.96	0.97	1
$\delta_a(t) = 0.5\delta_a^*(t)$	0.74	0.43	0.23	0.11
$\delta_a(t) = 0.6\delta_a^*(t)$	0.42	0.16	0.03	0.01
$\delta_a(t) = 0.7\delta_a^*(t)$	0.42	0.07	0.01	0
$\delta_a(t) = 0.8\delta_a^*(t)$	0.32	0.07	0.01	0
Icing, $\eta = 0.2$	0.24	0.05	0.01	0

Table 9 Estimated metrics $\hat{\Psi}$ for residual ξ_6 .

	$\xi_6^{th} = 0.07$	$\xi_6^{th} = 0.08$	$\xi_6^{th} = 0.09$	$\xi_6^{th} = 0.10$
No faults	0.58	0.88	0.98	1
$\delta_r(t) = 0.6\delta_r^*(t)$	0.68	0.90	0.99	1
$\delta_r(t) = 0.7\delta_r^*(t)$	0.57	0.85	0.97	1
$\delta_r(t) = 0.8\delta_r^*(t)$	0.60	0.90	0.98	0.99
$\delta_r(t) = 0.9\delta_r^*(t)$	0.60	0.88	0.95	0.99
$\delta_e(t) = 0.75\delta_e^*(t)$	0.75	0.91	0.98	0.99
$\delta_e(t) = 0.8\delta_e^*(t)$	0.61	0.81	0.97	0.99
$\delta_e(t) = 0.85\delta_e^*(t)$	0.51	0.79	0.92	0.97
$\delta_e(t) = 0.9\delta_e^*(t)$	0.68	0.95	0.97	0.98
$\delta_a(t) = 0.5\delta_a^*(t)$	0.55	0.86	0.97	1
$\delta_a(t) = 0.6\delta_a^*(t)$	0.57	0.85	0.95	0.98
$\delta_a(t) = 0.7\delta_a^*(t)$	0.59	0.93	0.99	1
$\delta_a(t) = 0.8\delta_a^*(t)$	0.64	0.91	0.98	1
Icing, $\eta = 0.2$	0.29	0.08	0.02	0

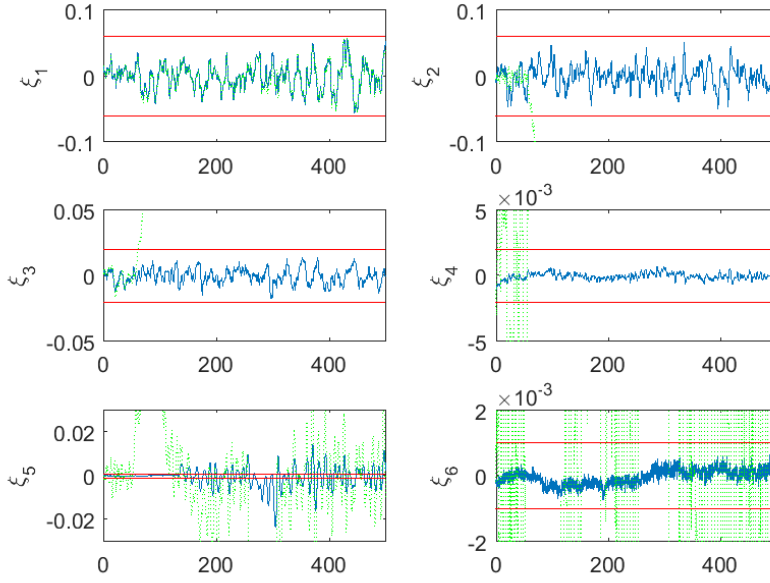


Fig. 5 Residuals $\xi_i(t)$, $i = 1, \dots, 6$, in fault scenario 3 (without sensor noise/uncertainties).

desired, appropriate values for the thresholds are as follows: $\xi_1^{th} = 0.20$, $\xi_2^{th} = 0.35$, $\xi_3^{th} = 0.25$, $\xi_4^{th} = 0.35$, $\xi_5^{th} = 0.08$, $\xi_6^{th} = 0.10$.

