Annual Reviews in Control 57 (2024) 100952

Contents lists available at ScienceDirect

Annual Reviews in Control

journal homepage: www.elsevier.com/locate/arcontrol

Fault tolerant control of an octorotor UAV using sliding mode for applications in challenging environments

Ahmed Khattab, Ibrahim Mizrak, Halim Alwi *

Department of Engineering, University of Exeter, UK

ARTICLE INFO

Keywords:
Fault tolerant control
Sliding mode control

ABSTRACT

This paper presents the development of fault-tolerant controller and their application for multirotor unmanned aerial vehicles – specifically an octorotor – in challenging environments e.g. nuclear power plant inspection or other dull, dirty and dangerous applications. This paper considers a combination of sliding mode control robustness properties (to deal with actuator faults) and control allocation (to automatically redistribute the control signals to healthy actuators, especially in the event of actuator failures). The resultant controller has the ability to operate in both fault-free and fault/failure conditions without reconfiguring the main baseline controller. The proposed controller also has the ability to operate for up to six rotor failures which represent an under-actuation condition i.e., a case when only two rotors are available. The under-actuation scenarios are conditions when most FTC schemes are not able to operate due to the lack of redundancy. The simulation results conducted on the nonlinear model with wind/gusts and sensor noise, show a good tracking performance under various fault-free and fault/failure scenarios (over-actuation, sufficient actuation and under-actuation conditions).

1. Introduction

In recent decades, the role and applications of unmanned aerial vehicles (UAVs — also usually known as 'drones') have become increasingly crucial in commercial and civilian sectors. UAVs are capable of undertaking tasks that are often considered dangerous, hazardous, or inaccessible for manned aircraft or workers. In the absence of an onboard pilot, the use of UAVs ensures pilots' safety and operates in hazardous and challenging environments such as nuclear power plants, chemical-polluted areas, forest fires, and active volcanoes (Garcia, 2021; Jordan et al., 2018; Mohsan, Khan, Fazal, Ullah, & Alsharif, 2022; Tabor, 2022). Moreover, UAVs offer certain advantages over larger aircraft due to their lightweight, low cost and simplicity. As a result, there has been a significant rise in the utilization of UAVs across various civil and commercial applications in recent years. The list of applications of UAVs is non-exhaustive, for example, photography, filmmaking, survey, delivery, communications, search and rescue, and infrastructure inspections (Kim, Lee, & Sohn, 2016; Lucey & Davis, 2016; Mohsan et al., 2022; Moormann, 2015).

However, the growing utilization of UAVs, particularly in civil and commercial applications, brings the need to prioritize their safe operation, especially in the absence of human pilots. This concern becomes even more crucial in the case of autonomous self-flying passenger transport UAV taxi drones, where passenger safety is of utmost importance. Despite the fact that UAVs are easily replaceable due to their low manufacturing and operating costs, the safety of individuals and, in the case of nuclear plant inspection, the protection of properties are of vital importance. Unfortunately, the rise in UAV usage in civil and commercial applications has caused an increase in reported incidents of collisions and injuries to the general public and infrastructure. As reported recently in *Annual safety review 2021* (2022), UAVs account for 26 percent of all UK aviation accident reports in 2021. While it could be argued that these incidents resulted from operator errors, they highlight a significant concern regarding UAV safety. This concern becomes even more pronounced when considering the vast number of autonomously operating without an onboard pilot, especially in densely populated areas or challenging environments such as nuclear power inspection. Consequently, there is an urgent need to address faults and failures that develop onboard during flight to ensure the safe operation of UAVs. In the case of nuclear power inspection, fault tolerant control is crucial to ensure that the UAVs can continue to operate safely, or return to the user or at least land safely when actuator faults/failures occur. Notably, most recreational and commercial UAVs are likely to lack any fault-tolerant control (FTC) capability to deal with the faults and failures that develop during flight.

Conventional feedback control strategies commonly lack the capability to handle system faults, failures, or extraordinary conditions

* Corresponding author.
E-mail address: h.alwi@exeter.ac.uk (H. Alwi).

https://doi.org/10.1016/j.arcontrol.2024.100952
Received 30 July 2023; Accepted 15 March 2024
Available online 3 April 2024

(Zhang & Jiang, 2008). Cases such as sensor, actuator, or component failures can lead to a significant degradation in system performance, potentially resulting in a catastrophic system collapse (Chen, Fekih, & Mao, 2016; Liang, Wang, Hu, & Dong, 2020; Liu, Yuan, Zhang, & Luo, 2016). Therefore, control schemes that exhibit fault tolerance without relying on complex and redundant hardware are highly advantageous.

FTC, which refers to control schemes having the ability to withstand faults, has been extensively studied in the case of manned aircraft over the past few decades (see for example various European-funded projects such as GARTEUR FM-AG16 Edwards, Lombaerts, & Smaili, 2010, ADDSAFE Goupil & Marcos, 2012, RECONFIGURE Alwi, Chen, & Edwards, 2014 and VISION Sato et al., 2018). Nevertheless, its application to commercial aircraft has been scarce and limited. Conversely, FTC for UAVs, particularly multirotor UAVs, is an emerging field. It is anticipated that FTC will be implemented in UAVs and become a fundamental feature, potentially even before its integration into manned aircraft, given the urgent necessity to ensure safety in the absence of human pilots onboard UAVs.

In the existing literature, most work on FTC for multirotor UAVs focused on the standard quadrotors, which are equipped with four vertical rotors. It is important to note that most FTC work on quadrotors primarily addresses faults rather than total actuator failures, as these systems lack redundancy. Some examples of papers from a non-exhaustive list of literature are (Chamseddine, Zhang, Rabbath, Apkarian, & Fulford, 2016; Guzmán-Rabasa et al., 2019) (gain scheduled PID and LQR), Izadi, Zhang, and Gordon (2011) (model predictive control integrated with horizon prediction and an unscented Kalman filter for parameter estimation), Ahmadi, Asadi, Merheb, Nabavi-Chashmi, and Tutsoy (2023), Gao, Liu, and Liu (2022), Li, Zhang, and Gordon (2013), Merheb, Noura, and Bateman (2015), Nguyen and Pitakwatchara (2023) (sliding mode control), Avram (2016) (nonlinear adaptive control). There have been notable works that consider octorotors and hexarotors for FTC (see for example Alwi & Edwards, 2015; Saied, Lussier, Fantoni, Shraim, & Francis, 2017, 2020; Schneider, Ducard, Rudin, & Strupler, 2012; Yu & Dong, 2019). The presence of redundant actuators in these UAV configurations offers the potential to handle total failures of specific actuators rather than just faults in the quadrotor.

Due to the availability of redundant actuators, the work on octorotors and hexarotors can handle actuator failures with the assumption that there is still enough actuator to control the UAVs. However, the studies conducted in Khattab, Alwi, and Edwards (2019b), Mueller and D. Andrea (2015), Mueller and D'Andrea (2014) on a quadrotor specifically address total failures of rotors, which cause the system to be under-actuated. In this situation, typical FTC schemes will not be able to stabilize the system. In the event of a total failure in one rotor, the controller allows the quadrotor to rotate freely around the yaw (vertical) axis, while utilizing the remaining healthy rotors to maintain control over roll and pitch angles. The control strategy employed is based on a linear quadratic regulator (LQR). The implementation results on a quadrotor demonstrate a successful safe landing in the presence of failures in one, as well as two rotors. These studies highlight the importance of developing control strategies that can effectively handle total actuator failures, especially in under-actuated systems such as quadrotors, ensuring the safety and stability of the UAV in challenging cases.

Sliding mode control (SMC) has received considerable interest in the field of FTC due to its inherent robustness against a specific type of uncertainty known as 'matched' uncertainty (Alwi, Edwards, & Tan, 2011; Edwards & Spurgeon, 1998). One of the notable advantages of SMC is its ability to directly handle actuator faults without the need for fault detection or controller reconfiguration (Alwi et al., 2011). These robustness properties make SMC an attractive approach for addressing faults in control systems. SMC is commonly categorized as a form of passive FTC due to this ability. However, it is important to note that SMC can also incorporate information provided by a fault detection and isolation (FDI) unit, which would classify it as an active FTC. By utilizing the information from the FDI unit, SMC can be implemented with less strict requirements on the control law, potentially leading to improved performance and a more aggressive level of performance compared to its passive FTC counterpart.

