machines

MDPI

Article

Active Fault Diagnosis and Control of a Morphing Multirotor Subject to a Stuck Arm †

Alessandro Baldini *, Riccardo Felicetti , Alessandro Freddi and Andrea Monteriù

Department of Information Engineering, Università Politecnica delle Marche, Via Brecce Bianche 12, 60131 Ancona, Italy; r.felicetti@univpm.it (R.F.); a.freddi@univpm.it (A.F.); a.monteriu@univpm.it (A.M.)
* Correspondence: a.baldini@univpm.it; Tel.: +39-071-220-4462

† This paper is an extended version of our paper published in 2022 30th Mediterranean Conference on Control and Automation (MED '22), titled "Modeling and Control of a Telescopic Quadrotor Using Disturbance Observer Based Control".

Abstract: In this paper, we propose a fault tolerant control law for a morphing quadrotor, where the considered morphing ability is that of extendable/telescopic arms. This quite recent class of systems is able to provide a good trade-off between payload capabilities, maneuverability, and space occupancy. However, such degrees of freedom require dedicated servomotors, which in turn implies more possible faults. Thus, the problem of diagnosis for the telescopic servo motors subject to a stuck fault is considered. System symmetries are exploited and used in a residual generator design, which triggers an active fault isolation/identification phase. External disturbances are also taken into account and estimated through a nonlinear disturbance observer. A classical double-loop controller closes the loop, providing an overall control system structure that follows the disturbance observer-based control paradigm. The control scheme is validated through realistic numerical simulations, and the closed-loop performances are analyzed.

Keywords: active fault diagnosis; morphing quadrotor; fault-tolerant control; disturbance observer

check for updates

Citation: Baldini, A.; Felicetti, R.; Freddi, A.; Monteriù, A. Active Fault Diagnosis and Control of a Morphing Multirotor Subject to a Stuck Arm. *Machines* **2023**, *11*, 511. https://doi.org/10.3390/machines11050511

Academic Editor: Dan Zhang

Received: 30 March 2023
Revised: 20 April 2023
Accepted: 21 April 2023
Published: 25 April 2023

1. Introduction

The size of Unmanned Aerial Vehicles (UAVs) is not a detail, as for different sizes different challenges arise. For example, mini, micro, and nano UAVs [1,2] have been largely studied in the past years, but the downside of miniaturization is the lower payload, which is not acceptable in many applications. On the contrary, maneuvering becomes more involved as the vehicle mass and inertia increase, and, therefore, a trade-off between the system bandwidth and the maximum payload is a current research problem. The solution is investigated through two main layers, the actuation and the mechanical system structure of the UAV. Concerning the first, variable pitch actuators are the most remarkable example [3]. Among the advantages, they show faster time response, leading to reduced power requirements and motor size [4]. It is worth noting that variable pitch configurations allow downward thrusts (without reversing the angular speed of the propellers), thus showing increased fault tolerant capabilities [5,6]. Regarding the second, two alternative ways have been investigated to reduce the size without compromising payload capabilities. The first class to note is the so-called folding multirotors, which can be split into multi-joint structures [7] and continuous deformable structures, such as origami foldable structures [8]. Such systems consist of UAVs being able to be folded/disassembled after the operation, even reaching a pocket-size dimension for a comfortable transportation [9]. During the operation, however, they exhibit a fixed structure, and no system reconfiguration is allowed, except for exceptional triggering events [10]. The second class is represented by reconfigurable or morphing multirotors that allow for in-flight morphing of the UAV [11]. The most common degrees of freedom are represented by extensible arms [12] (telescopic or sliding arms) and rotating arms [13,14]. By transforming its structure, the UAV does modify its space occupancy, its inertia [15], and its maneuverability based on the required task [16]. From a control law perspective, the morphing ability represents an additional degree of freedom [17], which is usually handled by a continuous refresh of the center of mass, the inertia and the allocation matrices, leading to a morphology-dependent control. Thus, assuming the transformation is relatively slow, only minor modifications need to be applied to the most common control techniques, such as proportional integral control [18], linear quadratic regulator [13], and active disturbance rejection control [19]. External disturbances cope with the system dynamics in the same way as standard multirotors. Many estimation methods have been proposed in the literature, ranging from specific oriented solutions (see [20] and references therein) to general purpose solutions such as Disturbance Observer-Based Control (DOBC), where the disturbance attenuation is provided by an observer-based feed-forward law [21]. Finally, the transformation ability implies additional actuation degrees, which in turn means possible additional faults with respect to those of standard multirotors, due to the servomotors involved in the morphing mechanisms. Such an acquired over-actuation brings the need for directional residues, which we can find in the observer-based residual generators [22], a possible solution with strong evidence in the literature and general purpose results.

In this paper, we propose an active fault-tolerant control scheme for morphing quadrotors, where the terminology fault-tolerant refers to a control law which is able to cope with faults, and by active it is meant that the fault is directly estimated and feedforwarded in the control loop [23,24]. The solution extends our previous work [25], where a DOBC scheme which copes with external disturbances was designed by joining a Nonlinear Disturbance Observer (NDO) with an inner/outer loop feedback linearization control law. Thus, the main objective is that of extending [25] by considering stuck faults in the morphology-related servomotors. The considered morphing ability is that of telescopic arms (extendable arms), which represent a promising solution to flight into narrow spaces while keeping payload and maneuvering capabilities [12]. The diagnosis is performed by exploiting the system symmetries. It turns out that a classical bank of observers (i.e., a residual generator) is not sufficient to cope with the problem, which represents a main step for the solution. In particular, after one residual component triggers, an active fault isolation, and identification phase is required, i.e., the isolation/identification is performed by the injection of a specific class of control inputs.

The paper is structured as follows. In Section 2, we propose and detail the mathematical model of a telescopic quadrotor. System symmetries are discussed and exploited in Section 3. Section 4 handles the overall fault diagnosis, both the residual generator and the active fault diagnosis. Section 5 briefly recalls the DOBC solution provided in [25], which closes the loop. Finally, numerical simulations are discussed in Section 6, where masses and dimensions are taken from commercial components, and practical real-world problems (e.g., input saturation, sensor noise, and different sample rate between the two control loops) are also taken into account.

2. Mathematical Model

Remark 1. *From now on, standard basis vectors of $\mathbb{R}^n$ are denoted as e_k with $k = 1, \dots, n$, diag$(\cdot)$ denoting the diagonal matrix with the elements of the argument in the main diagonal, and col$(\cdot)$ denoting a column vector whose elements are listed in the argument.*

Telescopic quadrotors are quadrotors with extendable (telescopic) arms. The whole system can be considered as a collection of $N = 17$ rigid bodies subject to constraints. The frame is denoted with B_0, while the remaining bodies are denoted with $B_{i,j}$, where $i \in \{1,2,3,4\}$ denotes the arm and $j \in \{1,2,3,4\}$ denotes the component (Figure 1). For each arm, we denote with l_i the length of the arm.

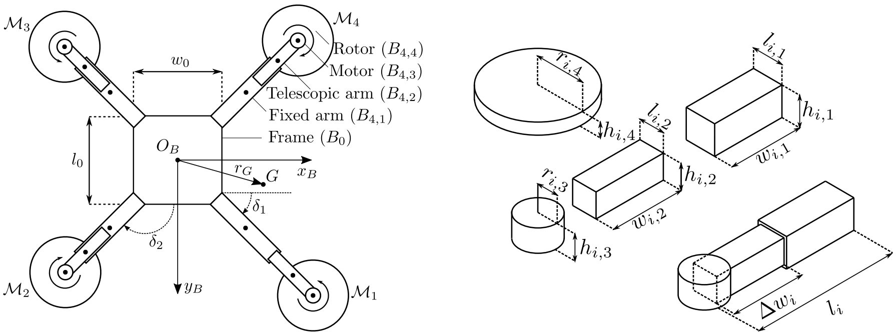

Figure 1. Schematic representations of the morphing quadrotor (**left**) and the telescopic arm (**right**). Each arm is independently controlled by a servomotor, possibly leading to an asymmetric structure. The telescopic arm is considered as a union of four bodies, where for each one the geometric dimensions are assumed known. For control purposes, as well as for a better interpretation, the overall arm length l_i will be always made explicit. The bodies' centers of mass are depicted with a bold point, and we assume they coincide with each body's geometric center.

Two reference frames are considered. The first reference frame is an earth fixed frame $R_E = (O_E, \mathcal{T}_E)$ with a right-handed orthonormal triad $\mathcal{T}_E = (x_E, y_E, z_E)$, which is assumed to be inertial for simplicity. Then, a second frame $R_B = (O_B, \mathcal{T}_B)$ with a right-handed orthonormal triad $\mathcal{T}_B = (x_B, y_B, z_B)$ is placed in the geometric center O_B of the frame B_0. Reference frame R_B is a rest frame for the quadrotor frame, i.e., it is a body fixed frame for B_0 (see Figure 1). For slowly varying arm lengths, the telescopic quadrotor can be approximated as a rigid body that has its own center of mass, denoted as G, and described by the following equations [26]

$$\begin{aligned} m\ddot{p}_F &= -k_t\dot{p}_F - mR_B^E(\dot{\omega} \times r_G^B(l) + \omega \times (\omega \times r_G^B(l))) + F_g^E + R_B^E F_m^B(l) + F_w^E \\ J(l)\dot{\omega} &= -k_r\omega - \omega \times J(l)\omega + M_m^B(l) \\ \dot{\eta} &= T(\eta)\omega, \end{aligned} \tag{1}$$

where each term is as follows:

- $p_F = \overrightarrow{O_E O_B} = \mathrm{col}(x_F, y_F, z_F)$ is the position of R_B (decomposed in R_E);
- $\omega = \mathrm{col}(p, q, r)$ is the angular velocity of R_B relative to R_E (decomposed in R_B);
- m is the total system mass, g is the gravitational acceleration, and k_t and k_r are the linear and angular friction coefficients;
- $F_g^E = mge_3$ and F_w^E are the gravitational and the (unknown) wind forces (decomposed in R_E);
- $l = \mathrm{col}(l_1, \ldots, l_4)$ is the vector containing the arm lengths of each motor (see Figure 1);
- $r_G^B(l) = \overrightarrow{O_B G}$ is the arm-dependent center of mass (decomposed in R_B);
- $J(l)$ is the arm-dependent inertia matrix of the overall system, relative to the center of mass;
- $F_m^B(l)$ and $M_m^B(l)$ are the arm-dependent thrust and torque due to the actuators (both decomposed in R_B);
- $\eta = \mathrm{col}(\varphi, \theta, \psi)$ is the vector composed by the roll (φ), pitch (θ), and yaw (ψ) angles, which let us express the rotation matrix from R_B to R_E, denoted as R_B^E, as

$$R_B^E = R_B^E(\eta) = \begin{bmatrix} c_\psi\, c_\theta & c_\psi\, s_\varphi\, s_\theta - c_\varphi\, s_\psi & s_\varphi\, s_\psi + c_\varphi\, c_\psi\, s_\theta \\ c_\theta\, s_\psi & c_\varphi\, c_\psi + s_\varphi\, s_\psi\, s_\theta & c_\varphi\, s_\psi\, s_\theta - c_\psi\, s_\varphi \\ -s_\theta & c_\theta\, s_\varphi & c_\varphi\, c_\theta \end{bmatrix}, \tag{2}$$

  where $\cos(\cdot) = c_{(\cdot)}$ and $\sin(\cdot) = s_{(\cdot)}$ are considered for the sake of brevity;
- $T(\eta)$ is the kinematic coordinate transformation related to the adopted roll–pitch–yaw rotation.

Remark 2. *In* (1)*, the time dependencies are omitted for the sake of brevity. We remark that both l and η are time-dependent variables.*

Remark 3. *In the following, the arm and angle dependencies are often omitted for the sake of brevity.*

Remark 4. *The following common approximations were made in the model* [6]*:*

- *no torques due to the wind are considered;*
- *additional forces and torques due to blade flapping are of a smaller magnitude and neglected;*
- *the friction force is assumed to be linear, with proportional coefficients k_t, k_r.*

For a brief description, let us denote the mass of the body $B_{i,j}$ with $m_{i,j}$, and the vector connecting the body frame with the body $B_{i,j}$ with $r^B_{i,j} = \overrightarrow{O_B G_{i,j}}$ (decomposed in R_B). The remaining variables are as follows.