5 Conclusion

This paper has proposed a discrete-time LPV UIO fault/icing diagnosis scheme for unmanned aerial vehicles (UAVs). It has been shown that the nonlinear 6-DOF model of a UAV can be brought to a quasi-LPV form using the nonlinear embedding in the parameters approach. Wind disturbance, actuator faults and icing are added into the model as additive disturbances. Then, the LPV UIO matrices are designed such that different output directions of the residuals are assigned for the unknown inputs acting on the system. In this way, an algorithm for deciding about the occurrence of faults/icing can be proposed, under the assumption that at a given instant either a single fault or icing could act on the system. In addition, a metric that can be used for analysing the robustness of the proposed method has been suggested, along with a Monte Carlo strategy for estimating this metric with some desired bound on the standard deviation.

The proposed actuator faults and icing diagnosis method has been validated using the case study of a typical small UAV. Four different scenarios have been analysed, i.e. i) fault in thrust; ii) fault in elevator; iii) fault in ailerons; and iv) icing. The results obtained without sensor noise and uncertainties show that the proposed approach is

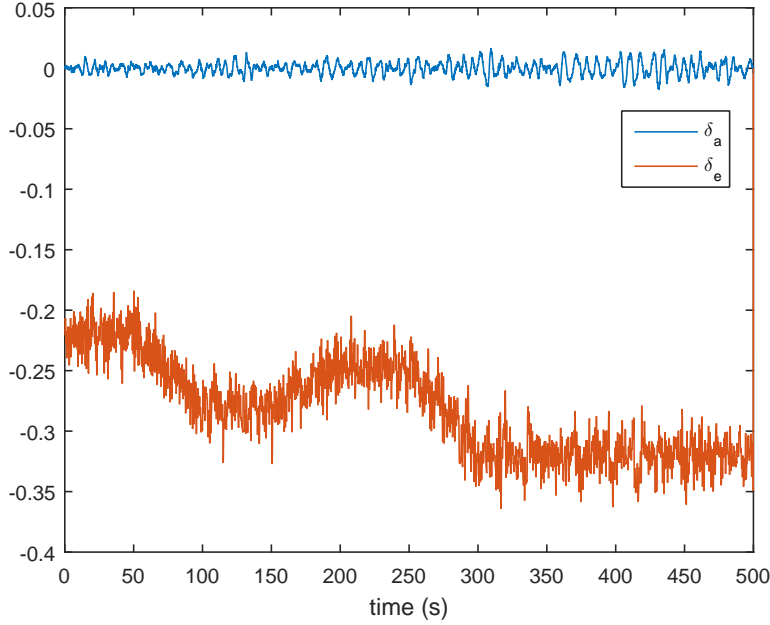


Fig. 6 Inputs δ_a and δ_e in nominal scenario (without sensor noise/uncertainties).

able to diagnose correctly all the considered faults when the theoretical assumptions under which the UIO has been developed hold. However, when sensor noise and uncertainties are included into the simulations, the trade-off between avoiding false alarms and making the LPV UIO-based scheme sensitive to faults/icing must be taken into account, in order to choose appropriate values for the residual thresholds. In this case, the robustness metric and the extensive Monte Carlo simulations have proven to be useful tools for performing this choice.

It seems evident that before achieving a successful application of the proposed strategy on a real UAV operating under harsh weather conditions, it will be necessary to enhance the sensitivity of the UIO to faults and icing while decreasing the sensitivity to noise and uncertainties. Also, active trajectory planning will be investigated, in particular with the aim of allowing a successful aileron fault diagnosis.