Actuator faults, such as 'loss of effectiveness', can be classified as matched uncertainty, and classical SMC schemes can handle them directly without requiring modifications to the control design, as long as sliding can be maintained. However, in the case of complete actuator failure (assuming redundancy is available), classical sliding mode techniques are unable to cope with failure. As a result, alternative fault-tolerant techniques need to be considered. This limitation is also seen in other conventional control schemes. However, the issue of total actuator failure can be mitigated without altering the controller's structure by combining sliding mode control with a method known as 'control allocation' (Durham, Bordignon, & Beck, 2017; Johansen & Fossen, 2013), as shown by the work by Alwi and Edwards (2008) which considers this combination (sliding mode – control allocation) for fault tolerant control.

This paper presents an FTC scheme for octorotors that are designed for operation in a challenging environment and applications such as nuclear plant inspection. Octorotors are considered over-actuated with four redundant actuators and require at least four rotors to fully control the UAV attitude. Various work on the fault-tolerant control of octorotors considers up to four failed rotors (sufficient actuation condition — see for example, Alwi & Edwards, 2015; Ijaz, Fuyang, & Hamayun, 2020; Saied, Lussier, Fantoni, Francis, & Shraim, 2015; Saied et al., 2017; Sanwale, Dahiya, Trivedi, & Kothari, 2023; Xiong, Guo, Mao, & Wang, 2023; Zeghlache, Mekki, Bouguerra, & Djerioui, 2018). However, one of the contributions of the paper is that the FTC scheme proposed in this paper has the ability to operate in both nominal and fault/failure conditions without changing the overall structure of the controller. The main contribution of the proposed controller is the ability to handle total failures of up to six motors (which constitutes under-actuation scenarios). This is different to the work in Mueller and D. Andrea (2015), Mueller and D'Andrea (2014) since these work only operates in fault/failure conditions (under-actuated case), and different to Alwi and Edwards (2015) since this controller only can handle failures up to 4 remaining rotors. Another contribution of the paper is the typical Euler angles representation used in this paper, which makes the analysis easier to understand and more natural to design. This is different compared to the work in Mueller and D. Andrea (2015), Mueller and D'Andrea (2014) and Khattab et al. (2019b), where the body acceleration unit vector representation was used as the inner loop system.

Another contribution of the paper is the utilization of an outer loop position and an inner loop attitude control which are based on sliding mode control methods. The inner loop control combines SMC with online control allocation to exploit the available actuator redundancy in the over-actuation scenario. The differential flatness property of the system (as described in Mellinger & Kumar, 2011 and Ferrin, Leishman, Beard, & McLain, 2011) is exploited in the synthesis of the outer loop position control. The simulations are conducted on the nonlinear model with wind/gusts and sensor noise, and tracking performance will be considered under various fault-free and fault/failure scenarios. The failure scenarios cover the over-actuation, sufficient actuation and under-actuation conditions.

2. Equations of motion

2.1. Generic UAVs

The rigid body equations of motion of any aircraft (fixed wing or multirotor) in body axes are given by (Beard & McLain, 2012; Nelson, 1998):

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\\ \dot{\mathcal{U}}\\ \dot{\mathcal{V}}\\ \dot{\mathcal{W}}\\ \dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix} p + qsin(\phi)tan(\theta) + rcos(\phi)tan(\theta)\\ qcos(\phi) - rsin(\phi)\\ qsin(\phi)cos^{-1}(\theta) + rcos(\phi)cos^{-1}(\theta)\\ \mathcal{V}r - q\mathcal{W} - gsin(\theta)\\ p\mathcal{W} - \mathcal{U}r + gcos(\theta)sin(\phi)\\ \mathcal{U}q - p\mathcal{V} + gcos(\theta)cos(\phi)\\ qr(I_{yy} - I_{zz})/I_{xx}\\ pr(I_{zz} - I_{xx})/I_{yy}\\ qr(I_{xx} - I_{yy})/I_{zz}\end{bmatrix}$$
$$+ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0\\ \frac{1}{m_{kg}} & 0 & 0 & 0 & 0 & 0\\ 0 & \frac{1}{m_{kg}} & 0 & 0 & 0 & 0\\ 0 & 0 & \frac{1}{m_{kg}} & 0 & 0 & 0\\ 0 & 0 & 0 & \frac{1}{I_{xx}} & 0 & 0\\ 0 & 0 & 0 & 0 & \frac{1}{I_{yy}} & 0\\ 0 & 0 & 0 & 0 & 0 & \frac{1}{I_{zz}}\end{bmatrix} \begin{bmatrix}F_x\\ F_y\\ F_z\\ \mathcal{L}\\ \mathcal{M}\\ \mathcal{N}\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 0\\ 0\\ 0\\ 0\\ J_r q\Omega_r/I_{xx}\\ -J_r p\Omega_r/I_{yy}\\ 0\end{bmatrix} \tag{1}$$

where g and m_{kg} are gravity and mass of aircraft. The states ϕ, θ, ψ are the Euler angles (roll, pitch and yaw), p, q, r are the angular velocities in the body axes (roll, pitch and yaw rates) and $\mathcal{U}, \mathcal{V}, \mathcal{W}$ are the linear velocities in body axes respectively. The parameters I_{xx}, I_{yy}, I_{zz} are the components of the moment of inertia about the x, y, z body axes. Note that in (1), it is assumed that the UAV is symmetric and therefore no off-diagonal terms of the moment of inertia (i.e. $I_{xy} = I_{yz} = 0$ and I_{xz} is small compared to I_{xx}, I_{yy}, I_{zz} and therefore, neglected — which is common in the literature) (Beard & McLain, 2012).

Note that the last term in (1) is typically specific for multirotor UAVs, and does not appear in fixed-wing UAVs. For a multirotor UAV, J_r is the vertical rotor inertia and Ω_r is the overall residual propeller speed from unbalanced vertical rotor rotation (and typically small and usually neglected in nominal situations). This will be described in the next section and depends on the number of vertical rotors.

In (1), the variables F_x, F_y, F_z are the total force in the x, y, z body axes, and $\mathcal{L}, \mathcal{M}, \mathcal{N}$ are the moments and torques around the x, y, z body axes respectively. For generic UAVs (including fixed-wing UAVs), the forces and moment equations are defined as

$$\begin{bmatrix}F_x & F_y & F_z\end{bmatrix}^T = F_{aero} + F_g + F_t \tag{2}$$

$$\begin{bmatrix}\mathcal{L} & \mathcal{M} & \mathcal{N}\end{bmatrix}^T = M_{aero} + M_t \tag{3}$$

where F_g is the gravitational forces, F_t and M_t are forces and moments generated by thrusts (e.g. engines/motors) and F_{aero}, M_{aero} are the aerodynamic forces and moments. The exact equations for the variables $F_x, F_y, F_z, \mathcal{L}, \mathcal{M}$ and $\mathcal{N}$ will depend on the UAV. For example, for multirotor UAVs, there are no aerodynamic components (e.g. fixed-wing UAVs) and therefore, $F_{aero} = M_{aero} = 0$ and the forces and moment equations are only dependent on the vertical rotors (see for example Khattab, Alwi, & Edwards, 2018). For fixed-wing UAVs, F_{aero}, M_{aero} are mainly produced by the wings (and therefore dependent on the states of the UAVs) and the control surfaces and therefore nonzero (see for example Vile, Alwi, Edwards, & Yates, 2021). A special case is the spherical UAV (e.g. Khattab, Alwi, & Edwards, 2019a), which is a type of multirotor UAV, but also contains control surfaces to provide inputs to the system (and therefore $F_{aero} \neq 0$ and $M_{aero} \neq 0$).

The velocity of the octorotor in the earth axes is given by

$$\begin{bmatrix}\dot{x}_e\\ \dot{y}_e\\ \dot{z}_e\end{bmatrix} = (R_b^i(\phi, \theta, \psi))^T \begin{bmatrix}\mathcal{U}\\ \mathcal{V}\\ \mathcal{W}\end{bmatrix} \tag{4}$$

where $R_b^i(\phi, \theta, \psi)$ is the Direction Cosine Matrix (DCM) and is given by:

$$R_b^i(\phi, \theta, \psi) = \begin{bmatrix} cos(\theta)cos(\psi) & cos(\theta)cos(\psi) & -sin(\theta)\\ sin(\phi)sin(\theta)cos(\psi) - cos(\phi)sin(\psi) & sin(\phi)sin(\theta)sin(\psi) + cos(\phi)cos(\psi) & sin(\theta)cos(\theta)\\ cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi) & cos(\phi)sin(\theta)sin(\psi) - sin(\phi)cos(\psi) & cos(\phi)cos(\theta)\end{bmatrix} \tag{5}$$

Note that the DCM matrix in (5) is orthogonal. The position of the octorotor can be obtained by integrating the velocity in Eq. (4).