2.1. Total Mass

The total mass m is the constant real number

$$m = m_0 + \sum_{i=1}^{4} \sum_{j=1}^{4} m_{i,j}. \tag{3}$$

2.2. Center of Mass

The center of mass is the arm-dependent vector

$$r^B_G = \frac{1}{m}\left(m_0 r^B_0 + \sum_{i=1}^{4} \sum_{j=1}^{4} m_{i,j} r^B_{i,j} \right). \tag{4}$$

2.3. Inertia Matrix

The inertia is the arm-dependent matrix

$$J = J_0 + \sum_{i=1}^{4} \sum_{j=1}^{4} J_{i,j}, \tag{5}$$

where each $J_{i,j}$ denotes the inertia of the body $B_{i,j}$ relative to axes which are parallel to x_B, y_B, and z_B, and which pass through the center of mass, i.e., parallel the R_B axes. In turn, each $J_{i,j}$ is dependent from the geometric dimensions, i.e., from the width $w_{i,j}$, the length $l_{i,j}$, the height $h_{i,j}$, and the orientation of $B_{i,j}$ (see Figure 1). Denoting the angle of the i-th arm with δ_i (around z_B, from x_B, as depicted in Figure 1), the orientation of the body $B_{i,j}$ is described by the rotation matrix

$$R^B_{i,j} = R^B_{i,j}(\delta_i) = \begin{bmatrix} \cos(\delta_i) & -\sin(\delta_i) & 0 \\ \sin(\delta_i) & \cos(\delta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad i = 1,2,3,4, \qquad j = 1,2,3,4. \tag{6}$$

Using the $R^B_{i,j}$s and $r^B_{i,j}$s, it is possible to express the inertia matrices $J_{i,j}$ through the parallel axis theorem. The reader can refer to [13,15,25] for details about the calculation of each $J_{i,j}$.

2.4. Actuation Force and Moment

For each propeller i, let f_i be the (scalar) upward lift force. Then $f^B_i = -f_i e_3$ is the vector lift force (decomposed in R_B), and the actuation force is

$$F^B_m = \sum_{i=1}^{4} f^B_i = -\sum_{i=1}^{4} f_i e_3. \tag{7}$$

The actuation torque M_m^B can be decomposed into the thrust component and the drag component, namely $M_m^B = M_{thrust}^B + M_{drag}^B$. In particular, we have

$$M_{thrust}^B = -\sum_{i=1}^{4}\Big((r_{i,4}^B - r_G^B) \times f_i e_3\Big), \tag{8}$$

$$M_{drag}^B = \sum_{i=1}^{4} c_{w_i} \frac{c_D}{c_L} f_i e_3, \tag{9}$$

where c_L and c_D are the lift and drag coefficients, while $c_{w_i} = +1$ if the i-th motor rotates counter-clockwise, and $c_{w_i} = -1$ otherwise.

2.5. Inputs and Faults

Each lift force f_i is assumed directly controlled, hence the vector $u = \mathrm{col}(f_1, f_2, f_3, f_4)$ serves as control input. The extension of each arm l_i is performed by a servomotor, which can be affected by a stuck fault. If l_{ni} denotes the nominal length (the reference for l_i), the servomotor provides

$$l_i = \begin{cases} l_{n_i} & \text{if the } i\text{-th servomotor is healthy} \\ \bar{l}_i & \text{if the } i\text{-th servomotor is stuck,} \end{cases} \tag{10}$$

where in case of a stuck fault, the constant $\bar{l}_i$ is unknown. Introducing the stuck offset $s_o = \mathrm{col}(s_{01}, s_{02}, s_{03}, s_{04})$, where $s_{oi} = \bar{l}_i - l_{ni}$ is the stuck offset of the i-th arm, we have

$$l = l_n + s_o. \tag{11}$$

Remark 5. *In the following, the assumption of at most one stuck fault at a time is made, that is, if $s_{oi} \neq 0$ for some i, then $s_{oj} = 0$ for $j \neq i$. In these terms, Fault Detection (FD) consists of determining if $s_{oi} \neq 0$ for some i, Fault Isolation (FI) consists of determining which i is such that $s_{oi} \neq 0$, and Fault Identification (FId) consists of the estimation of $\bar{l}_i$.*

Remark 6. *Both u and l_n are available inputs. The inputs u are fast and can be used for actual flight control; on the other hand, the inputs l_n are slow, and commonly set by an external module for specific maneuvers (e.g., a supervisory module which commands a shrinking/enlarging phase). As such, only u will be used for the tracking control goal, while l_n will play a main role in the active stuck fault isolation. To this end, let $l^\star = col(l_1^\star, l_2^\star, l_3^\star, l_4^\star)$ be the desired/optimal arm lengths provided by the external module; ideally, we would like to achieve $l = l^\star$ by setting $l_n = l^\star$. This will be not true during FI.*

Remark 7. *In a matrix-like fashion, it is possible to rewrite F_m^B and M_m^B in terms of u as*

$$F_m^B = F_1 u, \qquad M_m^B = F_2 u, \tag{12}$$

where $F_1 \in \mathbb{R}^{3\times4}$ is the control force input matrix and F_2 is the so called control moment input matrix [27]. Trivially,

$$F_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & -1 \end{bmatrix}, \tag{13}$$

while F_2 will be exploited later.

Remark 8. *Note that r_G^B, J and F_2 are arm-dependent and, as such, time-varying. These quantities can be calculated online. As such, in the next sections, their dependence on the arms' length will be omitted for brevity.*

3. Symmetries and Sensitivity

To design the fault detection, isolation, and identification policy of Section 4, the following symmetries are presented and will be exploited later.

1. The arms are counter-clockwise numbered and equally distributed on the so-called "x" configuration, leading to the angles

$$\delta_i = \frac{\pi}{4} + (i-1)\frac{\pi}{2}, \qquad i = 1, \ldots, 4. \tag{14}$$

2. The overall geometry of each arm is identical to the others. In particular, the geometry constraints of each fixed arm $B_{i,1}$ and telescopic arm $B_{i,2}$ lead to

$$w_{i,j} = w_{1,j}, \qquad i = 2,3,4, \qquad j = 1,2, \tag{15}$$
$$l_{i,j} = l_{1,j}, \qquad i = 2,3,4, \qquad j = 1,2, \tag{16}$$
$$h_{i,j} = h_{1,j}, \qquad i = 2,3,4, \qquad j = 1,2, \tag{17}$$

while the geometry constraints of each motor $B_{i,3}$ and rotor $B_{i,4}$ lead to

$$r_{i,j} = r_{1,j}, \qquad i = 2,3,4, \qquad j = 3,4, \tag{18}$$
$$h_{i,j} = h_{1,j}, \qquad i = 2,3,4, \qquad j = 3,4. \tag{19}$$

3. For each arm, the mass of each component is equal to the corresponding component of the other arms, which leads to

$$m_{i,j} = m_{1,j}, \qquad i = 2,3,4, \qquad j = 1,2,3,4. \tag{20}$$

4. Motors 1 and 3 are counter-clockwise, while 2 and 4 are clockwise, hence

$$c_{w_i} = (-1)^{i+1}, \qquad i = 1,2,3,4. \tag{21}$$

Several implications follow.

5. The total mass of the system is

$$m = m_0 + 4(m_{1,1} + m_{1,2} + m_{1,3} + m_{1,4}). \tag{22}$$

6. Since $\delta_3 = \delta_1 + \pi$ and $\delta_4 = \delta_2 + \pi$, for each vector $v \in \operatorname{span}\{e_1, e_2\}$ we have

$$R^B_{3,j} v = -R^B_{1,j} v, \qquad j = 1,2,3,4, \tag{23}$$
$$R^B_{4,j} v = -R^B_{2,j} v, \qquad j = 1,2,3,4. \tag{24}$$

Remark 9. *The center of mass r^B_G is linear in l. In particular, we have $r^B_G = K_G l$, where*

$$K_G = \frac{\sqrt{2}}{2}\frac{m_{1,2} + m_{1,3} + m_{1,4}}{m}\begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{25}$$

See Appendix A.1 for details.

Remark 10. *The actuation torque M_m^B is affine in l, that is, it can be rewritten as $M_m^B = v(u) + K_m(u)l$, where $v(u) \in \mathbb{R}^3$ and $K_m(u) \in \mathbb{R}^{3\times 4}$. Moreover, $v(u)$ is linear in u and it can be rewritten as $v(u) = K_v u$, where*

$$K_v = \begin{bmatrix} -\frac{l_0}{2} & -\frac{l_0}{2} & \frac{l_0}{2} & \frac{l_0}{2} \\ \frac{w_0}{2} & -\frac{w_0}{2} & -\frac{w_0}{2} & \frac{w_0}{2} \\ \frac{c_D}{c_L} & -\frac{c_D}{c_L} & \frac{c_D}{c_L} & -\frac{c_D}{c_L} \end{bmatrix}. \tag{26}$$

Finally, the i-th column of $K_m(u)$, denoted as $K_m^i(u)$, is linear in u, that is $K_m^i(u) = D_i u$, where

$$D_i = R_{i,1}^B E_{2,i} - \frac{m_{1,2} + m_{1,3} + m_{1,4}}{m} R_{i,1}^B (E_{2,1} + E_{2,2} + E_{2,3} + E_{2,4}), \tag{27}$$

where $E_{i,j} \in \mathbb{R}^{3\times 4}$ is defined as a matrix with zero elements but the i, j element which is equal to "1". See Appendix A.2 for details.

Remark 11. *Defining $\bar{m} = \frac{m_{1,2}+m_{1,3}+m_{1,4}}{m}$, it is possible to rewrite D_1, D_2, D_3, D_4 in a matrix fashion as*

$$D_1 = R_{1,1}^B \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1-\bar{m} & -\bar{m} & -\bar{m} & -\bar{m} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad D_2 = R_{2,1}^B \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\bar{m} & 1-\bar{m} & -\bar{m} & -\bar{m} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{28}$$

$$D_3 = R_{3,1}^B \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\bar{m} & -\bar{m} & 1-\bar{m} & -\bar{m} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad D_4 = R_{4,1}^B \begin{bmatrix} 0 & 0 & 0 & 0 \\ -\bar{m} & -\bar{m} & -\bar{m} & 1-\bar{m} \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{29}$$

Remark 12. *The control moment input matrix F_2 can be rewritten as*

$$F_2 = K_v + l_1 D_1 + \ldots + l_4 D_4. \tag{30}$$

Indeed, we have $M_m^B = F_2 u$, and, at the same time,

$$M_m^B = K_v u + K_m(u) l \tag{31}$$

$$= K_v u + \sum_{i=1}^{4} (l_i D_i u) \tag{32}$$

$$= (K_v + \sum_{i=1}^{4} (l_i D_i)) u. \tag{33}$$

Remark 13. *Consider the index*

$$\frac{||J(l_n) - J(l)||}{||J(l)||}, \tag{34}$$

where the demanded arm lengths l_n differ from the actual ones for one component only. Each arm is demanded a length of α i.e., $l_n = col(\alpha, \alpha, \alpha, \alpha) \triangleq \bar{\alpha}$, and the i-th arm is considered to have a stuck displacement s_{o_i}. Therefore, we consider the indexes

$$r_{i,\alpha}(s_{oi}) = \frac{||J(\bar{\alpha}) - J(\bar{\alpha} + e_i s_{oi})||}{||J(\bar{\alpha} + e_i s_{oi})||}, \qquad i = 1, \ldots, 4. \tag{35}$$

Figure 2 shows the numerical evaluation of the index $r_{1,\alpha}$. The indexes $r_{2,\alpha}, r_{3,\alpha}, r_{4,\alpha}$ are identical due to the symmetry of the system.

For each index, the evaluation is performed for $\alpha = 0.2$ (m), $\alpha = 0.3$ (m), and $\alpha = 0.4$ (m), which represent the (minimum, average, and maximum) lengths of the arms of the vehicle used in the numerical simulations, as described in Table 1. In the worst case, that is, when the stuck offset is 0.2 (m), the variation of $||J(l)||$ is up to 30%. It is, however, reasonable to fix the arm set point in *the middle range ($\alpha = 0.3$ (m)), leading to a variation up to 15%. Thus, for the rest of the paper, the design is performed assuming $J(l) \approx J(l_n)$, which will be, therefore, simply written as J.*

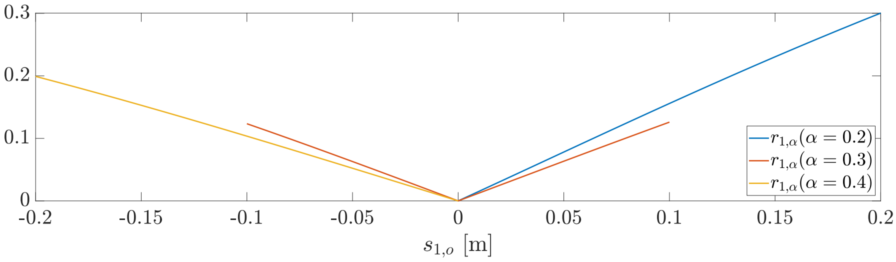

Figure 2. Sensitivity indexes for $J(l)$. The calculation is performed with the numerical parameters reported in Section 6.

4. Fault Detection, Isolation, and Identification

4.1. Fault Detection

Following [22], we approach the problem in the following way.