Appendix

$$a_{12}(\cdot) = \sin \theta$$

$$a_{13}(\cdot) = -\sin \phi \cos \theta$$

$$a_{14}(\cdot) = -\cos \phi \cos \theta$$

$$a_{22}(\cdot) = \frac{\rho u S}{2m} \left[(C_{L_0} + C_{L_\alpha} \alpha) \sin \alpha - (C_{D_0} + C_{D_\alpha} \alpha) \cos \alpha - \frac{S_{prop} C_{prop}}{S} \right]$$

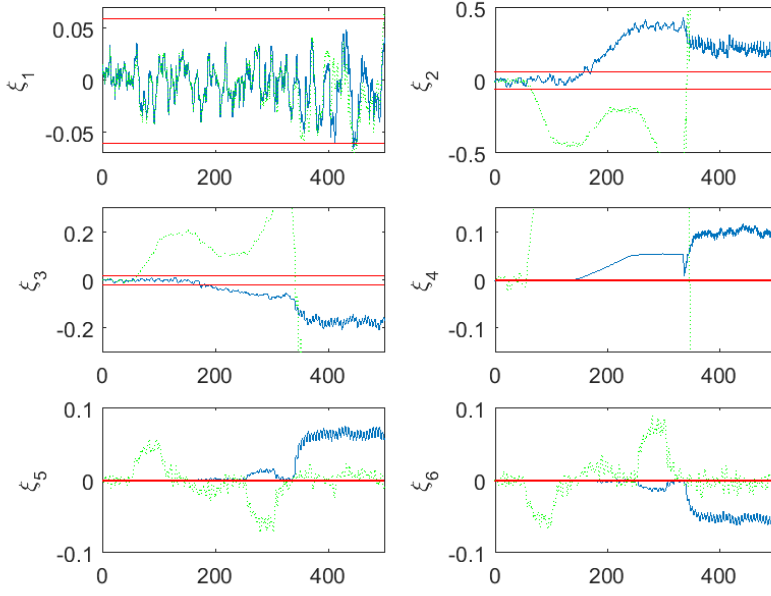


Fig. 7 Residuals $\xi_i(t)$, $i = 1, \dots, 6$, in fault scenario 4 (without sensor noise/uncertainties).

$$a_{23}(\cdot) = \frac{\rho v S}{2m} \left[(C_{L_0} + C_{L_\alpha} \alpha) \sin \alpha - (C_{D_0} + C_{D_\alpha} \alpha) \cos \alpha - \frac{S_{prop} C_{prop}}{S} \right]$$

$$a_{24}(\cdot) = \frac{\rho w S}{2m} \left[(C_{L_0} + C_{L_\alpha} \alpha) \sin \alpha - (C_{D_0} + C_{D_\alpha} \alpha) \cos \alpha - \frac{S_{prop} C_{prop}}{S} \right]$$

$$a_{29}(\cdot) = -w + \frac{\rho S c V_a}{4m} (C_{L_q} \sin \alpha - C_{D_q} \cos \alpha)$$

$$a_{210}(\cdot) = v$$

$$a_{32}(\cdot) = \frac{\rho u S}{2m} (C_{Y_0} + C_{Y_\beta} \beta)$$

$$a_{33}(\cdot) = \frac{\rho v S}{2m} (C_{Y_0} + C_{Y_\beta} \beta)$$

$$a_{34}(\cdot) = \frac{\rho w S}{2m} (C_{Y_0} + C_{Y_\beta} \beta)$$

$$a_{38}(\cdot) = w + \frac{\rho S b V_a}{4m} C_{Y_p}$$

$$a_{310}(\cdot) = -u + \frac{\rho S b V_a}{4m} C_{Y_r}$$

$$a_{42}(\cdot) = -\frac{\rho u S}{2m} [(C_{D_0} + C_{D_\alpha} \alpha) \sin \alpha + (C_{L_0} + C_{L_\alpha} \alpha) \cos \alpha]$$

$$a_{43}(\cdot) = -\frac{\rho v S}{2m} [(C_{D_0} + C_{D_\alpha} \alpha) \sin \alpha + (C_{L_0} + C_{L_\alpha} \alpha) \cos \alpha]$$

$$a_{44}(\cdot) = -\frac{\rho w S}{2m} [(C_{D_0} + C_{D_\alpha} \alpha) \sin \alpha + (C_{L_0} + C_{L_\alpha} \alpha) \cos \alpha]$$