2.2. Equations of motion for an octorotor

The octorotor with configurations shown in Fig. 1 will be considered in this paper. Here, it is assumed that all the rotors are vertical (not tilted) and therefore, $F_x = F_y = 0$ (since the rotors do not produce any forces in the x and y axes). Therefore Eq. (1) becomes

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\\ \dot{\mathcal{U}}\\ \dot{\mathcal{V}}\\ \dot{\mathcal{W}}\\ \dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix} p + qsin(\phi)tan(\theta) + rcos(\phi)tan(\theta)\\ qcos(\phi) - rsin(\phi)\\ qsin(\phi)cos^{-1}(\theta) + rcos(\phi)cos^{-1}(\theta)\\ \mathcal{V}r - q\mathcal{W} - gsin(\theta)\\ \mathcal{W}p - \mathcal{U}r + gcos(\theta)sin(\phi)\\ \mathcal{U}q - p\mathcal{V} + gcos(\theta)cos(\phi)\\ qr(I_{yy} - I_{zz})/I_{xx}\\ pr(I_{zz} - I_{xx})/I_{yy}\\ qr(I_{xx} - I_{yy})/I_{zz}\end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ \frac{1}{m_{kg}} & 0 & 0 & 0\\ 0 & \frac{1}{I_{xx}} & 0 & 0\\ 0 & 0 & \frac{1}{I_{yy}} & 0\\ 0 & 0 & 0 & \frac{1}{I_{zz}}\end{bmatrix}}_{B_\tau}$$
$$\times \underbrace{\begin{bmatrix}F_z\\ \mathcal{L}\\ \mathcal{M}\\ \mathcal{N}\end{bmatrix}}_{\tau(t)} + \begin{bmatrix}0\\ 0\\ 0\\ 0\\ 0\\ 0\\ J_r q\Omega_r/I_{xx}\\ -J_r p\Omega_r/I_{yy}\\ 0\end{bmatrix} \tag{6}$$

For an octorotor in Fig. 1, the inputs of the system (6) are the total force in the z body axis (F_z), and the moments around the x, y, z body axes - ($\mathcal{L}, \mathcal{M}, \mathcal{N}$) respectively. These total forces and moments are mapped from all eight individual rotors given by

$$\underbrace{\left[\begin{array}{c}F_z\\ \hline \mathcal{L}\\ \mathcal{M}\\ \hline \mathcal{N}\end{array}\right]}_{\tau(t)} = \underbrace{\left[\begin{array}{cccccccc} -b & -b & -b & -b & -b & -b & -b & -b\\ \hline -b\ell_1 & b\ell_1 & b\ell_2 & b\ell_2 & b\ell_1 & -b\ell_1 & -b\ell_2 & -b\ell_2\\ b\ell_2 & b\ell_2 & b\ell_1 & -b\ell_1 & -b\ell_2 & -b\ell_2 & -b\ell_1 & b\ell_1\\ \hline d & -d & d & -d & d & -d & d & -d\end{array}\right]}_{B_\Omega} \underbrace{\begin{bmatrix}\Omega_1^2(t)\\ \cdot\\ \cdot\\ \cdot\\ \Omega_8^2(t)\end{bmatrix}}_{u_\Omega(t)} \tag{7}$$

where b, d are the motor's constant thrust and torque coefficients and $\Omega_1, \ldots, \Omega_8$ are the individual propeller angular speeds. From Fig. 1, L is the motor arm of the octorotor, hence $\ell_1 = Lcos(\gamma)$ and $\ell_2 = Lsin(\gamma)$ where $\gamma = 22.5^o$. As described in the last section, the last term in (1) and (6) is nonzero and Ω_r is defined as

$$\Omega_r = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 + \Omega_5 - \Omega_6 + \Omega_7 - \Omega_8 \tag{8}$$

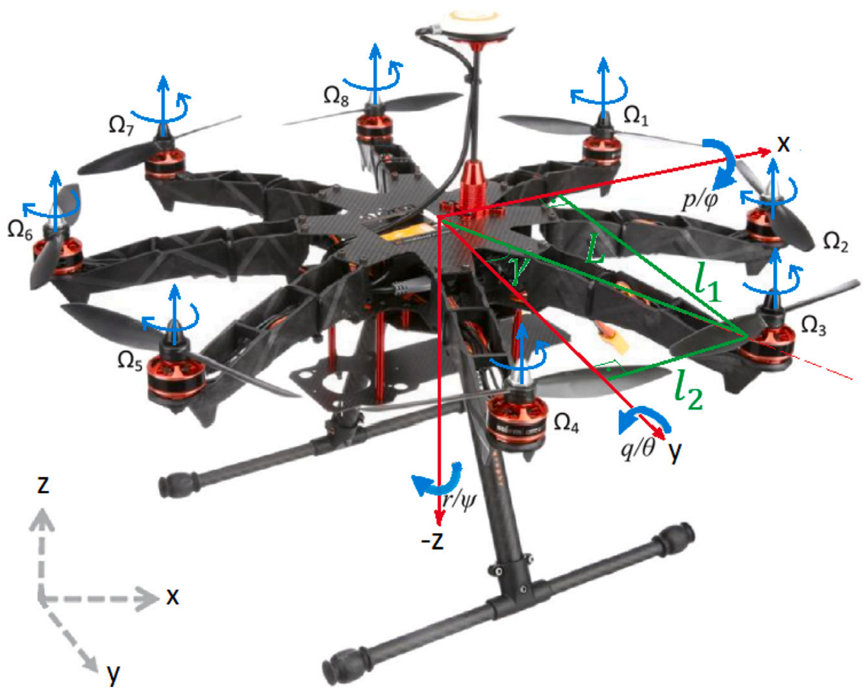


Fig. 1. Octorotor Motor Configuration.

3. Overall control strategy - dealing with up to six rotor failures

Octorotors are considered over-actuated and potentially can handle up to four actuator failures using typical FTC (see for example, Alwi & Edwards, 2015 and Saied et al., 2015). However, this paper will consider failures of up to six rotors which constitutes under-actuated cases. Specifically, two cases will be considered for the under-actuated scenario; the first scenario involves five rotor failures (three remaining rotors – where the yaw angle control will be sacrificed) and the second scenario involves six rotor failures (two remaining rotors – which will sacrifice both yaw and pitch angle controls in order to maintain control of the translational position of the UAV).

The overall control scheme presented in this paper is depicted in Fig. 2. It consists of two cascaded control loops; the first one is the inner loop utilizing the SMC attitude control to generate the "virtual" control laws responsible for determining the desired roll moment $\mathcal{L}$ and desired pitch moment $\mathcal{M}$. Moreover, a separate yaw control mechanism, employing a proportional–integral–derivative (PID) controller, is implemented to generate the desired yaw moment $\mathcal{N}$ under normal operating conditions, assuming there are no faults and adequate control capabilities. The differential flatness properties, as described in Mellinger and Kumar (2011) and Ferrin et al. (2011), are directly utilized to convert the outer-loop altitude position control into the desired vertical thrust F_z. To convert the virtual control laws $(F_z, \mathcal{L}, \mathcal{M}, \mathcal{N})$ into individual rotor speeds, a CA unit is employed. Additionally, an outer-loop position control, based on SMC, utilizes the differential flatness properties to generate the desired roll and pitch angles for the inner-loop control, as well as determine the total thrust F_z. Fig. 2 includes two switches, which are activated by an FDI unit which is assumed to be available. The first switch is used to disable the yaw moment $(\mathcal{N})$ control when only three rotors remain operational, indicating an underactuated scenario. In the case of two remaining operational rotors (which is the worst underactuated case considered in this paper), the FDI unit activates the second switch to disable the pitch moment $(\mathcal{M})$ control. In this scenario, only roll moment $(\mathcal{L})$ and altitude control (though vertical thrust F_z) remain functional, but they are sufficient to provide a certain level of control for the octocopter. Detailed definitions of each block shown in Fig. 2 will be given in the following sections of the paper.