Definition 1. *A (fault detection) residual generator is a dynamic system with the structure*

$$\begin{aligned} \dot{z} &= f(z, u, p_F, \dot{p}_F, \omega, \eta), \\ r &= h(z, u, p_F, \dot{p}_F, \omega, \eta), \end{aligned} \tag{36}$$

where $z(t) \in \mathbb{R}^{n_z}$ and $r(t) \in \mathbb{R}^{n_r}$, such that

1. *whenever no fault occurs, $r(t)$ exhibits convergent dynamics;*
2. *whenever a fault occurs, at least one component of $r(t)$ does not exhibit convergent dynamics.*

If (36) is a (fault detection) residual generator, then $r(t)$ is an (observer based) fault detection residue.

Remark 14. *In the common definitions, the fault detection residue $r(t)$ is a scalar one, that is, $r(t) \in \mathbb{R}$ for each t. This condition can be matched by considering the new residue $||r(t)||$.*

Consider the dynamical system

$$\Sigma_{FD} : \quad \begin{aligned} &\dot{z} = -k_r J^{-1}\omega - J^{-1}(\omega \times J\omega) + J^{-1}v(u) + J^{-1}K_m(u)l_n + H(\omega - z) \\ &r = \omega - z, \end{aligned} \tag{37}$$

where $-H \in \mathbb{R}^{3\times3}$ is a Hurwitz matrix.

Proposition 1. *If $u(t) \notin \ker(D_i)$ for $i = 1, \ldots, 4$, then Σ_{FD} is a (fault detection) residual generator.*

Proof. The residue dynamics are

$$\dot{r} = -Hr + J^{-1}K_m s_o. \tag{38}$$

If no telescopic arm is stuck, then $s_o = 0$, the residue r exhibits asymptotically stable dynamics. Let the j-th arm be the stuck one, i.e., let $s_{oj}(t) \neq 0$ and $s_{oi}(t) = 0$ for $i \neq j$. The residue dynamics are then

$$\begin{aligned} \dot{r} &= -Hr + (J^{-1}K_m e_j)s_{oj} && (39) \\ &= -Hr + (J^{-1}D_j u)s_{oj}, && (40) \end{aligned}$$

and because $(J^{-1}D_j u)s_{oj} \neq 0$, $r(t)$ does not converge to the origin. □

4.2. Fault Isolation

Following again a simplified version of [22], we approach the fault isolation problem as follows. Let $\mathcal{C}_1, \dots, \mathcal{C}_k$ be fault classes to isolate, and let us denote with $\mathcal{C}_0$ the absence of faults. We assume in every instant of time one $\mathcal{C}_i$ is active for some unknown i, while $\mathcal{C}_j$ is not active for every $j \neq i$.

Definition 2. *A (fault isolation) residual generator for $\mathcal{C}_1, \dots, \mathcal{C}_k$ is a dynamical system with the structure*

$$\begin{aligned} \dot{z} &= f(z, u, p_F, \dot{p}_F, \omega, \eta), \\ r_i &= h_i(z, u, p_F, \dot{p}_F, \omega, \eta), \qquad i = 1, \dots, k, \end{aligned} \tag{41}$$

where $z(t) \in \mathbb{R}^{n_z}$ and $r_i(t) \in \mathbb{R}$, such that

1. *whenever $\mathcal{C}_0$ occurs, $r_i(t)$ exhibits convergent dynamics for $i = 1, \dots, k$;*
2. *whenever $\mathcal{C}_i$ occurs, for $i \in \{1, \dots, k\}$, $r_i(t)$ does not exhibit convergent dynamics, while $r_j(t)$ exhibits convergent dynamics for each $j \neq i$.*

If (41) is a (fault isolation) residual generator for $\mathcal{C}_1, \dots, \mathcal{C}_k$, then $r(t) = col(r_1(t), \dots, r_k(t))$ is a (directional observer-based) fault isolation residue.

In our purpose, we define the fault classes

$$\mathcal{C}_0 : \quad \text{if } s_{o1} = s_{o2} = s_{o3} = s_{o4} = 0, \tag{42}$$

$$\mathcal{C}_1 : \quad \text{if } s_{o1} \neq 0 \text{ or } s_{o3} \neq 0, \tag{43}$$

$$\mathcal{C}_2 : \quad \text{if } s_{o2} \neq 0 \text{ or } s_{o4} \neq 0. \tag{44}$$

Consider the dynamical system

$$\Sigma_{FDI} : \quad \begin{aligned} \dot{z}_1 &= -k_r v_1^T \omega - v_1^T(\omega \times J\omega) + v_1^T v(u) + v_1^T K_m(u) l_n + H_1(v_1^T J\omega - z_1) \\ \dot{z}_2 &= -k_r v_2^T \omega - v_2^T(\omega \times J\omega) + v_2^T v(u) + v_2^T K_m(u) l_n + H_2(v_2^T J\omega - z_2) \\ r_1 &= v_1^T J\omega - z_1 \\ r_2 &= v_2^T J\omega - z_2, \end{aligned} \tag{45}$$

where $H_1, H_2 > 0$, $v_1 = -e_1 + e_2$ and $v_2 = e_1 + e_2$.

Proposition 2. *If $u \notin \ker(v_1^T D_i)$ for $i = 1, 3$ and $u \notin \ker(v_2^T D_j)$ for $j = 2, 4$, then Σ_{FDI} is a (fault isolation) residual generator for $\mathcal{C}_1, \mathcal{C}_2$.*

Proof. We have

$$\ker(D_1^T) = \ker(D_3^T) = \text{span}\{e_1 + e_2, e_3\}, \tag{46}$$

$$\ker(D_2^T) = \ker(D_4^T) = \text{span}\{e_1 - e_2, e_3\}, \tag{47}$$

therefore, $v_1^T D_2 = v_1^T D_4 = v_2^T D_1 = v_2^T D_3 = 0$ and

$$v_1^T D_1 \neq 0 \qquad v_1^T D_3 \neq 0 \qquad v_2^T D_2 \neq 0 \qquad v_2^T D_4 \neq 0. \tag{48}$$

By differentiation, we have

$$\dot{r}_1 = -H_1 r_1 + (v_1^T D_1 u) s_{o1} + (v_1^T D_3 u) s_{o3} \tag{49}$$

$$\dot{r}_2 = -H_2 r_2 + (v_2^T D_2 u) s_{o2} + (v_1^T D_4 u) s_{o4}. \tag{50}$$

In the absence of faults, we have $s_{oi}(t) = 0$ for $i = 1, \ldots, 4$, and, therefore, both $r_1(t)$ and $r_2(t)$ exhibit asymptotically stable (mutually decoupled) dynamics. In case of a fault of class $\mathcal{C}_1$, we have $s_{oi}(t) \neq 0$ for $i \in \{1, 3\}$, and the residue dynamics are

$$\dot{r}_1 = -H_1 r_1 + (v_1^T D_i u) s_{oi} \tag{51}$$
$$\dot{r}_2 = -H_2 r_2, \tag{52}$$

hence $r_2(t)$ exhibits an asymptotically stable dynamics, while $r_1(t)$ does not. Finally, in the case of a fault of class $\mathcal{C}_2$, we have $s_{oj}(t) \neq 0$ for $j \in \{2, 4\}$, and the residue dynamics are

$$\dot{r}_1 = -H_1 r_1 \tag{53}$$
$$\dot{r}_2 = -H_2 r_2 + (v_2^T D_j u) s_{oj}, \tag{54}$$

hence $r_1(t)$ exhibits an asymptotically stable dynamics, while $r_2(t)$ does not. □

Remark 15. *The fault signature matrix provided by Proposition (2) is summarized in Table 1. In practice, the isolation is provided through a threshold-based policy.*

Table 1. Fault signature matrix of the fault classes $\mathcal{C}_1, \mathcal{C}_2$ provided by Σ_{FDI}. The isolation is achieved through the evaluation of the non-zero value of $r_1(t), r_2(t)$.

	$\mathcal{C}_0$	$\mathcal{C}_1$	$\mathcal{C}_2$
$r_1(t)$	0	$\neq 0$	0
$r_2(t)$	0	0	$\neq 0$

Remark 16. *Without a specific control input, the filter Σ_{FDI} is not able to isolate which arm is stuck, but just to isolate the pairs $(1, 3)$ and $(2, 4)$. This is a consequence of the symmetries of the system, and the motivation for achieving active isolation in the following. The diagnostic information provided by $r_1(t)$ and $r_2(t)$ can, however, lead to finer isolation.*

Let us split $\mathcal{C}_1$ and $\mathcal{C}_2$ into two subclasses each, considering that the overall fault classes

$$\mathcal{C}_0 : \quad \text{if } s_{o1} = s_{o2} = s_{o3} = s_{o4} = 0, \tag{55}$$
$$\mathcal{C}_1^+ : \quad \text{if } s_{o1} > 0 \text{ or } s_{o3} < 0, \tag{56}$$
$$\mathcal{C}_1^- : \quad \text{if } s_{o1} < 0 \text{ or } s_{o3} > 0, \tag{57}$$
$$\mathcal{C}_2^+ : \quad \text{if } s_{o2} < 0 \text{ or } s_{o4} > 0, \tag{58}$$
$$\mathcal{C}_2^- : \quad \text{if } s_{o2} > 0 \text{ or } s_{o4} < 0. \tag{59}$$

The residue generator Σ_{FDI} can be modified for the isolation of the fault classes $\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$. To this end, consider the function $ramp : \mathbb{R} \to \mathbb{R}$ defined as

$$ramp(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0. \end{cases} \tag{60}$$

Note that the function $ramp(x)$ provides the absolute value of the arguments x whenever x is positive, while the function $ramp(-x)$ provides the absolute value for $x < 0$. From now on, we will consider the following assumption.

Assumption 1. *The inequalities*

$$f_i(t) - \bar{m}(f_1(t) + f_2(t) + f_3(t) + f_4(t)) \geq \gamma_i, \qquad i = 1, 2, 3, 4, \tag{61}$$

are satisfied for each t and for some $\gamma_1, \gamma_2, \gamma_3, \gamma_4 > 0$.

Remark 17. *In practice, we have $f_i(t) \geq 0$ for each i. Then, noting that $f_1(t) + f_2(t) + f_3(t) + f_4(t)$ is the upward lift force at time t, Assumption (61) bounds the displacement between the $f_i(t)$s. Thus, it bounds the motor torques, which is coherent with the saturation, as well as with the fact that during the flight the quadrotor exhibits relatively small angles. Additionally, because $f_i(t) \neq 0$ for each i, strictly positive control inputs are implied.*

Lemma 1. *Consider Σ_{FDI} as in Proposition (2) and let $\mathcal{C}$ be the actual fault class.*

1. *If $\mathcal{C} = \mathcal{C}_0$, then $r_1(t)$ and $r_2(t)$ exhibits asymptotically stable dynamics.*
2. *If $\mathcal{C} = \mathcal{C}_1^+$, then $r_2(t)$ exhibits asymptotically stable dynamics, while $r_1(t) > 0$ is achieved in finite time.*
3. *If $\mathcal{C} = \mathcal{C}_1^-$, then $r_2(t)$ exhibits asymptotically stable dynamics, while $r_1(t) < 0$ is achieved in finite time.*
4. *If $\mathcal{C} = \mathcal{C}_2^+$, then $r_1(t)$ exhibits asymptotically stable dynamics, while $r_2(t) > 0$ is achieved in finite time.*
5. *If $\mathcal{C} = \mathcal{C}_2^-$, then $r_1(t)$ exhibits asymptotically stable dynamics, while $r_2(t) < 0$ is achieved in finite time.*

Proof. Point 1 has been proven in Proposition (2), so let us discuss point 2. Direct calculations lead to

$$v_1^T D_1 u(t) = \sqrt{2}(f_1(t) - \bar{m}(f_1(t) + f_2(t) + f_3(t) + f_4(t))) \geq \sqrt{2}\gamma_1, \tag{62}$$

$$v_1^T D_3 u(t) = -\sqrt{2}(f_3(t) - \bar{m}(f_1(t) + f_2(t) + f_3(t) + f_4(t))) \leq -\sqrt{2}\gamma_3, \tag{63}$$

$$v_2^T D_2 u(t) = -\sqrt{2}(f_2(t) - \bar{m}(f_1(t) + f_2(t) + f_3(t) + f_4(t))) \leq -\sqrt{2}\gamma_2, \tag{64}$$

$$v_2^T D_4 u(t) = \sqrt{2}(f_4(t) - \bar{m}(f_1(t) + f_2(t) + f_3(t) + f_4(t))) \geq \sqrt{2}\gamma_4. \tag{65}$$

Since $\mathcal{C}_1^+$ is a sub-case of $\mathcal{C}_1$, we have already proved that $r_2(t)$ converges to zero. Let $s_{o1}(t)$ be strictly positive, that is, let $s_{o1}(t) \geq s_{min} > 0$. The rate of $r_1(t)$ is

$$\dot{r}_1 = -H_1 r_1 + (v_1^T D_1 u) s_{o1} \geq -H_1 r_1 + \sqrt{2}\gamma_1 s_{min}, \tag{66}$$

implying

$$r_1(t) \geq \left(r_1(0) - \frac{\sqrt{2}\gamma_1 s_{min}}{H_1} \right) e^{-H_1 t} + \frac{\sqrt{2}\gamma_1 s_{min}}{H_1}, \tag{67}$$

which becomes strictly positive in finite time, depending on $r_1(0)$ and H_1. Let us now explore the remaining possibility for $\mathcal{C}_1^+$, that is, $s_{o3}(t) \leq -s_{max}$, for some $s_{max} > 0$. The rate of $r_1(t)$ is

$$\dot{r}_1 = -H_1 r_1 + (v_1^T D_3 u) s_{o3} \leq -H_1 r_1 + \sqrt{2}\gamma_3 s_{max}, \tag{68}$$

implying

$$r_1(t) \geq \left(r_1(0) - \frac{\sqrt{2}\gamma_3 s_{max}}{H_1} \right) e^{-H_1 t} + \frac{\sqrt{2}\gamma_3 s_{max}}{H_1}, \tag{69}$$

which becomes strictly positive in finite time, depending on $r_1(0)$ and H_1. The remaining points proceed identically. □

Proposition 3. *Consider $u(t) \notin \ker(v_1^T D_i)$ for $i = 1, 3$ and $u(t) \notin \ker(v_2^T D_j)$ for $j = 2, 4$. Then, the system Σ_{FDI} together with the new output functions*

$$r_1^+ = ramp(v_1^T J\omega - z_1) \tag{70}$$

$$r_1^- = ramp(-v_1^T J\omega + z_1) \tag{71}$$

$$r_2^+ = ramp(v_2^T J\omega - z_2) \tag{72}$$

$$r_2^- = ramp(-v_2^T J\omega + z_2) \tag{73}$$

is a (fault isolation) residual generator for $\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$.