$$a_{48}(\cdot) = -v$$

$$a_{49}(\cdot) = u - \frac{\rho S c V_a}{4m} (C_{L_q} \cos \alpha + C_{D_q} \sin \alpha)$$

$$a_{59}(\cdot) = \sin \phi \tan \theta$$

$$a_{510}(\cdot) = \cos \phi \tan \theta$$

$$a_{69}(\cdot) = \cos \phi$$

$$a_{610}(\cdot) = -\sin \phi$$

$$a_{79}(\cdot) = \sin \phi \sec \theta$$

$$a_{710}(\cdot) = \cos \phi \sec \theta$$

$$a_{82}(\cdot) = \frac{1}{2} \rho S b u (C_{p_0} + C_{p_\beta} \beta)$$

$$a_{83}(\cdot) = \frac{1}{2} \rho S b v (C_{p_0} + C_{p_\beta} \beta)$$

$$a_{84}(\cdot) = \frac{1}{2} \rho S b w (C_{p_0} + C_{p_\beta} \beta)$$

$$a_{88}(\cdot) = \frac{\Gamma_1 q}{2} + \frac{\rho V_a S b^2}{4} C_{p_p}$$

$$a_{89}(\cdot) = \frac{\Gamma_1 p}{2} - \frac{\Gamma_2 r}{2}$$

$$a_{810}(\cdot) = -\frac{\Gamma_2 q}{2} + \frac{\rho V_a S b^2}{4} C_{p_r}$$

$$a_{92}(\cdot) = \frac{\rho S c u}{2J_y} (C_{m_0} + C_{m_\alpha} \alpha)$$

$$a_{93}(\cdot) = \frac{\rho S c v}{2J_y} (C_{m_0} + C_{m_\alpha} \alpha)$$

$$a_{94}(\cdot) = \frac{\rho S c w}{2J_y} (C_{m_0} + C_{m_\alpha} \alpha)$$

$$a_{98}(\cdot) = \frac{\Gamma_5 r}{2} - \Gamma_6 p$$

$$a_{99}(\cdot) = \frac{\rho V_a S c^2}{4J_y} C_{m_q}$$

$$a_{910}(\cdot) = \frac{\Gamma_5 p}{2} + \Gamma_6 r$$

$$a_{101}(\cdot) = \frac{1}{2} \rho S b u (C_{r_0} + C_{r_\beta} \beta)$$

$$\begin{aligned}
a_{102}(\cdot) &= \frac{1}{2} \rho S b v \left(C_{r_0} + C_{r_\beta} \beta \right) \\
a_{103}(\cdot) &= \frac{1}{2} \rho S b w \left(C_{r_0} + C_{r_\beta} \beta \right) \\
a_{108}(\cdot) &= \frac{\Gamma_7 q}{2} + \frac{\rho V_a S b^2}{4} C_{r_p} \\
a_{109}(\cdot) &= \frac{\Gamma_7 q}{2} - \frac{\Gamma_1 q}{2} \\
a_{1010}(\cdot) &= -\frac{\Gamma_1 q}{2} + \frac{\rho V_a S b^2}{4} C_{r_r} \\
b_{21} &= \frac{\rho S_{prop} C_{prop}}{2m} k_m^2 \\
b_{22}(\cdot) &= \frac{\rho V_a^2 S}{2m} \left(C_{L_{\delta_e}} \sin \alpha - C_{D_{\delta_e}} \cos \alpha \right) \\
b_{33}(\cdot) &= \frac{\rho V_a^2 S}{2m} C_{Y_{\delta_a}} \\
b_{34}(\cdot) &= \frac{\rho V_a^2 S}{2m} C_{Y_{\delta_r}} \\