3.1. Inner loop control

3.1.1. Roll and pitch moment $(\mathcal{L}, \mathcal{M})$ control

This section presents the control for the inner loop roll and pitch moments $(\mathcal{L}, \mathcal{M})$. By exploiting the decoupling property of the input force and moments in Eq. (6), the reduced-order dynamics can be written as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{p} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} p + qsin(\phi)tan(\theta) + rcos(\phi)tan(\theta) \\ qcos(\phi) - rsin(\phi) \\ c_1 qr + c_3 q\Omega_r \\ c_2 pr - c_4 p\Omega_r \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{\mathcal{L}}{I_{xx}} \\ \frac{\mathcal{M}}{I_{yy}} \end{bmatrix} \tag{9}$$

Here, the constant values of c_i given by $c_1 = \frac{I_{yy}-I_{zz}}{I_{xx}}$, $c_2 = \frac{I_{zz}-I_{xx}}{I_{yy}}$, $c_3 = \frac{J_r}{I_{xx}}$ and $c_4 = \frac{J_r}{I_{yy}}$. The reduced order dynamics described in (9) are nonlinear and part of the overall nonlinear equations of motion in (6). For the design objectives, the system in (9) can be written in a matrix form as

$$\underbrace{\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{p} \\ \dot{q} \end{bmatrix}}_{\dot{x}_p(t)} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_p} \underbrace{\begin{bmatrix} \phi \\ \theta \\ p \\ q \end{bmatrix}}_{x_p(t)} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{Ixx} & 0 \\ 0 & \frac{1}{Iyy} \end{bmatrix}}_{B_{p\tau}} \underbrace{\begin{bmatrix} \mathcal{L} \\ \mathcal{M} \end{bmatrix}}_{\tau_p(t)}$$

$$+ \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ c_3 & 0 \\ 0 & c_4 \end{bmatrix}}_{D_p} \underbrace{\begin{bmatrix} q\Omega_r(t) \\ p\Omega_r(t) \end{bmatrix}}_{\varsigma(t)} + \underbrace{\begin{bmatrix} sin(\phi)tan(\theta)q + cos(\phi)tan(\theta)r \\ (cos(\phi) - 1)q - sin(\phi)r \\ c_1 qr \\ c_2 pr \end{bmatrix}}_{\zeta(t)} \tag{10}$$

The term Ω_r in (10) is a 'matched' uncertainty (Alwi et al., 2011; Edwards & Spurgeon, 1998) due to the structure of D_p. The nonlinear

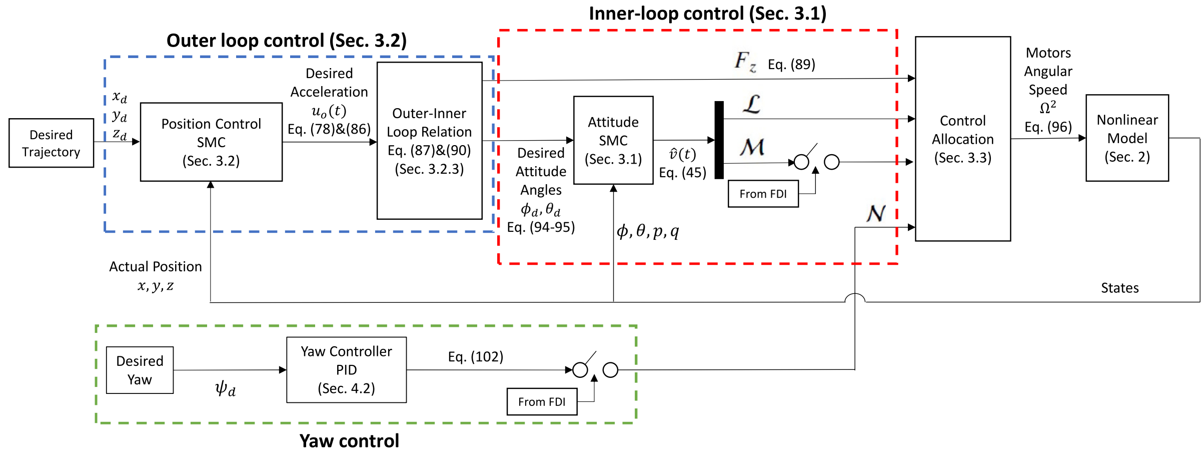

Inner-loop control (Sec. 3.1)
Outer loop control (Sec. 3.2)
Desired Trajectory
x_d
y_d
z_d
Position Control SMC (Sec. 3.2)
Desired Acceleration $u_o(t)$ Eq. (78)&(86)
Outer-Inner Loop Relation Eq. (87)&(90) (Sec. 3.2.3)
Desired Attitude Angles ϕ_d, θ_d Eq. (94-95)
Attitude SMC (Sec. 3.1)
$\hat{v}(t)$ Eq. (45)
F_z Eq. (89)
$\mathcal{L}$
$\mathcal{M}$
$\mathcal{N}$
From FDI
Control Allocation (Sec. 3.3)
Motors Angular Speed Ω^2 Eq. (96)
Nonlinear Model (Sec. 2)
ϕ, θ, p, q
Actual Position x, y, z
States
Desired Yaw
ψ_d
Yaw Controller PID (Sec. 4.2)
Eq. (102)
From FDI
Yaw control

Fig. 2. Storm8 octorotor Control Block Diagram.

term $\zeta(t)$ in (10) depends on known constants c_1 and c_2, as well as the states ϕ, θ, ψ, p, q, and r which are assumed to be measured. Therefore $\zeta(t)$ can be eliminated through control which will be described in the sequence.

Using (7), the roll and pitch torque vector $\tau_p(t)$ can be mapped back into the individual rotor contributions given by

$$\tau_p(t) = B_{p\Omega} u_p(t) \tag{11}$$

where $B_{p\Omega}$ is defined as:

$$B_{p\Omega} = \begin{bmatrix} -b\ell_1 & b\ell_1 & b\ell_2 & b\ell_2 & b\ell_1 & -b\ell_1 & -b\ell_2 & -b\ell_2 \\ b\ell_2 & b\ell_2 & b\ell_1 & -b\ell_1 & -b\ell_2 & -b\ell_2 & -b\ell_1 & b\ell_1 \end{bmatrix} \tag{12}$$

while

$$u_p(t) = \begin{bmatrix} \Omega_1^2 & \Omega_2^2 & \Omega_3^2 & \Omega_4^2 & \Omega_5^2 & \Omega_6^2 & \Omega_7^2 & \Omega_8^2 \end{bmatrix}^T \tag{13}$$

where $\Omega_1, \dots, \Omega_8$ are the individual rotor speeds. Note that $u_p(t)$ will contribute to the overall input of the system $u_\Omega(t)$ defined in (7). Using (11), Eq. (10) can be written as

$$\dot{x}_p = A_p x_p + \underbrace{B_{p\tau} B_{p\Omega}}_{B_p} u_p(t) + D_p \varsigma(t) + \zeta(t) \tag{14}$$

where $B_{p\tau}$ is defined in (10). To assist in the controller synthesis, the matrix B_p in (14) can be explicitly expressed as:

$$B_p = B_{p\tau} B_{p\Omega} = \begin{bmatrix} 0_{2\times 8} \\ B_2 \end{bmatrix} \tag{15}$$

where B_2 is given by

$$B_2 \\ = \begin{bmatrix} -\frac{1}{I_{xx}} b\ell_1 & \frac{1}{I_{xx}} b\ell_1 & \frac{1}{I_{xx}} b\ell_2 & \frac{1}{I_{xx}} b\ell_2 & \frac{1}{I_{xx}} b\ell_1 & -\frac{1}{I_{xx}} b\ell_1 & -\frac{1}{I_{xx}} b\ell_2 & -\frac{1}{I_{xx}} b\ell_2 \\ \frac{1}{I_{yy}} b\ell_2 & \frac{1}{I_{yy}} b\ell_2 & \frac{1}{I_{yy}} b\ell_1 & -\frac{1}{I_{yy}} b\ell_1 & -\frac{1}{I_{yy}} b\ell_2 & -\frac{1}{I_{yy}} b\ell_2 & -\frac{1}{I_{yy}} b\ell_1 & \frac{1}{I_{yy}} b\ell_1 \end{bmatrix} \tag{16}$$

3.1.2. States tracking - sliding mode integral action approach

The idea is to utilize sliding mode control for the inner-loop controller, where the states to be controlled are roll angle ϕ and pitch angle θ. Define a new state $x_r(t) \in \mathbb{R}^2$ as follows:

$$\dot{x}_r(t) = \bar{r}(t) - C x_p(t) \tag{17}$$

where $\bar{r}(t) \in \mathbb{R}^2$ is defined as the desired ϕ and θ signals and

$$C = \begin{bmatrix} I_2 & 0_{2\times 2} \end{bmatrix} \tag{18}$$

Augment the states of the system with the additional states $x_r(t)$ in (10), to yield the augmented states

$$x(t) = \begin{bmatrix} x_r(t) \\ x_p(t) \end{bmatrix} \tag{19}$$

Thus, the augmented system can be expressed as follows:

$$\dot{x}(t) = Ax(t) + Bu_p(t) + B_r \bar{r}(t) + D\varsigma(t) + E\zeta(t) \tag{20}$$

where the system matrices are

$$A = \begin{bmatrix} 0_{2\times 2} & -C \\ 0_{4\times 2} & A_p \end{bmatrix}, B = \begin{bmatrix} 0_{2\times 8} \\ B_p \end{bmatrix}, B_r = \begin{bmatrix} I_2 \\ 0_{4\times 2} \end{bmatrix}, D = \begin{bmatrix} 0_{2\times 2} \\ D_p \end{bmatrix}, E = \begin{bmatrix} 0_{2\times 4} \\ I_4 \end{bmatrix} \tag{21}$$