Proof. Just merge Lemma (1) with the fact that

$$r_1^+(t) = ramp(r_1(t)), \qquad r_1^-(t) = ramp(-r_1(t)), \tag{74}$$

$$r_2^+(t) = ramp(r_2(t)), \qquad r_2^-(t) = ramp(-r_2(t)). \tag{75}$$

□

Remark 18. *The fault signature matrix provided by Lemma (1) and Proposition (3) is summarized in Table 2. In practice, the isolation is provided through a threshold-based policy.*

Table 2. Fault signature matrix of the fault classes $\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$ provided by Σ_{FDI}. The first part of the table refers to the evaluation through the sign of $r_1(t), r_2(t)$, while the lower part refers to the evaluation through the non-zero value of $r_1^+(t), r_1^-(t), r_2^+(t), r_2^-(t)$.

	$\mathcal{C}_0$	$\mathcal{C}_1^+$	$\mathcal{C}_1^-$	$\mathcal{C}_2^+$	$\mathcal{C}_2^-$
$r_1(t)$	0	>0	<0	0	0
$r_2(t)$	0	0	0	>0	<0
$r_1^+(t)$	0	$\neq 0$	0	0	0
$r_1^-(t)$	0	0	$\neq 0$	0	0
$r_2^+(t)$	0	0	0	$\neq 0$	0
$r_2^-(t)$	0	0	0	0	$\neq 0$

4.3. Active Fault Isolation and Identification

Based on the residue evaluation, the isolation and identification are here discussed. The isolation is undertaken through an active approach. Fault isolation is said active if the isolation is provided through a restriction of the control input behavior, i.e., the control input is constrained to a subclass of signals. In the study case, the overall control inputs are the lift forces $u = \text{col}(f_1, f_2, f_3, f_4)$ and the nominal arm lengths $l_n = \text{col}(l_{n1}, l_{n2}, l_{n3}, l_{n4})$. Because the overall system stability is strongly affected by forces, the input restriction is inspected for l_n only.

Lemma 2. *Let $\mathcal{C}_1^+$ be the actual fault class.*

1. *If $s_{o1}(t) > 0$ and $l_{n1}(t), l_{n3}(t)$ are strictly increasing, then $r_1^+(t)$ becomes strictly decreasing after a finite time.*
2. *If $s_{o3}(t) < 0$ and $l_{n1}(t), l_{n3}(t)$ are strictly decreasing, then $r_1^+(t)$ becomes strictly decreasing after a finite time.*

Proof. We discuss the first point. Let $s_{o1}(t) > 0$ and $\dot{l}_{n1}(t), \dot{l}_{n3}(t) \geq d_{min} > 0$. The stuck arm is the first, and $\dot{s}_{01}(t) = \dot{l}_1(t) - \dot{l}_{n1}(t) = -\dot{l}_{n1}(t) \leq -d_{min}$. Then

$$s_{01}(t) = s_{01}(t_0) + \int_{t_0}^{t} \dot{s}_{01}(\tau) d\tau \leq s_{01}(t_0) - d_{min}(t - t_0), \tag{76}$$

and

$$\begin{aligned} \dot{r}_1^+ &= -H_1 r_1^+ + (v_1^T D_1 u) s_{01} && (77)\\ &\leq +(v_1^T D_1 u) s_{01}(t_0) - d_{min}(v_1^T D_1 u)(t - t_0) && (78)\\ &\leq \sqrt{2}\gamma_1 s_{01}(t_0) - \sqrt{2} d_{min} \gamma_1 (t - t_0). && (79) \end{aligned}$$

Point 2 proceeds in the same way. □

Remark 19. *We actively set increasing/decreasing $l_{n1}(t)$ and $l_{n3}(t)$, and then the stuck arm is determined based on which motion leads to $r_1^+(t) \approx 0$. Not only the isolation is finally provided, but also an estimation of the stuck arm is provided. Indeed, when $l_{ni}(t) \approx l_i$ we have $s_{01}(t) \approx 0$, and as discussed, all the residual components convergence to zero.*

Remark 20. *The active isolation is described for the class $\mathcal{C}_1^+$. However, mirrored results hold for $\mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$ due to the system symmetries, following the same steps.*

Then, the supervisor can be described as a discrete event system, where the nominal arm lengths $l_n(t)$ represents the output, and it depends on the system state. The set of the supervisor states is

$$\mathcal{S} = \{S_0, S_1, S_2, S_3, S_4, S_{13}^{++}, S_{13}^{+-}, S_{13}^{-+}, S_{13}^{--}, S_{24}^{++}, S_{24}^{+-}, S_{24}^{-+}, S_{24}^{--}\}, \tag{80}$$

where:

- S_0: no fault is detected or isolated. The output is $l_n(t) = l^\star(t)$;
- S_1, S_2, S_3, S_4: the i-th arm has been isolated (respectively, for some $i = 1,2,3,4$) and the corresponding stuck fault is estimated as $\bar{l}$. Denoting the stuck arm with i, the output is $l_{nj}(t) = l_j^\star(t)$ for $j \neq i$ and $l_{ni}(t) = \bar{l}_i$;
- S_{13}^{++} and S_{13}^{+-}: the fault class $\mathcal{C}_1^+$ for the first or $\mathcal{C}_1^-$ for the second has been isolated (as described by the second "+" and "−" signs, respectively). Denoting the switching time with t_0, the supervisor overrides $l_{n1}^\star(t)$ and $l_{n3}^\star(t)$ with the increasing commands (as described by the first "+" sign in both system states)

$$l_{ni}(t) = l_{ni}(t_0) + \frac{l_{max} - l_{ni}(t_0)}{T}(ramp(t - t_0) - ramp(t - t_0 - T)), \quad i = 1, 3; \tag{81}$$

- S_{13}^{-+} and S_{13}^{--}: the fault class $\mathcal{C}_1^+$ (for the first) or $\mathcal{C}_1^-$ (for the second) has been isolated. Denoting the switching time with t_0, the supervisor overrides $l_{n1}^\star(t)$ and $l_{n3}^\star(t)$ with the decreasing commands

$$l_{ni}(t) = l_{ni}(t_0) + \frac{l_{min} - l_{ni}(t_0)}{T}(ramp(t - t_0) - ramp(t - t_0 - T)), \quad i = 1, 3; \tag{82}$$

- S_{24}^{++} and S_{24}^{+-}: the fault class $\mathcal{C}_2^+$ (for the first) or $\mathcal{C}_2^-$ (for the second) has been isolated. Denoting with t_0 the switching time, the supervisor overrides $l_{n2}^\star(t)$ and $l_{n3}^\star(t)$ with the increasing commands

$$l_{ni}(t) = l_{ni}(t_0) + \frac{l_{max} - l_{ni}(t_0)}{T}(ramp(t - t_0) - ramp(t - t_0 - T)), \quad i = 2, 4; \tag{83}$$

- S_{24}^{-+} and S_{24}^{--}: the fault class $\mathcal{C}_2^+$ (for the first) or $\mathcal{C}_2^-$ (for the second) has been isolated. Denoting the switching time as t_0, the supervisor overrides $l_{n2}^\star(t)$ and $l_{n4}^\star(t)$ with the decreasing commands

$$l_{ni}(t) = l_{ni}(t_0) + \frac{l_{min} - l_{ni}(t_0)}{T}(ramp(t - t_0) - ramp(t - t_0 - T)), \quad i = 2, 4; \tag{84}$$

The set of events is

$$E = \{\mathcal{C}_0, \mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-, l_{13}^{max}, l_{13}^{min}, l_{24}^{max}, l_{24}^{min}\}, \tag{85}$$

where:

- $\mathcal{C}_0$: no fault class is currently isolated from the Σ_{FDI};
- $\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$: the corresponding fault class has been isolated from the Σ_{FDI};
- l_{13}^{max} and l_{13}^{min}: $l_{n_1}(t) = l_{n_3}(t) = l_{max}$ is achieved for the former, while $l_{n_1}(t) = l_{n_3}(t) = l_{min}$ for the latter;

- l_{24}^{max} and l_{24}^{min}: $l_{n_2}(t) = l_{n_4}(t) = l_{max}$ is achieved for the former, while $l_{n_2}(t) = l_{n_4}(t) = l_{min}$ for the latter.

  The graphical supervisor transition map is reported in Figure 3.

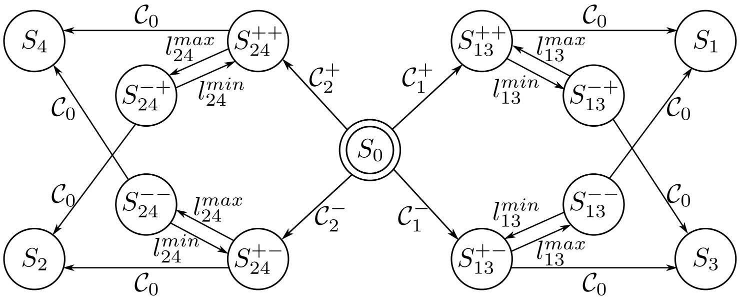

Figure 3. Supervisor. The active fault isolation is undertaken during $S_{13}^{++}, S_{13}^{+-}, S_{13}^{-+}, S_{13}^{--}, S_{24}^{++}, S_{24}^{+-}, S_{24}^{-+}, S_{24}^{--}$. The events which triggers the active fault isolation are $\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$, while the only event which ends it is $\mathcal{C}_0$.

Remark 21. *The override signals let the related reference $l_{ni}(t)$ be piece-wise continuous, with piece-wise constant derivative (i.e., piece-wise constant speed). Moreover, $l_{ni}(t+T) = l_{max}$ (resp. $l_{ni}(t+T) = l_{min}$) is achieved in case of increasing (resp. decreasing) signal.*

Remark 22. *The isolation is ideally achieved by the evaluation of the non-zero values of $r_1^+(t)$, $r_1^-(t)$, $r_2^+(t)$, $r_2^-(t)$. However, for robustness issues, a threshold-based evaluation is considered. More precisely, two thresholds are actually set, a first one denoted $t_{h_1} > 0$ is considered during S_0, S_1, S_2, S_3, S_4 (i.e., when no active isolation is achieved), while a second threshold $t_{h_2} > 0$ is considered for the remaining states (i.e., during the active isolation). In order to avoid high-frequency triggering events, we set $t_{h_1} > t_{h_2}$. The difference $t_{h_1} - t_{h_2}$, together with t_{h_1} and t_{h_2} themselves is a trade off between the robustness, the false negative/positive triggering events, and final estimation error of the stuck arm.*

Remark 23. *The supervisor should be able to trigger whenever an event occurs. However, due to the noise, the residuals $r_1^+(t), r_1^-(t), r_2^+(t), r_2^-(t)$ can pass the thresholds (up and down) with high frequency, causing high rate triggering events. Then, once a transition to a new state is achieved, a minimum time $t_{min} > 0$ is waited before the evaluation of any event. Formally, we could model this by adding a new (temporized) condition*

$$ok_{t_{min}} = \begin{cases} true & \text{if the current state is active for at least } t_{min} \text{ seconds,} \\ false & \text{otherwise.} \end{cases} \tag{86}$$

Then, each event $e \in E$ is replaced with the new one

$$e' = e \text{ and } ok_{t_{min}}. \tag{87}$$

Remark 24. *Figure 4 represents the graphical point of view about the active isolation and stuck estimation. Based on the isolated fault class ($\mathcal{C}_1^+, \mathcal{C}_1^-, \mathcal{C}_2^+, \mathcal{C}_2^-$), the supervisor starts to provide increasing or decreasing nominal arm lengths of the involved pair. The procedure stops when all the residual components are under the threshold. As it can be seen in Figure 3, livelocks are, however, possible. Then, a correct evaluation of the thresholds and the residual generator gains H_1, H_2, H_3, H_4 is needed.*

Remark 25. *The overall active isolation procedure does not affect $u(t)$ (directly). Then, there is no need to stop the system motion or force hovering.*

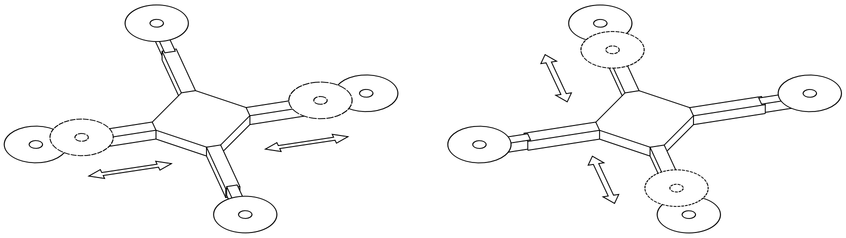

Figure 4. During the active fault isolation (and estimation), both the arms in the isolated axis are increased/decreased.