b_{42}(\cdot) &= -\frac{\rho V_a^2 S}{2m} \left(C_{L_{\delta_e}} \cos \alpha + C_{D_{\delta_e}} \sin \alpha \right) \\
b_{83}(\cdot) &= \frac{1}{2} \rho V_a^2 S b C_{p_{\delta_a}} \\
b_{84}(\cdot) &= \frac{1}{2} \rho V_a^2 S b C_{p_{\delta_r}} \\
b_{92}(\cdot) &= \frac{\rho V_a^2 S c}{2J_y} C_{m_{\delta_e}} \\
b_{103}(\cdot) &= \frac{1}{2} \rho V_a^2 S b C_{r_{\delta_a}} \\
b_{104}(\cdot) &= \frac{1}{2} \rho V_a^2 S b C_{r_{\delta_r}} \\
g_2(\cdot) &= \frac{\rho V_a^2 S}{2m} \left(K_{L_\alpha} C_{L_\alpha} \alpha \sin \alpha - K_{D_0} C_{D_0} \cos \alpha + K_{L_q} C_{L_q} \sin \alpha \frac{cq}{2V_a} + K_{L_{\delta_e}} C_{L_{\delta_e}} \delta_e \sin \alpha \right) \\
g_3(\cdot) &= \frac{\rho V_a^2 S}{2m} \left(K_{Y_\beta} C_{Y_\beta} \beta + K_{Y_{\delta_r}} C_{Y_{\delta_r}} \delta_r \right) \\
g_4(\cdot) &= -\frac{\rho V_a^2 S}{2m} \left(K_{L_\alpha} C_{L_\alpha} \alpha \cos \alpha + K_{D_0} C_{D_0} \sin \alpha + K_{L_q} C_{L_q} \sin \alpha \frac{cq}{2V_a} + K_{L_{\delta_e}} C_{L_{\delta_e}} \delta_e \sin \alpha \right) \\
g_8(\cdot) &= \frac{1}{2} \rho V_a^2 S b \left(\Gamma_3 K_{l_\beta} C_{l_\beta} \beta + \Gamma_4 K_{n_\beta} C_{n_\beta} \beta + \Gamma_3 K_{l_p} C_{l_p} \frac{bp}{2V_a} \right. \\
&\quad \left. + \Gamma_4 K_{n_r} C_{n_r} \frac{br}{2V_a} + \Gamma_3 K_{l_{\delta_a}} C_{l_{\delta_a}} \delta_a + \Gamma_3 K_{l_{\delta_r}} C_{l_{\delta_r}} \delta_r + \Gamma_4 K_{n_{\delta_r}} C_{n_{\delta_r}} \delta_r \right)
\end{aligned}$$

$$\begin{aligned}
g_9(\cdot) &= \frac{\rho V_a^2 S c}{2 J_y} \left(K_{m_\alpha} C_{m_\alpha} \alpha + K_{m_q} C_{m_q} \frac{c q}{2 V_a} + K_{m_{\delta_e}} C_{m_{\delta_e}} \delta_e \right) \\
g_{10}(\cdot) &= \frac{1}{2} \rho V_a^2 S b \left(\Gamma_4 K_{l_\beta} C_{l_\beta} \beta + \Gamma_8 K_{n_\beta} C_{n_\beta} \beta + \Gamma_4 K_{l_p} C_{l_p} \frac{b p}{2 V_a} \right. \\
&\quad \left. + \Gamma_8 K_{n_r} C_{n_r} \frac{b r}{2 V_a} + \Gamma_4 K_{l_{\delta_a}} C_{l_{\delta_a}} \delta_a + \Gamma_4 K_{l_{\delta_r}} C_{l_{\delta_r}} \delta_r + \Gamma_8 K_{n_{\delta_r}} C_{n_{\delta_r}} \delta_r \right)
\end{aligned}$$