The synthesis of the sliding mode controller will be based on the augmented system presented in (20)–(21). The augmented system can be partitioned into the following components:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0_{4\times 8} \\ B_2 \end{bmatrix} u_p(t) + \begin{bmatrix} \bar{B}_{r1} \\ 0_{4\times 8} \end{bmatrix} \bar{r}(t) \\ + \begin{bmatrix} 0_{4\times 2} \\ D_2 \end{bmatrix} \varsigma(t) + \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} \zeta(t) \tag{22}$$

Here $x_1 \in \mathbb{R}^4$, $x_2 \in \mathbb{R}^2$, and the matrices in (22) can be further partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \left[\begin{array}{cc|c} 0 & -C_1 & -C_2 \\ 0 & A_{11} & A_{12} \\ \hline 0 & A_{21} & A_{22} \end{array}\right], \bar{B}_{r1} = \begin{bmatrix} I_2 \\ 0_{2\times 2} \end{bmatrix}, \\ D_2 = \begin{bmatrix} c_3 & 0 \\ 0 & c_4 \end{bmatrix}, E_1 = \begin{bmatrix} 0_{2\times 2} & 0_{2\times 2} \\ I_2 & 0_{2\times 2} \end{bmatrix}, E_2 = \begin{bmatrix} 0_{2\times 2} & I_2 \end{bmatrix} \tag{23}$$

3.1.3. Feedback linearization

In the case of rotor faults/failures, the augmented system (20)–(23) becomes

$$\dot{x}(t) = Ax(t) + Bu_p(t) - BKu_p(t) + B_r \bar{r}(t) + D\varsigma(t) + E\zeta(t) \tag{24}$$

where $K = diag(k_1, \dots, k_8)$ defines the fault or failure in each rotor. The scalar k_i corresponds to the reduction in the effectiveness of each rotor and satisfies the condition $0 \leq k_i \leq 1$. In the case of no faults $k_i = 0$, while in the presence of faults $0 < k_i < 1$, and if there is a complete failure of rotor $k_i = 1$. The proposed control law consists of two components:

$$u_p(t) = u(t) + \underbrace{(-B^\dagger E\zeta(t))}_{u_F(t)} \tag{25}$$

where $B^{\dagger}$ refers to the pseudo-inverse of B which is defined as:

$$B^{\dagger} = B^T(BB^T)^{-1} \tag{26}$$

In (25), $u(t)$ represents the stabilizing sliding mode controller (to deal with faults and 'matched' uncertainties which will be discussed in the following sections). The control signal $u_F(t)$ is the feedback linearization term which is used to cancel the nonlinear term $E\zeta(t)$ in (24), thus linearizing the model, which then will be used for the sliding mode design. This cancellation is possible since it is assumed that ζ defined in (10) is known. The assumption is reasonable since in aerospace systems and especially UAVs, the states ϕ, θ, p, q, r are measured, and systems constants c_1, c_2 depend on the moment of inertias I_{xx}, I_{yy}, I_{zz} which can be obtained either through experimentation (see for example Habeck & Seiler, 2016), or through computer aided drawing software (e.g. SolidWorks). By substituting (25) into (24), the following linear system can be obtained

$$\dot{x}(t) = Ax(t) + Bu(t) - BK\underbrace{(u(t) + u_F(t))}_{u_p(t)} + B_r\bar{r}(t) + D\varsigma(t) \tag{27}$$

It is important to note that the system in (27) will be utilized for the synthesis of the sliding mode control allocation controller in the sequence.

3.1.4. Control allocation

The controller synthesis, define

$$W = I - K = diag(w_1, \dots, w_8) \tag{28}$$

where W defines the effectiveness of each rotor. The scalar values w_i model the effectiveness level of each rotor and satisfy the condition $0 \leq w_i \leq 1$. In the fault-free case $w_i = 1$, while in the presence of faults $w_i < 1$, and when $w_i = 0$, it indicates a complete failure of the rotor. By applying the control law in Eq. (25) and using (28), the linearized augmented system given by Eq. (27) can be expressed as:

$$\dot{x}(t) = Ax(t) + BWu(t) + B_r\bar{r}(t) + \underbrace{\begin{bmatrix} D & -B \end{bmatrix}}_{D_a} \underbrace{\begin{bmatrix} \varsigma(t) \\ Ku_F(t) \end{bmatrix}}_{\xi(t)} \tag{29}$$

where $\xi(t)$ is regarded as an accumulated uncertainty, which is assumed to be bounded and satisfies the following condition:

$$\|\xi(t)\| \leq \beta\|\hat{v}(t)\| + \alpha(t, x) \tag{30}$$

where $\alpha(\cdot)$ and β are a known function and constant, respectively. Also note that $\xi(t)$ is 'matched' uncertainty (Alwi et al., 2011; Edwards & Spurgeon, 1998) due to the specific structure of B and D in (22). Due to the structure of the matrix B_p as shown in (15), it is possible to factorize the augmented matrix B in (21) and (29) as

$$B = \underbrace{\begin{bmatrix} 0_{4\times2} \\ I_2 \end{bmatrix}}_{B_v} B_2 \tag{31}$$

where B_2 is defined in (16). To facilitate the controller synthesis, define a virtual control input $v(t)$ as

$$v(t) = B_2u(t) \tag{32}$$

Utilizing Eq. (32), the control signal $u(t)$ can be expressed as:

$$u(t) = B_2^{\dagger}v(t) \tag{33}$$

where $B_2^{\dagger}$ is the weighted right pseudo-inverse of B_2 given by

$$B_2^{\dagger} = WB_2^T(B_2WB_2^T)^{-1} \tag{34}$$

In (34), the pseudo-inverse is weighted by the effectiveness level of each rotor through the matrix W. Employing (31), (32), and (34), the augmented system in (29) can be expressed as:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_vB_2WB_2^{\dagger}v(t) + B_r\bar{r}(t) + D_a\xi(t) \\ &= Ax(t) + B_v\hat{v}(t) + B_r\bar{r}(t) + D_a\xi(t) \end{aligned} \tag{35}$$

where $\hat{v}(t)$ represents the virtual control given by

$$\hat{v}(t) := B_2W^2B_2^T(B_2WB_2^T)^{-1}v(t) \tag{36}$$

3.1.5. Sliding mode control

In the following analysis, an SMC is designed based on the 'virtual' system presented in (35). As the reference signal $\bar{r}(t)$ in (35) is known and bounded, it does not affect the stability of the system and will be neglected in the following analysis. Moreover, the term $\xi(t)$ in (35) represents 'matched' uncertainty (Alwi et al., 2011; Edwards & Spurgeon, 1998) due to the specific structure of D and B in (22) and (31). SMC inherently exhibits robustness against such uncertainty and does not require control reconfiguration. In the event of total rotor failure, assuming there is still sufficient redundancy in the system (i.e. the condition $\det(B_2WB_2^T) \neq 0$ is satisfied), CA will be employed to redistribute the control signals among the remaining healthy rotors to achieve fault tolerance.

Neglecting the $\bar{r}(t)$ term, (35) can be expressed as

$$\underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} 0_{4\times2} \\ I_2 \end{bmatrix}}_{B_v} \hat{v}(t) + \underbrace{\begin{bmatrix} 0_{3\times10} \\ D_{a2} \end{bmatrix}}_{D_a} \xi(t) \tag{37}$$

where D_{a2} represents the bottom half of the matrix D_a. Since B_v in (37) is in the 'regular form' (Alwi et al., 2011; Edwards & Spurgeon, 1998), standard sliding mode scheme in the literature can be used to directly design the virtual controller. For the synthesis of the 'virtual' control $\hat{v}(t)$, first define

$$s(t) = Sx(t) \tag{38}$$

which represents the sliding mode switching function, and $S \in \mathbb{R}^{2\times6}$ is the sliding hyperplane matrix. For convenience in the subsequent analysis, it will be assumed that the system has been scaled such that $SB_v = I_2$. Also, define

$$\mathcal{S} = \{x \in \mathbb{R}^6 \ : \ Sx = 0\}$$

which represents the sliding hyperplane. The selection of the sliding hyperplane is the first part of any sliding mode design and defines the system's closed-loop performance. The second part is the synthesis of a control law to guarantee that the surface is reached in finite time and sliding is maintained.