5. Fault-Tolerant DOBC Control

The feedback linearization fault-tolerant tracking controller is based on [25], where a DOBC approach is adopted for wind estimation and compensation [21].

Remark 26 (Disturbance observer)**.** *Let $\tilde{F}_w^E(t) = F_w^E(t) - \hat{F}_w^E(t)$ be the estimation error. The disturbance observer*

$$\Sigma_{DO}: \quad \begin{aligned} \dot{s} &= -Hs - H(\Phi + H\dot{p}_F) \\ \hat{F}_w^E &= ms + mH\dot{p}_F, \end{aligned} \tag{88}$$

where $-H \in \mathbb{R}^{3\times 3}$ is a Hurwitz matrix and

$$\Phi = -R_B^E S(r_G^B) J^{-1} S(\omega) J\omega - R_B^E S^2(\omega) r_G^B + ge_3 + \frac{1}{m} f_z R_B^E e_3 + R_B^E S(r_G^B) J^{-1} M_m^B, \tag{89}$$

makes the estimation error dynamics

$$\dot{\tilde{F}}_w^E = -H\tilde{F}_w^E + \dot{F}_w^E, \tag{90}$$

which is a bounded input bounded output linear time-invariant system (with respect to $\dot{F}_w^E = 0$) (see [25] for details).

Remark 27 (Inner loop)**.** *Let $v(t) = col(v_z(t), v_\eta(t))$ be an auxiliary input, where $v_z(t) \in \mathbb{R}$ and $v_\eta(t) \in \mathbb{R}^3$. If $\tilde{F}_w^E(t) = 0$, the regular static state feedback*

$$u = F^{-1} A^{-1} (g(\omega, \eta, l) + v) \tag{91}$$

provides the input–output decoupling between (z_F, η) and (v_z, v_η), where

$$g(\omega, \eta, l) = \begin{bmatrix} -e_3^T(-R_B^E S(r_G^B) J^{-1} S(\omega) J\omega - R_B^E S^2(\omega) r_G^B + ge_3) - \frac{1}{m} e_3^T \hat{F}_w^E \\ -(\dot{T}(\eta)\omega - T(\eta) J^{-1} S(\omega) J\omega) \end{bmatrix}, \tag{92}$$

$$A = \begin{bmatrix} -\frac{1}{m} e_3^T R_B^E e_3 & e_3^T R_B^E S(r_G^B) J^{-1} \\ 0 & T(\eta) J^{-1} \end{bmatrix}. \tag{93}$$

Moreover, let $z_{Fr}(t) \in \mathbb{R}$ and $\eta_r(t) = \mathrm{col}(\varphi_r(t), \theta_r(t), \psi_r(t)) \in \mathbb{R}^3$ be the time varying reference signals for $z_F(t)$ and $\eta(t)$. The controls

$$v_z = \ddot{z}_{Fr} - \alpha_{z1}(\dot{z}_F - \dot{z}_{Fr}) - \alpha_{z0}(z_F - z_{Fr}), \tag{94}$$

$$v_\eta = \ddot{\eta}_r - \alpha_{\eta 1}(T(\eta)\omega - \dot{\eta}_r) - \alpha_{\eta 0}(\eta - \eta_r), \tag{95}$$

make the overall closed-loop error system bounded input bounded output with respect to $\dot{F}_w^E$ whenever the coefficients $\alpha_{z1}, \alpha_{z0}, \alpha_{\eta 1}, \alpha_{\eta 0}$ solve the pole placement (see [25] for details).

Remark 28 (Outer loop)**.** *Let us assume near hovering conditions (i.e., $\varphi(t) \approx \theta(t) \approx 0$) and let $e_x(t) = x_F(t) - x_{Fr}(t)$ and $e_y(t) = y_F(t) - y_{Fr}(t)$ be the error variables between $x_F(t)$ and $y_F(t)$ and their references $x_{Fr}(t)$ and $y_{Fr}(t)$. The control law*

$$\begin{bmatrix} \varphi_r \\ \theta_r \end{bmatrix} = \frac{m}{f_z} \begin{bmatrix} s_\psi & -c_\psi \\ c_\psi & s_\psi \end{bmatrix} \left(- \begin{bmatrix} e_1^T \\ e_2^T \end{bmatrix} (\Gamma + \frac{1}{m} \hat{F}_w^E) + \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right), \tag{96}$$

where

$$\Gamma = -R_B^E S(r_G^B) J^{-1} S(\omega) J\omega - R_B^E S^2(\omega) r_G^B + g e_3 + R_B^E S(r_G^B) J^{-1} M_m^B, \tag{97}$$
$$\tau_x = \ddot{x}_{Fr} - \alpha_{x1}(\dot{x}_F - \dot{x}_{Fr}) - \alpha_{x0}(x_F - x_F), \tag{98}$$
$$\tau_y = \ddot{y}_{Fr} - \alpha_{y1}(\dot{y}_F - \dot{y}_{Fr}) - \alpha_{y0}(y_F - y_F), \tag{99}$$

forces the tracking error dynamics

$$\ddot{e}_x + \alpha_{x1}\dot{e}_x + \alpha_{x0} e_x \approx e_1^T \tilde{F}_w \tag{100}$$
$$\ddot{e}_y + \alpha_{y1}\dot{e}_y + \alpha_{y0} e_y \approx e_2^T \tilde{F}_w. \tag{101}$$

The coefficients $\alpha_{x1}, \alpha_{x0}, \alpha_{y1}, \alpha_{y0}$ are then chosen according to a pole placement (see [25] for details).

The overall control scheme is reported in Figure 5.

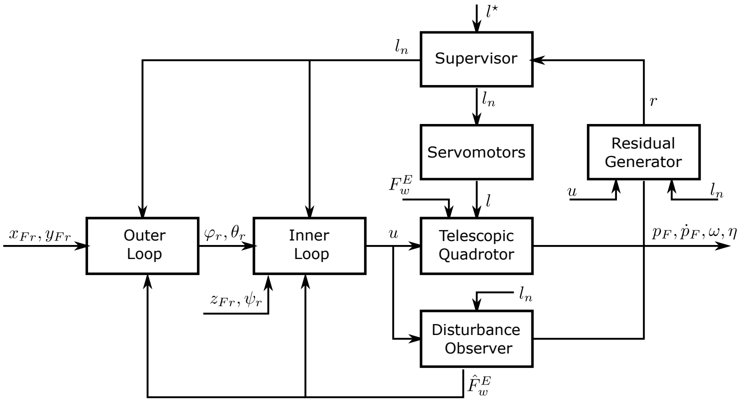

Figure 5. Control scheme. The vector $l^\star$ describes the optimal/desired arm lengths provided by external software (e.g., an onboard artificial intelligence or a planning algorithm). Based on the residue $r = \mathrm{col}(r_1^+, r_1^-, r_2^+, r_2^-)$, the supervisor detects/isolates/identificates a fault ($\bar{l}_i$), and eventually overrides $l^\star$ with a new reference l_n. Based on a stuck fault in the servomotors, the actual arm lengths l could differ from l_n.

6. Numerical Simulations

The solution is tested in simulation using MATLAB under the following settings.

Frequencies. The simulation is carried out for a total of 30 s. According to the double-loop structure, faster inner loop dynamics are achieved by both control parameters and different sample times. This solution takes into consideration computational limitations on a future implementation. In particular, the inner loop is simulated at 1 kHz, while the outer loop runs at 10 Hz. A zero order hold is applied for $\varphi_r(t)$ and $\theta_r(t)$ between consecutive samples.

Plant parameters. The frame and fixed arms are those of a DJI Flamewheel 450, the telescopic arms are of the same size and mass as the fixed ones, and the electric motors are the T-Motor AirGear 350. Table 3 summarizes the geometric dimensions and the dynamic parameters.

Sensors and Inertial Measurement Unit (IMU). The commonly adopted MPU-9250 IMU [28] is taken into consideration. Additive white Gaussian noise is applied to accelerations and gyroscopes, with standard deviation $7.8480 \cdot 10^{-2}$ and $5.5192 \cdot 10^{-3}$, respectively. Finally, due to Kalman filtering, a smaller noise is assumed for simplicity on attitude, linear velocity, and linear position (i.e., $2.7596 \cdot 10^{-3}$, $1.5696 \cdot 10^{-2}$, $7.8480 \cdot 10^{-3}$, respectively).

Input saturation. A saturation $0 \leq f_i(t) \leq 9\,N$ is considered for each actual lift force. The total mass of the system is 1.448 kg, resulting in a thrust-to-weight ratio equal to 2.5.

Control parameters. All the control parameters have been heuristically set. The parameters for both the control law and the observer are summarized in Table 3.

Table 3. Geometric dimensions, dynamical parameters, and control parameters (see [29] for lift and drag coefficients).

Parameters	Variables	Value	Unit
Frame width and length	w_0, l_0	0.1	(m)
Frame height	h_0	0.5	(m)
Fixed and telescopic arm widths	$w_{i,j}$ $(i \leq 4, j \leq 2)$	0.2	(m)
Fixed and telescopic arm lengths	$l_{i,j}$ $(i \leq 4, j \leq 2)$	0.025	(m)
Motor radius	$r_{i,3}$ $(i \leq 4)$	0.012	(m)
Rotor radius	$r_{i,4}$ $(i \leq 4)$	0.12	(m)
Fixed and telescopic arms heights	$h_{i,j}$ $(i \leq 4, j \leq 2)$	0.01	(m)
Motor heights	$h_{i,3}$ $(i \leq 4)$	0.03	(m)
Rotor heights	$h_{i,4}$ $(i \leq 4)$	0.003	(m)
Minimum arm length	l_{min}	0.2	(m)
Maximum arm length	l_{max}	0.4	(m)
Frame mass	m_0	0.8	(kg)
Fixed and telescopic arm masses	$m_{i,j}$ $(i \leq 4, j \leq 2)$	0.05	(kg)
Motor masses	$m_{i,3}$ $(i \leq 4)$	0.054	(kg)
Rotor masses	$m_{i,4}$ $(i \leq 4)$	0.008	(kg)
Total system mass	m	1.448	(kg)
Gravitational acceleration	g	9.81	$(\mathrm{m/s^2})$
Linear friction coefficient	k_t	10^{-3}	(N·s/m)
Angular friction coefficient	k_r	10^{-2}	(N·s·m)
Lift coefficient	c_L	$3.13 \cdot 10^{-5}$	$(\mathrm{N{\cdot}s^2})$
Drag coefficient	c_D	$7.5 \cdot 10^{-7}$	$(\mathrm{N{\cdot}\ m{\cdot}s^2})$
Inner loop frequency		10^3	(Hz)
Outer loop frequency		10	(Hz)
Control input lower saturation		0	(N)
Control input upper saturation		9	(N)
x_F and y_F closed loop poles		$-1, -2$	$(-)$
z_F closed loop poles		$-1, -2$	$(-)$
φ and θ closed loop poles		$-10, -20$	$(-)$
ψ closed loop poles		$-1, -2$	$(-)$
Residual generator gains	H_1, H_2	10	$(-)$
Observer gain	H	$4\,I$	$(-)$
Threshold 1	t_{h_1}	10	$(-)$
Threshold 2	t_{h_2}	3	$(-)$
Minimum time for each state	t_{min}	1	(s)

In the following, four scenarios are simulated using the same tracking reference and the same control law parameters.