References

1. Zhang, Y., Jiang, J., "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual reviews in control*, Vol. 32, pp. 229–252 (2008).
2. Gao, Z., Cecati, C., Ding, S. X., "A survey of fault diagnosis and fault-tolerant techniques - Part I: fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, Vol. 62, No. 6, pp. 3757–3767 (2015).
3. De Persis, C., Isidori, A., "A geometric approach to nonlinear fault detection and isolation," *IEEE Transactions on Automatic Control*, Vol. 46, No. 6, pp. 863–865 (2001).
4. Kaboré, P., Wang, H., "Design of fault diagnosis filters and fault-tolerant control for a class of nonlinear systems," *IEEE Transactions on Automatic Control*, Vol. 46, No. 11, pp. 1805–1810 (2001).
5. López-Estrada, F. R., Ponsart, J. C., Theilliol, D., Zhang, Y., Astorga-Zaragoza, C. M., "LPV model-based tracking control and robust sensor fault diagnosis for a quadrotor UAV," *Journal of Intelligent and Robotic Systems*, Vol. 84, No. 1, pp. 163–177 (2016).
6. Caliskan, F., Hajjiev, C., "A review of in-flight detection and identification of aircraft icing and reconfigurable control," *Progress in Aerospace Sciences*, Vol. 60, pp. 12–34 (2013).
7. Gent, R., Dart, N., Cansdale, J., "Aircraft icing," *Phil. Trans. of the Royal Soc. of London, Series A: Mathematical, Physical and Engineering Sciences*, Vol. 358, pp. 2873–2911 (2000).
8. Lampton, A., Valasek, J., "Prediction of icing effects on the dynamic response of light airplanes," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 3, pp. 722–732 (2007).
9. Rutherford, R., Dudman, R., "Zoned aircraft de-icing system and method," U.S. Patent 6237874, filed 15 Oct. 1999. Available: <http://www.google.com/patents/US6237874>
10. Sørensen, K. L., Helland, A. S., Johansen, T. A., "Carbon nanomaterial-based wing temperature control system for in-flight anti-icing and de-icing of unmanned aerial vehicles," *IEEE Aerospace Conference*(2015).
11. Sørensen, K. L., Johansen, T. A., "Thermodynamics of a carbon nano-materials based icing protection system for unmanned aerial vehicle," *IEEE Aerospace Conference* (2016).
12. Hajjiev C., Caliskan, F., *Fault diagnosis and reconfiguration in flight control systems*, Springer Science and Business Media (2003).
13. Tousi, M., Khorasani, K., "Robust observer-based fault diagnosis for an unmanned aerial vehicle," *Proc. of the IEEE International Systems Conference (SysCon)*, pp. 428–434 (2011).
14. Sørensen, K. L., Blanke, M., Johansen, T. A., "Diagnosis of wing icing through lift and drag coefficient change detection for small unmanned aircraft," *Proc. of the 9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (2015).
15. Lee, H., Snyder, S., Hovakimyan, N., "An adaptive unknown input observer for fault detection and isolation of aircraft actuator faults," *AIAA Guidance, Navigation, and Control Conference* (2014).
16. Hammouri, H., Tmar, Z., "Unknown input observer for state affine systems: a necessary and sufficient condition," *Automatica*, Vol. 46, pp. 271–278 (2010).
17. Basile, G., Marro, G., "On the observability of linear, time-invariant systems with unknown inputs," *Journal of Optimization Theory and Applications*, Vol. 3, pp. 410–415 (1969).
18. Guidorzi, R., Marro, G., "On Wonham stabilizability condition in the synthesis of observers for unknown-input systems," *IEEE Transactions on Automatic Control*, Vol. 16, No. 5, pp. 499–500 (1971).
19. Bhattacharyya, S. P., "Observer design for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, Vol. 23, pp. 483–484 (1978).
20. Hou, M., Muller, P. C., "Design of observers for linear systems with unknown inputs," *IEEE Transactions on Automatic Control*, Vol. 37, pp. 871–875 (1992).