Given that the system (37) is in regular form, an appropriate design for the sliding hyperplane matrix can be selected. A suitable design choice is given by

$$S = \begin{bmatrix} M & I_2 \end{bmatrix} \tag{39}$$

where $M \in \mathbb{R}^{2\times4}$ is the design freedom, and chosen in such a way that $A_{11} - A_{12}M$ is stable. Various methods can be utilized for the choice of M, which include LQR-like quadratic minimization, eigenstructure assignment (Edwards & Spurgeon, 1998), and LMI-based design (Edwards., 2004). Define a transformation matrix to achieve $(x_1, x_2) \mapsto T_sx = (x_1, s)$

$$T_s = \begin{bmatrix} I_4 & 0_{4\times2} \\ M & I_2 \end{bmatrix} \tag{40}$$

where T_s is the nonsingular transformation matrix. In the new coordinates, Eq. (37) becomes

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{s}(t) \end{bmatrix} = \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} x_1(t) \\ s(t) \end{bmatrix} + \begin{bmatrix} 0_{4\times2} \\ I_2 \end{bmatrix} \hat{v}(t) + \begin{bmatrix} 0_{4\times10} \\ D_{a2} \end{bmatrix} \xi(t) \tag{41}$$

where

$$\hat{A}_{11} := A_{11} - A_{12}M, \quad \hat{A}_{21} := M\hat{A}_{11} + A_{21} - MA_{22}, \quad \hat{A}_{22} := MA_{12} + A_{22} \tag{42}$$

If a control law can be designed to induce sliding, then during ideal sliding $\dot{s}(t) = s(t) = 0$, and so the equivalent control (Edwards & Spurgeon, 1998) necessary to maintain sliding is obtained from solving for $\hat{v}_{eq}(t)$ from the lower equations of (41) to give

$$\hat{v}_{eq}(t) = -\hat{A}_{21}\hat{x}_1(t) - D_{a2}\xi(t)(t) \tag{43}$$

As a result, from the top partition of (41), the reduced order sliding motion is given by

$$\dot{x}(t) = \hat{A}_{11}\hat{x}_1(t) \tag{44}$$

By appropriately selecting the matrix M in (39), the matrix $\hat{A}_{11}$ (defined in (42)) can be made stable.

The virtual control law is defined by

$$\hat{v}(t) = \hat{v}_l(t) + \hat{v}_n(t) \tag{45}$$

where the linear part is given by

$$\hat{v}_l(t) = -\hat{A}_{21}\hat{x}_1(t) - (\hat{A}_{22} - \Phi)s(t) \tag{46}$$

where $\Phi \in \mathbb{R}^{2\times 2}$ is a stable design matrix. The nonlinear part is given by

$$\hat{v}_n(t) = -\rho(t,x)\frac{P_2 s(t)}{\|P_2 s(t)\|} \qquad \text{if } s(t) \neq 0 \tag{47}$$

and $P_2 \in \mathbb{R}^{2\times 2}$ is a symmetric positive definite matrix that satisfies the following condition:

$$P_2\Phi + \Phi^T P_2 = -I_2 \tag{48}$$

As shown in Edwards and Spurgeon (1998), the presence of 'matched' uncertainty in (41) does not affect the reduced order sliding motion described in (44). This property is well-known in SMC (Alwi et al., 2011; Edwards & Spurgeon, 1998) and hence, the stability analysis of the closed-loop system with matched uncertainty focuses on maintaining sliding behaviour despite the presence of uncertainties or faults.

It is shown in Alwi and Edwards (2008), Alwi et al. (2011), Edwards and Spurgeon (1998) that if the matrix M has been selected such that $\hat{A}_{11}$ in (42) is stable, and the sliding mode 'modulation gain' $\rho(t,x)$ in (47) is chosen to satisfy Proposition 1 below then the controller defined in (45)–(48) guarantees the sliding motion on the hyperplane $\mathcal{S}$ even in the presence of 'matched' uncertainty. The proof follows a similar approach as presented in Alwi and Edwards (2008), Alwi et al. (2011), Edwards and Spurgeon (1998). It should be noted that the bound of the uncertainty given in (30) has been used to derive (49).

Proposition 1 (*Alwi & Edwards, 2008; Alwi et al., 2011; Edwards & Spurgeon, 1998*). *If the design matrix M has been chosen such that $\hat{A}_{11}$ in (42) is stable, then choosing*

$$\rho(t,x) \geq \frac{d_2\left(\beta\|\hat{v}_l(t)\| + \alpha(t,x)\right) + \eta}{(1-\beta d_2)} \tag{49}$$

ensures a sliding motion takes place on $\mathcal{S}$ in finite time. In (49) the scalar d_2 is defined as

$$d_2 = \|D_{a2}\| \tag{50}$$

The variable η is a positive design scalar and β is a known constant from (30).

Proof. Substituting the control law in (45)–(47) into (41) yields

$$\dot{x}_1(t) = \hat{A}_{11}x_1(t) + \hat{A}_{12}s(t) \tag{51}$$

$$\dot{s}(t) = \Phi s(t) - \rho(t,x)\frac{P_2 s}{\|P_2 s\|} + D_{a2}\xi(t) \tag{52}$$

For Eq. (52), consider a quadratic Lyapunov function (Alwi et al., 2011; Edwards & Spurgeon, 1998) given by $V(s) = s^{\mathrm{T}}P_2 s$. Using the fact that $s^{\mathrm{T}}P_2P_2s = \|P_2s\|^2$ and $\Phi^{\mathrm{T}}P_2 + P_2\Phi = -I_2$, differentiating the Lyapunov function yields:

$$\begin{aligned}\dot{V} &= s^{\mathrm{T}}\left(\Phi^{\mathrm{T}}P_2 + P_2\Phi\right)s - 2\rho(t,x)\frac{1}{\|P_2s\|}\left(s^{\mathrm{T}}P_2P_2s\right) + 2s^{\mathrm{T}}P_2D_{a2}\xi(t) \\ &= -\|s\|^2 - 2\rho(t,x)\left\|P_2s\right\| + 2s^{\mathrm{T}}P_2D_{a2}\xi(t)\end{aligned} \tag{53}$$

Furthermore, since

$$\|s^{\mathrm{T}}P_2D_{a2}\xi(t)\| \leq \|P_2s\|\, d_2\, \|\xi(t)\|$$

using the Cauchy–Schwarz inequality, the following inequality can be obtained

$$\dot{V} \leq -\|s\|^2 - 2\|P_2s\|\left(\rho(t,x) - d_2\|\xi(t)\|\right) \tag{54}$$

To assist in the following analysis, the gain ρ in (54) will be expressed in terms of the uncertainty $\xi(t)$. First rewrite Eq. (49) as

$$\rho(t,x)\left(1 - \beta d_2\right) \geq d_2\left(\beta\|\hat{v}_l(t)\| + \alpha(t,x)\right) + \eta \tag{55}$$

Rearranging this equation in terms of $\rho(t,x)$ gives the expression

$$\begin{aligned}\rho(t,x) &\geq d_2\left(\beta\|\hat{v}_l(t)\| + \alpha(t,x)\right) + \eta + \rho(t,x)\beta d_2 \\ &\geq d_2\left(\beta\|\hat{v}_l(t)\| + \rho(t,x)\beta + \alpha(t,x)\right) + \eta \\ &\geq d_2\left(\beta\|\hat{v}(t)\| + \alpha(t,x)\right) + \eta \\ &\geq d_2\|\xi(t)\| + \eta\end{aligned} \tag{56}$$

Substituting Eq. (56) in (54) yields

$$\dot{V} \leq -\|s\|^2 - 2\eta\|P_2s\| \tag{57}$$

Eq. (57) shows that despite the presence of matched uncertainty, the control law (45)–(47) still induces sliding on the hyperplane $\mathcal{S}$. To show that sliding will occur on the hyperplane $\mathcal{S}$ in finite time, using the Rayleigh principle

$$\begin{aligned}\|P_2s\|^2 &= (P_2^{1/2}s)^{\mathrm{T}}P_2(P_2^{1/2}s) \geq \lambda_{min}(P_2)\|P_2^{1/2}s\|^2 \\ &= \lambda_{min}(P_2)V(s)\end{aligned} \tag{58}$$

Substituting (58) into (57) yields

$$\dot{V} \leq -2\eta\sqrt{\lambda_{min}(P_2)}\sqrt{V} \tag{59}$$

By integrating (59), the time taken for the trajectory of the closed-loop system to reach the hyperplane $\mathcal{S}$ represented by t_s, satisfies the following inequality

$$t_s \leq \eta^{-1}\sqrt{V(s_0)/\lambda_{min}(P_2)} \tag{60}$$

where s_0 represents the initial value of switching function $s(t)$ at time $t = 0$ (Alwi et al., 2011; Edwards & Spurgeon, 1998). □

Remark. Since by assumption $\beta d_2 < 1$, the expression on the right hand side of inequality (49) is positive.