6.1. Scenario 1

A stuck fault on l_1 is injected at $t = 10$ s, meanwhile, all the arm lengths commands $l_i^\star(t)$ are decreasing after the fault injection. The overall demanded trajectory is an ascending helix for the first half of the simulation, and a descending helix for the second half of the simulation, as shown in Figure 6. The yaw angle is always kept at zero (i.e., $\psi_r(t) \equiv 0$).

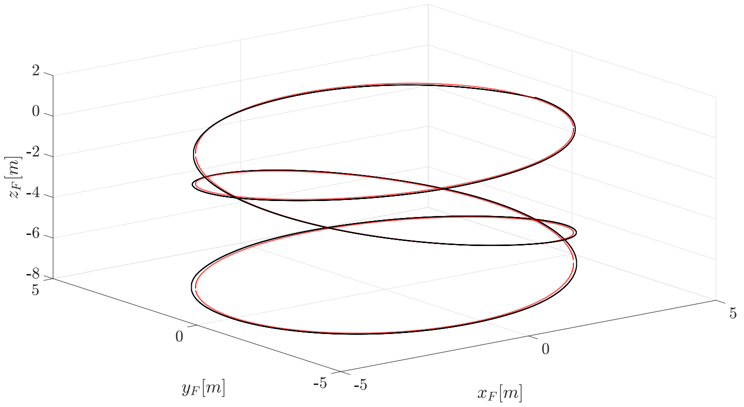

Figure 6. Trajectory of $p_F(t)$. The red plot represents the reference, while the black one represents the actual position.

The tracking performances of the linear positions and the yaw angle are reported in Figure 7. Linear position tracking errors are negligible, and the error on the yaw angle ψ is acceptable: such error magnitude is up to 0.1 rad, and this is a consequence of the chosen control law parameters.

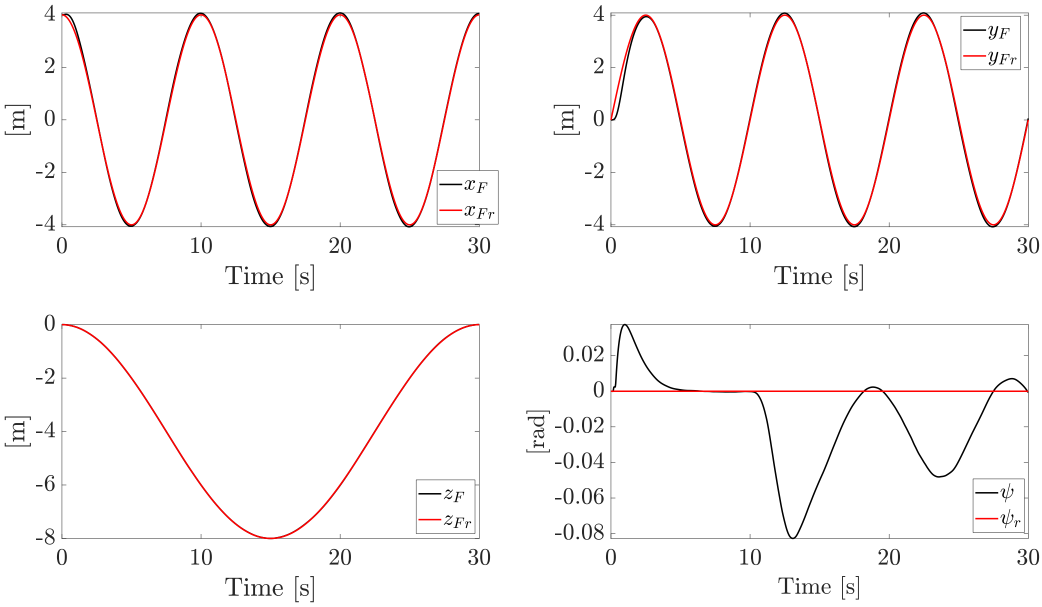

Figure 7. Tracking of $x_F(t)$, $y_F(t)$, $z_F(t)$ and $\psi(t)$.

The roll and pitch are reported in Figure 8. The effect of the zero-order hold is visible since the references are step-wise signals. The inner loop is able to achieve the goal, almost reaching the reference components before the successive sample time.

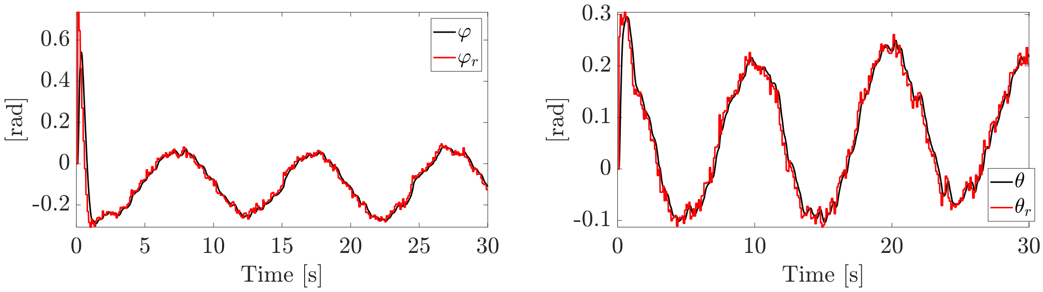

Figure 8. Tracking of $\varphi(t)$ and $\theta(t)$. The reference signals are provided by the outer loop, which runs at a lower frequency. A zero-order hold is used between consecutive samples.

The actual wind components and their estimations are reported in Figure 9. The disturbance observer is slower than the disturbance (due to the bandwidth of the observer), but it is able to provide a good estimation. Indeed, tracking is achieved with no displacement due to wind.

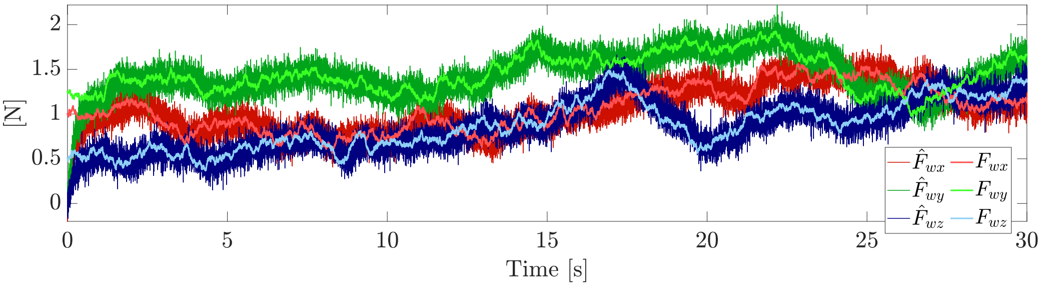

Figure 9. Wind components and their estimations.

The overall control inputs are reported in Figure 10. The saturation constraints are always matched.

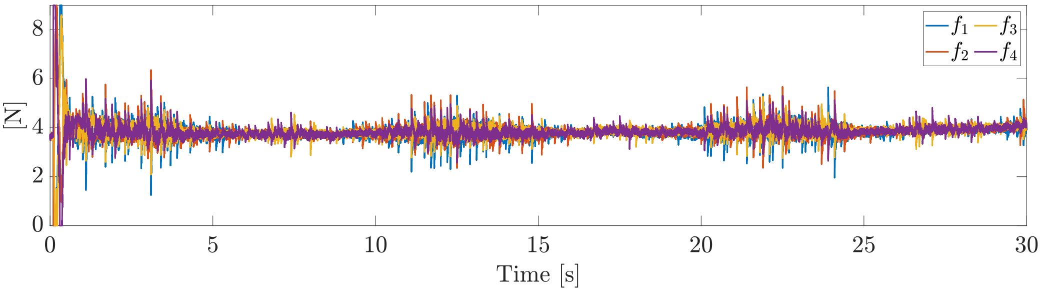

Figure 10. Control inputs.

Let us now compare the active isolation and diagnosis for the simulated scenarios. Actual arm lengths, estimations, and residues are plotted in Figure 11. Each arm starts with its maximum extension, while a shrinking phase on each arm is demanded starting from $t = 10$ s. The first arm is stuck, hence the decreasing desired arm length leads to $s_{o1}(t) > 0$, and the triggering of $r_1^+(t) > 0$. The supervisor enters the active isolation phase (state S_{13}^{++}, precisely) and overrides the arm length command for the pair $(1, 3)$. During the increase of $l_{n1}(t)$ and $l_{n3}(t)$ the residue $r_1^+(t)$ decreases (while the remaining residues are approximately null), hence when $r_1^+(t) \leq t_{h_2}$ the event $\mathcal{C}_0$ triggers. The correct isolation is then achieved, together with an estimation of the stuck $\bar{l}_1$. The isolation is quite fast since the state S_{13}^{++} directly moves the commanded signals toward the actual stuck arm length.

6.2. Scenario 2

In this scenario, each arm starts with its minimum extension, while an enlarging phase on each arm is demanded starting from $t = 10$ s. A stuck fault on l_1 is injected at $t = 10$ s. Plots about fault isolation are summarized in Figure 12, while the plots about tracking performances will be not analyzed, as they are close to those of the first scenario. The stuck leads to $r_1^-(t) > 0$ (in practice, $r_1^-(t) > t_{h_1}$), hence the supervisor enters the active fault isolation phase (state S_{13}^{+-}, precisely). The demanded arm lengths $l_{n1}^\star(t)$ and $l_{n3}^\star(t)$ are then again overrided with increasing signals. However, during this phase, the residues $r_1^-(t)$ will increase. When both $l_{n1}(t)$ and $l_{n_2}(t)$ reach the maximum length, the supervisor state transition to S_{13}^{--} happens, leading to decreasing references. Finally, as $l_{n1}(t)$ approaches to $\bar{l}_1$, the residue decreases, until $r_1^-(t) < t_{h_2}$. Then, the events $\mathcal{C}_0$ triggers, the first arm is isolated as stuck and the actual arm length is estimated.

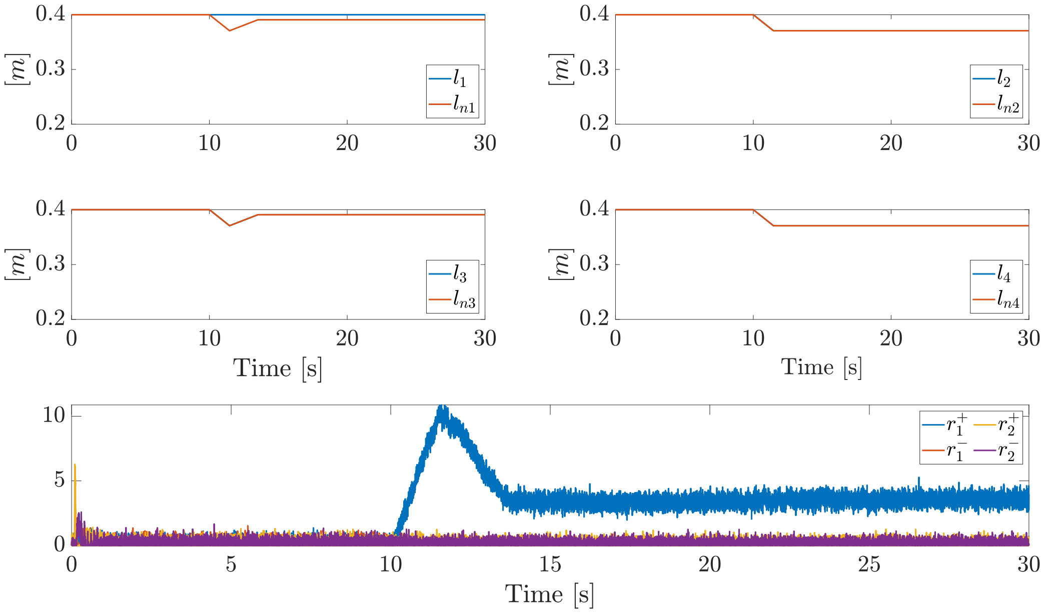

Figure 11. Actual arm lengths, nominal arm lengths, and residual components in Scenario 1.