21. Chen, J., Patton, R. J., Zhang, H. Y., "Design of unknown input observers and robust fault-detection filters," *International Journal of Control*, Vol. 63, No 1, pp. 85–105 (1996).
22. Cristofaro, A., Johansen, T. A., "Fault tolerant control allocation using unknown input observers," *Automatica*, Vol. 50, pp. 1891–1897 (2014).
23. Cristofaro, A., Johansen, T. A., "An unknown input observer approach to icing detection for unmanned aerial vehicles with linearized longitudinal motion," *Proc. of the American Control Conference (ACC)*, pp. 207–213 (2015).
24. Cristofaro, A., Johansen, T. A., Aguiar, A. P., "Icing detection and identification for unmanned aerial vehicles: multiple model adaptive estimation," *Proc. of the European Control Conference (ECC)*, pp.1645–1650 (2015).
25. Rotondo, D., Cristofaro, A., Johansen, T. A., Nejjari, F., Puig, V., "Icing detection in unmanned aerial vehicles with longitudinal motion," *Proc. of the IEEE Multi-conference on Systems and Control (MSC)*, pp. 984–989 (2015).
26. Seron, M. M., Johansen, T. A., De Doná, J. A., Cristofaro, A., "Detection and estimation of icing in unmanned aerial vehicles using a bank of unknown input observers," *Proc. of the Australian Control Conference (AuCC)*, pp. 87–92 (2015).
27. Hoffmann, C., Werner, H., "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 2, pp. 416–433 (2015).
28. Yu, Z., Chen, H., Woo, P. Y. "Gain scheduled output feedback control based on LTI controller interpolation that preserves LPV \mathcal{H}_∞ performance," *Journal of Intelligent and Robotic Systems*, Vol. 40, No. 2, pp. 183–206 (2004).
29. Shamma, J., Athans, M., "Guaranteed properties of gain scheduled control for linear parameter-varying plants," *Automatica*, Vol. 27, No. 3, pp. 559–564 (1991).
30. Shamma, J., "An overview of LPV systems," *Control of Linear Parameter Varying Systems with Applications*, edited by J. Mohammadpour and C. Scherer, Springer (2012).
31. Balas, G., "Linear parameter-varying control and its application to a turbofan engine," *International Journal of Robust and Nonlinear Control*, Vol. 12, pp. 763–796 (2002).
32. Beard, R. W., McLain, T. W., *Small Unmanned Aircraft: Theory and Practice*, Princeton University Press, Princeton, NJ, USA (2012).
33. Kwiatkowski, A., Boll, M.-T., Werner, H., "Automated generation and assessment of affine LPV models," *Proc. of the 45th IEEE Conference on Decision and Control (CDC)*, pp. 6690–6695 (2006).
34. Rotondo, D., Puig, V., Nejjari, F., Witczak, M., "Automated generation and comparison of Takagi-Sugeno and polytopic quasi-LPV models," *Fuzzy Sets and Systems*, Vol. 277, pp. 44–64 (2015).
35. Hoblit, F. M., *Gust loads on aircraft: concept and applications*, Washington, DC, USA: American Institute of Aeronautics and Astronautics (1988).
36. Johansen, T. A., Cristofaro, A., Sørensen, K. L., Hansen, J. M., Fossen, T. I., "On estimation of wind velocity, angle-of-attack and sideslip angle of small UAVs using standard sensors," *Proc. of the International Conference on Unmanned Aircraft Systems (ICUAS)* (2015).
37. Bragg, M. B., Hutchinson, T., Merret, J., Oltman, R., Pokhariyal, D., "Effect of ice accretion on aircraft flight dynamics," *Proc. of the 38th AIAA Aerospace Science Meeting and Exhibit* (2000).
38. MacDonald, S. A. F., *From the Ground Up*, Aviation Publishers Company, Limited (2000).
39. Hansen, S., Blanke M., "Diagnosis of airspeed measurement faults for unmanned aerial vehicles," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 50, No. 1, pp. 224–239 (2014).
40. Toth, R., Henberger, P. S. C., Van Den Hof, P. M. J., "Discretisation of linear parameter-varying state space representations," *IET Control Theory and Applications*, Vol. 4, No. 10, pp. 2082–2096 (2010).
41. Witczak, M., Pretki, P., "Design of an extended unknown input observer with stochastic robustness techniques and evolutionary algorithms," *International Journal of Control*, Vol. 80, No. 5, pp. 749–762 (2007).
42. Marrison, C. I., Stengel, R. F., "Robust control system design using random search and genetic algorithms," *IEEE Transactions on Automatic Control*, Vol. 42, pp. 835–839 (1997).
43. Doucet, A., de Freitas, N., Gordon, N. *Sequential Monte Carlo Methods in Practice*, Springer-Verlag London (2001).