3.1.6. *Design of matrix M*

The first step in sliding mode controller design is the selection of the sliding surface matrix $\mathcal{S}$. One methodology is the quadratic cost function approach (Alwi et al., 2011; Edwards & Spurgeon, 1998). First, consider the problem of designing a sliding surface matrix $\mathcal{S}$ for the nominal linear system associated with (37) which is already in the 'regular form'. Assume there are no faults (i.e. $K(t) = 0$) and there is no reference demand ($\bar{r}(t) = 0$). Also for the purpose of design, ignore the uncertainty term. For this nominal linear system, as in Alwi et al. (2011), Edwards. (2004), Edwards and Spurgeon (1998), consider the problem of minimizing the quadratic performance index

$$J = \frac{1}{2}\int_{t_s}^{\infty} x(t)^{\mathrm{T}}Qx(t)\, dt \tag{61}$$

where $Q \in \mathbb{R}^{6\times6}$ is a symmetric positive definite matrix and t_s is the time at which the sliding motion commences. Assume that, in regular form, the matrix Q associated with Eq. (61) has a block diagonal structure so that $Q = \text{diag}(Q_1{}^{\mathrm{T}}Q_1, Q_2{}^{\mathrm{T}}Q_2)$ where $Q_2{}^{\mathrm{T}}Q_1 = 0$ and the matrix $Q_2{}^{\mathrm{T}}Q_2 \in \mathbb{R}^{2\times2}$ is nonsingular. It follows that

$$J = \frac{1}{2}\int_{t_s}^{\infty} x_1(t)^{\mathrm{T}}Q_1{}^{\mathrm{T}}Q_1x_1(t) + x_2(t)^{\mathrm{T}}Q_2{}^{\mathrm{T}}Q_2x_2(t)\, dt \tag{62}$$

Because of the assumption of regular form, under nominal fault free operation, the differential Eq. (37), whilst sliding, may be written as

$$\dot{z}_1(t) = A_{11}x_1(t) + A_{12}x_2(t) \tag{63}$$

where the states x_2 act as a 'control input' and satisfies

$$Mx_1 + x_2 = 0 \tag{64}$$

Here Eq. (64) represents the hyperplane equation $Sx = 0$ for $S = [M \quad I_2]$. Substituting for x_2 from (64) in (63) gives an autonomous reduced order sliding motion. The matrix M must be chosen to make $(A_{11} - A_{12}M)$ stable. This is always possible since (A_{11}, A_{12}) is controllable if (A, B_v) is controllable. As argued in Alwi et al. (2011), Boyd, Ghaoui, Feron, and Balakrishnan (1994) the optimal cost is given by $J = x_1(t_s)^{\mathrm{T}}P_cx_1(t_s)$ where P_c is the symmetric positive definite solution to the Riccati equation

$$P_cA_{11} + A_{11}{}^{\mathrm{T}}P_c - P_cA_{12}(Q_2{}^{\mathrm{T}}Q_2)^{-1}A_{12}{}^{\mathrm{T}}P_c + Q_1{}^{\mathrm{T}}Q_1 = 0 \tag{65}$$

where $x_1(t_s)$ is the value of the state component x_1 at the time at which sliding occurs and the optimal choice of $M = (Q_2{}^{\mathrm{T}}Q_2)^{-1}A_{12}{}^{\mathrm{T}}P_c$. This problem can be posed as an LMI optimization: Minimize trace(X^{-1}) subject to

$$\begin{bmatrix} A_{11}X + XA_{11}{}^{\mathrm{T}} - A_{12}N - N^{\mathrm{T}}A_{12}{}^{\mathrm{T}} & (Q_1X - Q_2N)^{\mathrm{T}} \\ Q_1X - Q_2N & -I \end{bmatrix} < 0 \quad , \quad X > 0 \tag{66}$$

where $N := MX$. As argued on page 114 in Boyd et al. (1994), any solution to (66) satisfies $X^{-1} \geq P_c$. Consequently $trace(X^{-1}) \geq trace(P_c)$ and hence the minimization process results in $X^{-1} = P_c$.

3.1.7. Final control law

The overall control law is determined by (25), which can be expressed as:

$$u_p(t) = \underbrace{WB_2^T(B_2W^2B_2^T)^{-1}\hat{v}(t)}_{u(t)} + \underbrace{(-B^{\dagger}E\zeta(t))}_{u_F(t)} \tag{67}$$

where $u_F(t)$ represents the feedback linearization term as defined in Section 3.1.3. The control law $u(t)$ corresponds to the SMC term obtained utilizing (33)–(34), (36) and (45)–(48). Note that the SMC law $u(t)$, sent to the actuators, depends on the effectiveness gains of each actuator w_i through the diagonal weighting matrix W. In this paper, it is assumed that this information is available through an FDI scheme (see for example Alwi & Edwards, 2008).

3.2. Outer loop control

In most sliding mode fault-tolerant control (FTC) literature, such as in Alwi et al. (2011), the primary focus is the inner loop control, while the outer loop position tracking is commonly accomplished using a simple PID control strategy. Although this approach is simple and effective, it has certain limitations, such as restricted robustness guarantees for position control. In this paper, a different approach by incorporating SMC for both the inner and outer loops is adopted. This is obtained by using the differential flatness properties of the multirotor UAV, as discussed in Ferrin et al. (2011). The fundamental idea is to employ the outer loop SMC to generate the desired/commanded reference signals for the inner loop SMC. The desired position or trajectory of the UAV, supplied by the UAV operator, serves as the reference signal for the outer loop control. To facilitate the subsequent analysis, it is assumed that the desired trajectory provided by the operator is smooth up to the fourth order.

3.2.1. Position and altitude control

In the inertial axis, the equation defining the translational motion of the octorotor is given by

$$\begin{aligned} \begin{bmatrix} \ddot{x}_e(t) \\ \ddot{y}_e(t) \\ \ddot{z}_e(t) \end{bmatrix} &= R_b^i(\phi,\theta,\psi)^T \begin{bmatrix} 0 \\ 0 \\ -T_h \end{bmatrix} \left(\frac{1}{m_{kg}}\right) + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \\ &= \begin{bmatrix} cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi) \\ cos(\phi)sin(\theta)sin(\psi) - sin(\phi)cos(\psi) \\ cos(\phi)cos(\theta) \end{bmatrix} \left(\frac{-T_h}{m_{kg}}\right) + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \end{aligned} \tag{68}$$

where T_h represents the total thrust acting on the octorotor in the upward direction, and $R_b^i(\cdot)$ denotes the direction cosine matrix defined in (5). To facilitate outer loop position control, define the external loop states as

$$x_o(t) = \begin{bmatrix} x_e(t) & y_e(t) & z_e(t) & \dot{x}_e(t) & \dot{y}_e(t) & \dot{z}_e(t) \end{bmatrix}^T \tag{69}$$

The nonlinear dynamics of the translational motion can be represented in the form of state space as

$$\dot{x}_o(t) = A_ox_o(t) + B_ou_o(t) + b_og \tag{70}$$

where g represents gravity

$$A_o = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, \quad B_o = \begin{bmatrix} 0_{3\times3} \\ I_{3\times3} \end{bmatrix}, \quad b_o = \begin{bmatrix} 0_{5\times1} \\ 1 \end{bmatrix} \tag{71}$$

Similar to the approach in Ferrin et al. (2011), it will be assumed that the inner loop controller can attain the desired input attitude, i.e., $\phi = \phi_d$, $\theta = \theta_d$, and the desired total thrust $T_h = T_d$. In (70), the input $u_o(t)$ is a nonlinear 'mapping' defined by:

$$u_o(t) = \begin{bmatrix} u_{o1}(t) \\ u_{o2}(t) \\ u_{o3}(t) \end{bmatrix} = (R_b^i(\phi_d,\theta_d,\psi))^T \begin{bmatrix} 0 \\ 0 \\ -T_h \end{bmatrix} \left(\frac{1}{m_{kg}}\right) \tag{72}$$

Therefore, it becomes possible to represent the nonlinear model described in (68) as a linear state space form, as shown in (70).