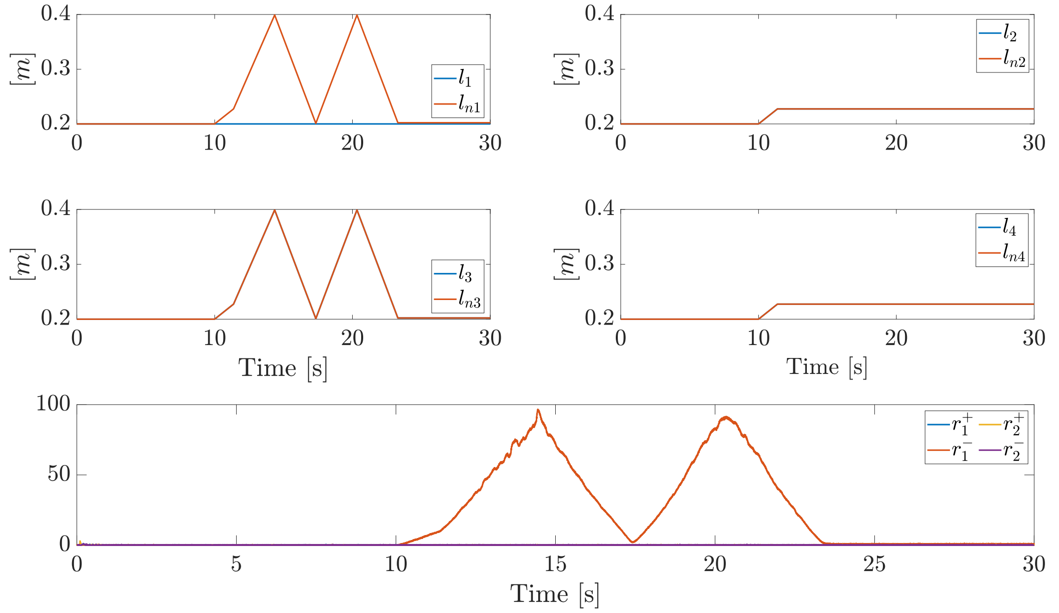

Figure 12. Actual arm lengths, nominal arm lengths, and residual components in Scenario 2.

6.3. Scenario 3

In this scenario, each arm starts with its maximum extension, while a shrinking phase on each arm is demanded starting from $t = 10$ s. A stuck fault on l_3 is injected at $t = 10$ s. From the isolation and identification point of view, this scenario is interesting: even if the stuck fault is on the third arm, the same fault class of Scenario 2 needs to be isolated by the residual generator, while the final result provided by the active fault isolation is expected to be different. Plots about fault isolation are summarized in Figure 13, while the plots about tracking performances will be not analyzed, as they are close to those of the earlier scenarios. The supervisor detects a fault and isolates the fault class $\mathcal{C}_1^+$. Differently from the Scenario 2, the supervisor policy is now an optimal one: by shrinking each arm, the triggered residue suddenly decreases, the scenario is isolated in less time, and fault isolation is achieved through the states sequence S_0, S_{13}^{+-}, S_3 (the minimum number of transitions).

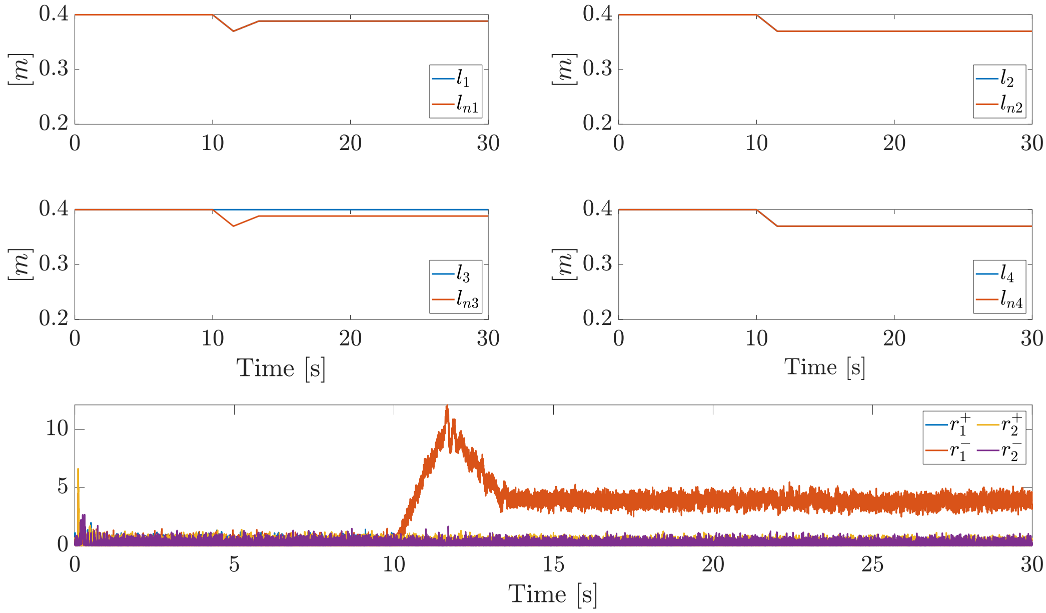

Figure 13. Actual arm lengths, nominal arm lengths, and residual components in Scenario 3.

6.4. Scenario 4

Finally, we consider a scenario where each arm starts with its minimum extension, while an enlarging phase on each arm is demanded starting from $t = 10$ s. From the isolation and identification point of view, this scenario is the symmetric counterpart of Scenario 1. Plots about fault isolation are summarized in Figure 14, while the plots about tracking performances will be not analyzed, as they are close to those of the earlier scenarios. The overall isolation is accomplished through the state sequence $S_0, S_{13}^{++}, S_{13}^{-+}, S_3$. The active policy demands an enlarging phase, which does not lead to a residue reduction: the transition to S_{13}^{-+} (shrinking phase) is required for the goal. The supervisor policy is not optimal, but isolation and identification are achieved, and no stability issues arise during the active isolation.

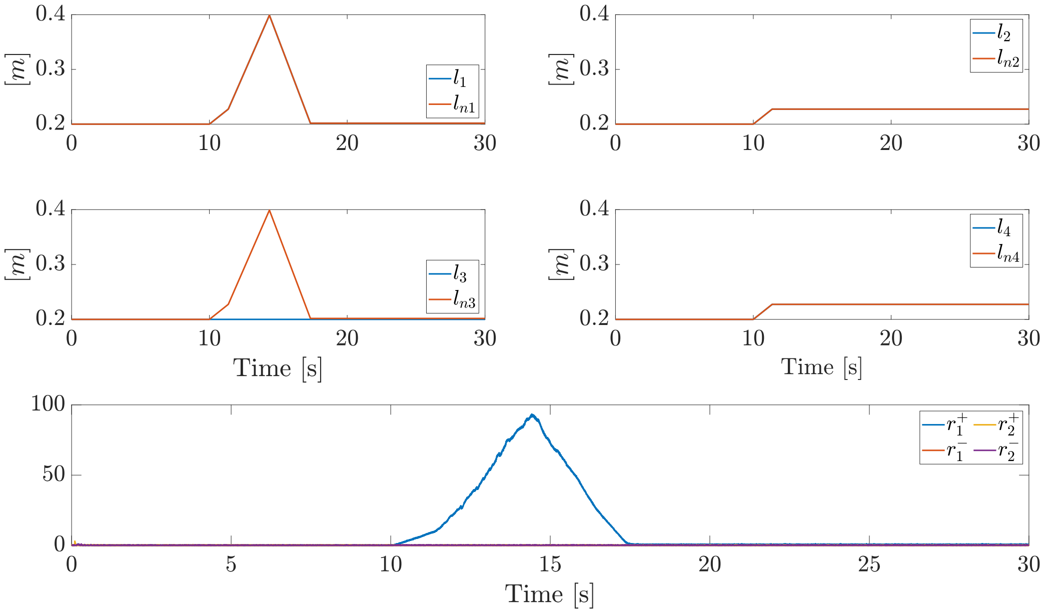

Figure 14. Actual arm lengths, nominal arm lengths, and residual components in Scenario 4.

7. Conclusions

The fault tolerant control of a telescopic quadrotor has been discussed. The fault diagnosis problem has been faced for servo motor stuck faults by exploiting the system symmetries. A residual generator provides the isolation into four fault classes, which in turn triggers an active fault isolation/identification phase, which finally solves the problem without constraining the lift forces. Indeed, such an active policy is handled by the supervisor, which has the main role of directly providing increasing/decreasing arm reference signals, ending the diagnosis. The wind is also taken into account and managed through a DOBC approach. The proposed solution is finally tested in simulation, where it can be seen how the goal is reached even under noise, actuators saturation, and model approximation.

Among the possible limitations which still affect the solution, we mention how the scheme is designed under the assumption of stuck faults only. Then, distinguishing between propeller faults and stuck servo faults is a future work, as well as extending the morphing degree of freedoms, and, finally, a hardware implementation of the solution.

Author Contributions: Conceptualization, A.B., R.F., A.F. and A.M.; methodology, A.B.; software, A.B. and R.F.; validation, A.B. and R.F.; formal analysis, A.B. and A.F.; writing—original draft preparation, A.B. and A.F.; writing—review and editing, R.F. and A.M.; visualization, A.B.; supervision, A.M.; project administration, A.F. and A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DO	Disturbance Observer
DOBC	Disturbance Observer-Based Control
FDD	Fault Detection and Diagnosis
FD	Fault Detection
FDI	Fault Detection and Isolation
FI	Fault Isolation
FId	Fault Identification
FTC	Fault Tolerant Control
NDO	Nonlinear Disturbance Observer
UAV	Unmanned Aerial Vehicle

Appendix A

Appendix A.1. Proof of Remark 9

For compactness, consider the functions

$$c_x(\delta) = \frac{\cos(\delta)}{|\cos(\delta)|}, \qquad c_y(\delta) = \frac{\sin(\delta)}{|\sin(\delta)|}, \tag{A1}$$

where $\delta \neq k\frac{\pi}{2}$. For the symmetry of the rigid body $B_0 \cup B_{1,1} \cup B_{1,2} \cup B_{1,3} \cup B_{1,4}$, we have $r^B_{1,1} = -r^B_{3,1}$, $r^B_{2,1} = -r^B_{4,1}$ and $m_{1,1} = m_{2,1} = m_{3,1} = m_{4,1}$. Moreover, $r^B_0 = 0$, hence

$$\frac{1}{m}\left(m_0 r_0^B + \sum_{i=1}^{4} m_{i,1} r_{i,1}^B\right) = 0. \tag{A2}$$

Additionally, we have $R_{1,1}^B e_1 = -R_{1,3}^B e_1$ and $R_{2,1}^B e_1 = -R_{3,3}^B e_1$. For each arm i, we have

$$r_{i,4}^B = r_{i,3}^B = c_x(\delta_i)\frac{w_0}{2}e_1 + c_y(\delta_i)\frac{l_0}{2}e_2 + l_i R_{i,1}^B e_1, \tag{A3}$$

$$r_{i,2}^B = r_{i,4}^B - \frac{w_{i,2}}{2} R_{i,1}^B e_1, \tag{A4}$$

hence

$$\begin{aligned}
\sum_{i=1}^{4} r_{i,4}^B = \sum_{i=1}^{4} r_{i,3}^B &= \sum_{i=1}^{4} c_x(\delta_i)\frac{w_0}{2}e_1 + \sum_{i=1}^{4} c_y(\delta_i)\frac{l_0}{2}e_2 + \sum_{i=1}^{4} l_i R_{i,1}^B e_1 && \text{(A5)}\\
&= \sum_{i=1}^{4} l_i R_{i,1}^B e_1, && \text{(A6)}\\
\sum_{i=1}^{4} r_{i,2}^B &= \sum_{i=1}^{4} r_{i,4}^B - \sum_{i=1}^{4} \frac{w_{i,2}}{2} R_{i,1}^B e_1 && \text{(A7)}\\
&= \sum_{i=1}^{4} r_{i,4}^B - 4\frac{w_{1,2}}{2}\sum_{i=1}^{4} R_{i,1}^B e_1 && \text{(A8)}\\
&= \sum_{i=1}^{4} r_{i,4}^B. && \text{(A9)}
\end{aligned}$$

Finally,

$$\begin{aligned}
r_G^B &= \frac{1}{m}\left(m_0 r_0^B + \sum_{i=1}^{4}\sum_{j=1}^{4} m_{i,j} r_{i,j}^B\right) && \text{(A10)}\\
&= \frac{1}{m}\left(m_0 r_0^B + \sum_{i=1}^{4} m_{i,1} r_{i,1}^B\right) + \frac{1}{m}\sum_{i=1}^{4}\sum_{j=2}^{4} m_{i,j} r_{i,j}^B && \text{(A11)}\\
&= \frac{1}{m}\left(\sum_{i=1}^{4} m_{i,2} r_{i,2}^B + \sum_{i=1}^{4} m_{i,3} r_{i,3}^B + \sum_{i=1}^{4} m_{i,4} r_{i,4}^B\right) && \text{(A12)}
\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{m}\left(m_{1,2}\sum_{i=1}^{4} r_{i,2}^B + m_{1,3}\sum_{i=1}^{4} r_{i,3}^B + m_{1,4}\sum_{i=1}^{4} r_{i,4}^B\right) && \text{(A13)}\\
&= \frac{m_{1,2}+m_{1,3}+m_{1,4}}{m}\sum_{i=1}^{4} r_{i,4}^B && \text{(A14)}\\
&= \frac{m_{1,2}+m_{1,3}+m_{1,4}}{m}\sum_{i=1}^{4} l_i R_{i,1}^B e_1 && \text{(A15)}\\
&= K_G l, && \text{(A16)}
\end{aligned}$$

where, in the last step, it is noted that K_G can be rewritten as

$$K_G = \frac{m_{1,2}+m_{1,3}+m_{1,4}}{m}\begin{bmatrix} R_{1,1}^B e_1 & R_{2,1}^B e_1 & R_{3,1}^B e_1 & R_{4,1}^B e_1 \end{bmatrix}, \tag{A17}$$

or, in a matrix fashion, as

$$K_G = \frac{m_{1,2} + m_{1,3} + m_{1,4}}{m} \begin{bmatrix} \cos(\delta_1) & \cos(\delta_2) & \cos(\delta_3) & \cos(\delta_4) \\ \sin(\delta_1) & \sin(\delta_2) & \sin(\delta_3) & \sin(\delta_4) \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{A18}$$

Appendix A.2. Proof of Remark 10

First, note that

$$(R^B_{i,1} e_1) \times e_3 = -R^B_{i,1}(e_3 \times e_1) = -R^B_{i,1} e_2. \tag{A19}$$

We have

$$-\sum_{i=1}^{4} r^B_{i,4} f_i \times e_3 = -\sum_{i=1}^{4} f_i \left(c_x(\delta_i) \frac{w_0}{2} e_1 + c_y(\delta_i) \frac{l_0}{2} e_2 \right) \times e_3 - \sum_{i=1}^{4} f_i l_i R^B_{i,1} e_1 \times e_3 \tag{A20}$$

$$= -\sum_{i=1}^{4} f_i \left(c_y(\delta_i) \frac{l_0}{2} e_1 - c_x(\delta_i) \frac{w_0}{2} e_2 \right) + \sum_{i=1}^{4} f_i l_i R^B_{i,1} e_2 \tag{A21}$$

and

$$\sum_{i=1}^{4} r^B_G \times f_i e_3 = \left(\sum_{i=1}^{4} f_i \right) r^B_G \times e_3 = -\left(\sum_{i=1}^{4} f_i \right) S_3 K_G l. \tag{A22}$$

The thrust torque is

$$M^B_{thrust} = -\sum_{i=1}^{4} \left((r^B_{i,4} - r^B_G) \times f_i e_3 \right) \tag{A23}$$

$$= -\sum_{i=1}^{4} (f_i r^B_{i,4}) \times e_3 + \sum_{i=1}^{4} r^B_G \times f_i e_3 \tag{A24}$$

$$= -\sum_{i=1}^{4} f_i \left(c_y(\delta_i) \frac{l_0}{2} e_1 - c_x(\delta_i) \frac{w_0}{2} e_2 \right) + \sum_{i=1}^{4} f_i l_i R^B_{i,1} e_2 - \left(\sum_{i=1}^{4} f_i \right) S_3 K_G l, \tag{A25}$$

and, considering that $M^B_m = M^B_{drag} + M^B_{torque}$, we have $M^B_m = v + K_m l$, where

$$v = \sum_{i=1}^{4} c_{w_i} \frac{c_D}{c_L} f_i - \sum_{i=1}^{4} f_i \left(c_y(\delta_i) \frac{l_0}{2} e_1 - c_x(\delta_i) \frac{w_0}{2} e_2 \right) \tag{A26}$$

$$= \sum_{i=1}^{4} f_i \left(c_{w_i} \frac{c_D}{c_L} e_3 - c_y(\delta_i) \frac{l_0}{2} e_1 + c_x(\delta_i) \frac{w_0}{2} e_2 \right) \tag{A27}$$

$$= K_v u, \tag{A28}$$

and

$$K_m = \begin{bmatrix} f_1 R^B_{11} e_2 & f_2 R^B_{21} e_2 & f_3 R^B_{31} e_2 & f_4 R^B_{41} e_2 \end{bmatrix} - \left(\sum_{i=1}^{4} f_i \right) S_3 K_G. \tag{A29}$$

It is indeed noted that K_v can be rewritten in a matrix fashion as

$$K_v = \begin{bmatrix} -c_y(\delta_1) \frac{l_0}{2} & -c_y(\delta_2) \frac{l_0}{2} & -c_y(\delta_3) \frac{l_0}{2} & -c_y(\delta_4) \frac{l_0}{2} \\ c_x(\delta_1) \frac{w_0}{2} & c_x(\delta_2) \frac{w_0}{2} & c_x(\delta_3) \frac{w_0}{2} & c_x(\delta_4) \frac{w_0}{2} \\ c_{w_1} \frac{c_D}{c_L} & c_{w_2} \frac{c_D}{c_L} & c_{w_3} \frac{c_D}{c_L} & c_{w_4} \frac{c_D}{c_L} \end{bmatrix}. \tag{A30}$$

For $i = 1, \ldots, 4$, we have

$$f_i R^B_{i,1} e_2 = R^B_{i,1} E_{2,1} u \tag{A31}$$

and

$$\left(\sum_{j=1}^{4} f_j\right) R_{i,1}^{B} e_2 = \left(\sum_{j=1}^{4} f_j R_{i,1}^{B} e_2\right) \tag{A32}$$

$$= \begin{bmatrix} R_{i,1}^{B} e_2 & R_{i,1}^{B} e_2 & R_{i,1}^{B} e_2 & R_{i,1}^{B} e_2 \end{bmatrix} u \tag{A33}$$

$$= R_{i,1}^{B} (E_{2,1} + E_{2,2} + E_{2,3} + E_{2,4}) u. \tag{A34}$$

Finally, the i-th column of $K_m(u)$ is

$$K_m^i(u) = f_i R_{i,1}^{B} e_2 - \frac{m_{1,2} + m_{1,3} + m_{1,4}}{m} \left(\sum_{j=1}^{4} f_j\right) S_3 R_{i,1}^{B} e_1 \tag{A35}$$

$$= f_i R_{i,1}^{B} e_2 - \frac{m_{1,2} + m_{1,3} + m_{1,4}}{m} \left(\sum_{j=1}^{4} f_j\right) R_{i,1}^{B} e_2 \tag{A36}$$

$$= R_{i,1}^{B} E_{2,i} u - \frac{m_{1,2} + m_{1,3} + m_{1,4}}{m} R_{i,1}^{B} (E_{2,1} + E_{2,2} + E_{2,3} + E_{2,4}) u \tag{A37}$$

$$= D_i u. \tag{A38}$$

References

1. Pütsep, K.; Rassõlkin, A. Methodology for Flight Controllers for Nano, Micro and Mini Drones Classification. In Proceedings of the 2021 International Conference on Engineering and Emerging Technologies (ICEET), Istanbul, Turkey, 27–28 October 2021; pp. 1–8. [CrossRef]
2. Longhi, M.; Taylor, Z.; Popović, M.; Nieto, J.; Marrocco, G.; Siegwart, R. RFID-Based Localization for Greenhouses Monitoring Using MAVs. In Proceedings of the 2018 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), Columbia, SC, USA, 10–14 September 2018; pp. 905–908. [CrossRef]
3. Cutler, M.; How, J.P. Analysis and control of a variable-pitch quadrotor for agile flight. *J. Dyn. Syst. Meas. Control* **2015**, *137*, 101002. [CrossRef]
4. Niemiec, R.; Gandhi, F.; Lopez, M.; Tischler, M. System identification and handling qualities predictions of an eVTOL urban air mobility aircraft using modern flight control methods. In Proceedings of the Vertical Flight Society 76th Annual Forum, Virtual, 5–8 October 2020.
5. Wang, Z.; Groß, R.; Zhao, S. Controllability analysis and controller design for variable-pitch propeller quadcopters with one propeller failure. *Adv. Control. Appl. Eng. Ind. Syst.* **2020**, *2*, e29. [CrossRef]
6. Baldini, A.; Felicetti, R.; Freddi, A.; Longhi, S.; Monteriù, A. Actuator fault tolerant control of variable pitch quadrotor vehicles. *IFAC-PapersOnLine* **2020**, *53*, 4095–4102. [CrossRef]
7. Tuna, T.; Ovur, S.E.; Gokbel, E.; Kumbasar, T. Design and development of FOLLY: A self-foldable and self-deployable quadcopter. *Aerosp. Sci. Technol.* **2020**, *100*, 105807. [CrossRef]
8. Peraza-Hernandez, E.A.; Hartl, D.J.; Malak, R.J., Jr.; Lagoudas, D.C. Origami-inspired active structures: A synthesis and review. *Smart Mater. Struct.* **2014**, *23*, 094001. [CrossRef]
9. Mintchev, S.; Floreano, D. A pocket sized foldable quadcopter for situational awareness and reconnaissance. In Proceedings of the 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Lausanne, Switzerland, 23–27 October 2016; pp. 396–401.
10. Shu, J.; Chirarattananon, P. A quadrotor with an origami-inspired protective mechanism. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3820–3827. [CrossRef]
11. Avant, T.; Lee, U.; Katona, B.; Morgansen, K. Dynamics, hover configurations, and rotor failure restabilization of a morphing quadrotor. In Proceedings of the 2018 IEEE American Control Conference (ACC), Milwaukee, WI, USA, 27–28 June 2018; pp. 4855–4862.
12. Brischetto, S.; Ciano, A.; Ferro, C.G. A multipurpose modular drone with adjustable arms produced via the FDM additive manufacturing process. *Curved Layer. Struct.* **2016**, *3*, 202–213. [CrossRef]
13. Falanga, D.; Kleber, K.; Mintchev, S.; Floreano, D.; Scaramuzza, D. The foldable drone: A morphing quadrotor that can squeeze and fly. *IEEE Robot. Autom. Lett.* **2018**, *4*, 209–216. [CrossRef]
14. Desbiez, A.; Expert, F.; Boyron, M.; Diperi, J.; Viollet, S.; Ruffier, F. X-Morf: A crash-separable quadrotor that morfs its X-geometry in flight. In Proceedings of the 2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS), Linkoping, Sweden, 3–5 October 2017; pp. 222–227.
15. Derrouaoui, S.; Guiatni, M.; Bouzid, Y.; Dib, I.; Moudjari, N. Dynamic modeling of a transformable quadrotor. In Proceedings of the 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 1–4 September 2020; pp. 1714–1719.

16. Zhao, M.; Kawasaki, K.; Chen, X.; Noda, S.; Okada, K.; Inaba, M. Whole-body aerial manipulation by transformable multirotor with two-dimensional multilinks. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 5175–5182.
17. Pose, C.; Giribet, J. Multirotor fault tolerance based on center-of-mass shifting in case of rotor failure. In Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 15–18 June 2021; pp. 38–46.
18. Kumar, R.; Deshpande, A.M.; Wells, J.Z.; Kumar, M. Flight control of sliding arm quadcopter with dynamic structural parameters. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 1358–1363.
19. Xu, C.; Yang, Z.; Zhang, Z.; Xu, H.; Wu, J.; Zhou, D.; Liao, L.; Zhang, Q. Design and Control of a Deformable Trees-Pruning Aerial Robot. *Complexity* **2020**, *2020*, 6627339. [CrossRef]
20. Tomić, T.; Ott, C.; Haddadin, S. External wrench estimation, collision detection, and reflex reaction for flying robots. *IEEE Trans. Robot.* **2017**, *33*, 1467–1482. [CrossRef]
21. Li, S.; Yang, J.; Chen, W.; Chen, X. *Disturbance Observer-Based Control: Methods and Applications*; CRC Press: Boca Raton, FL, USA; Taylor & Francis Group: Abingdon, UK, 2017.
22. De Persis, C.; Isidori, A. A geometric approach to nonlinear fault detection and isolation. *IEEE Trans. Autom. Control* **2001**, *46*, 853–865. [CrossRef]
23. Isermann, R. *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*; Springer: Berlin/Heidelberg, Germany, 2005.
24. Chen, J.; Patton, R. *Robust Model-Based Fault Diagnosis for Dynamic Systems*; Springer: New York, NY, USA, 2012.
25. Baldini, A.; Felicetti, R.; Freddi, A.; Longhi, S.; Monteriù, A. Modeling and Control of a Telescopic Quadrotor Using Disturbance Observer Based Control. In Proceedings of the 2022 30th Mediterranean Conference on Control and Automation (MED), Athens, Greece, 28 June–1 July 2022; pp. 396–402. [CrossRef]
26. Fossen, T. *Guidance and Control of Ocean Vehicles*; Wiley: Hoboken, NJ, USA, 1994.
27. Michieletto, G.; Ryll, M.; Franchi, A. Fundamental actuation properties of multirotors: Force–moment decoupling and fail–safe robustness. *IEEE Trans. Robot.* **2018**, *34*, 702–715. [CrossRef]
28. TDK InvenSense. MPU-9250, Nine-Axis (Gyro + Accelerometer + Compass) MEMS MotionTracking™ Device. 2016. Available online: https://invensense.tdk.com/download-pdf/mpu-9250-datasheet/ (accessed on 24 March 2023).
29. Becker, M.; Sampaio, R.C.B.; Bouabdallah, S.; Perrot, V.d.; Siegwart, R. In-flight collision avoidance controller based only on OS4 embedded sensors. *J. Braz. Soc. Mech. Sci. Eng.* **2012**, *34*, 294–307. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.