3.2.2. Outer–inner sliding mode control

The control law $u_o(t)$ in (70) will be designed using sliding mode methods. In this section, the gravitational term g will be neglected as it is known and can be cancelled using a simple feedforward term.

Similar to Section 3.1.2, integral action (Alwi et al., 2011; Edwards & Spurgeon, 1998) will be incorporated to enable tracking capability for the position variables $x_e(t)$, $y_e(t)$, and $z_e(t)$. Define the tracking state as

$$\dot{x}_c(t) = y_c(t) - C_cx(t) \tag{73}$$

where $y_c(t)$ represents a smooth and differentiable desired signal that serves as the command signal for the reference positions $x_e(t)$, $y_e(t)$, and $z_e(t)$. Also, $C_c = \begin{bmatrix} I_3 & 0_{3\times3} \end{bmatrix}$ represents the controlled output distribution matrix. To incorporate the integral action, augmenting the states in (70) with the integral action states described in (73), resulting in the following augmented state:

$$\dot{x}_a(t) = A_ax_a(t) + B_au_o(t) + B_cy_c(t) \tag{74}$$

where $x_a(t)$ represents the augmented states defined as

$$x_a(t) = \begin{bmatrix} x_c(t) \\ x_o(t) \end{bmatrix} \tag{75}$$

and the augmented system matrices are given by

$$A_a = \begin{bmatrix} 0_{3\times3} & -C_c \\ 0_{6\times3} & A_o \end{bmatrix}, \quad B_a = \begin{bmatrix} 0_{3\times3} \\ B_o \end{bmatrix}, \quad B_c = \begin{bmatrix} I_3 \\ 0_{6\times3} \end{bmatrix} \tag{76}$$

The augmented switching function is defined by

$$s_a(t) = S_a x_a(t) = \begin{bmatrix} M_a & I_3 \end{bmatrix} x_a(t) \tag{77}$$

where $M_a \in \mathbb{R}^{3\times6}$. Suppose the matrix S_a is designed so that the square matrix is nonsingular and without loss of generality, S_a can be designed so that $S_a B_a = I_3$ (Alwi et al., 2011; Edwards & Spurgeon, 1998). Also, define

$$S_o = \{x_a \in \mathbb{R}^9 \ : \ S_a x_a = 0\}$$

which represents the sliding hyperplane for the outer loop system. As described in Section 3.1.5, the control law consists of linear and nonlinear parts:

$$u_o(t) = u_{o_l}(t) + u_{o_n}(t) \tag{78}$$

The linear part includes a feed-forward reference term arising from the reference signal $y_c(t)$, defined as

$$u_{o_l}(t) = L_a x_a(t) + L_c y_c(t) \tag{79}$$

where

$$L_a = -(S_a A_a - \Phi_a S_a) \quad \text{and} \quad L_c = -S_a B_c \tag{80}$$

The matrices A_a, B_a, and S_a described in (76) and (77) are already in regular form due to the structure of B_o in (71). The matrix $\Phi_a \in \mathbb{R}^{3\times3}$ is a stable design matrix and represents the design freedom parameter (analogous to Φ in (48)), providing flexibility in the system's configuration. The nonlinear part of the control law is given by

$$u_{o_n}(t) = -\rho_o(t, x_a)\frac{s_a(t)}{\|s_a(t)\|} \quad \text{for } s_a(t) \neq 0 \tag{81}$$

where $\rho_o(t, x_a)$ represents the 'modulation gain', which is selected for the outer loop SMC to satisfy Proposition 2 below.

Proposition 2 (*Alwi et al., 2011; Edwards & Spurgeon, 1998*). *Analogous to Proposition 1, if the design matrix M_a has been chosen such that $A_{a,11} - A_{a,12}M_a$ for the system in (74) is stable, then choosing*

$$\rho_o(t, x_a) \geq \eta_o \tag{82}$$

ensures a sliding motion takes place on S_o in finite time. The variable η_o is a positive design scalar.

Proof. During ideal sliding $s_a(t) = \dot{s}_a(t) = 0$ and therefore substituting (74) and the control law (78)–(81) yield

$$\dot{s}_a(t) = \Phi_a s_a(t) - \rho_o(t, x_a)\frac{s_a(t)}{\|s_a(t)\|} \tag{83}$$

For Eq. (83), consider a quadratic Lyapunov function (Alwi et al., 2011; Edwards & Spurgeon, 1998) given by $V_o(s) = s_a^T s_a$. Similar to Proposition 1, using the fact that ${s_a}^{\mathrm{T}} s_a = \|s_a\|^2$ and ${\Phi_a}^{\mathrm{T}}\Phi_a = -I_3$, differentiating the Lyapunov function yields:

$$\begin{aligned} \dot{V}_o &= {s_a}^{\mathrm{T}}\left({\Phi_a}^{\mathrm{T}}\Phi_a\right)s_a - 2\rho_o(t, x)\frac{1}{\|s_a\|}\left({s_a}^{\mathrm{T}}s_a\right) \\ &= -\|s_a\|^2 - 2\rho_o(t, x)\,\|s_a\| \end{aligned} \tag{84}$$

Substituting (49) into (84) to yield

$$\dot{V}_o \leq -\|s_a\|^2 - 2\eta_o\|s_a\| \tag{85}$$

Eq. (85) shows that the control law (78)–(81) induces sliding on the hyperplane S_o. □

Remark. As shown above, since the system (74) contains no uncertainties, the choice of the modulation gain (82) is sufficient to ensure that sliding is attained and subsequently maintained.

3.2.3. Outer–inner loop relation

Once the design of the outer-loop SMC is complete, the idea is to utilize the control signal $u_o(t)$ from (78) to extract the desired thrust, as well as the desired roll and pitch angles. This is achieved by using the 'mapping' defined in (72). Employing (72), (70), and (68), the signal $u_o(t)$ can be viewed as the desired inertial acceleration.

$$u_o(t) = \begin{bmatrix} \ddot{x}_d & \ddot{y}_d & \ddot{z}_d \end{bmatrix}^T \tag{86}$$

Considering that the UAV is subject to gravitational acceleration $g = 9.807$ m/s^2

$$u_g(t) = u_o(t) - \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T \tag{87}$$

Using the free-body diagram analysis and evaluating the total forces acting in the vertical direction, the absolute magnitude of thrust, denoted as T_h, required from the rotors to handle the effects of gravity and weight is defined by

$$T_h = m_{kg}\left\|u_g(t)\right\| \tag{88}$$

Since the z-axis is oriented in the downward direction, the total required thrust T can be expressed as

$$F_{z_d} = -T_h \tag{89}$$

As shown in Fig. 2, the signal F_{z_d}, which is independent of the inner loop control, will be directly sent to the CA unit to determine the required speed of each rotor in order to control altitude. Also as shown in Fig. 2, the inner-loop controller's desired roll and pitch angles are provided by the outer-loop position control, exploiting the relationship between the outer and inner loops.

Following the approach in Ferrin et al. (2011), an additional transformation of the inertial axes (around the inertial axis of z) by a yaw angle ψ is considered. This transformation results in a 'control axes' system, where the demanded acceleration produced by the rotors is defined by:

$$u_C(t) = R(\psi) \times u_g(t) \tag{90}$$

where $u_g(t)$ is given in (87) and

$$R(\psi) = \begin{bmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{91}$$

During a steady-state condition, the linear acceleration affecting the body frame arises from only the thrust in the z body axis, therefore

$$u_C = \begin{bmatrix} u_{c_1} \\ u_{c_2} \\ u_{c_3} \end{bmatrix} = R^T(\theta_d)R^T(\phi_d)\begin{bmatrix} 0 \\ 0 \\ F_{z_D}/m_{kg} \end{bmatrix} \tag{92}$$

where

$$R(\phi_d) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi_d) & sin(\phi_d) \\ 0 & -sin(\phi_d) & cos(\phi_d) \end{bmatrix}, \quad R(\theta_d) = \begin{bmatrix} cos(\theta_d) & 0 & -sin(\theta_d) \\ 0 & 1 & 0 \\ sin(\theta_d) & 0 & cos(\theta_d) \end{bmatrix} \tag{93}$$

Hence, using the differential flatness property of the translational dynamics (Ferrin et al., 2011), it is possible to establish a relationship between the desired acceleration and attitude angles using (92). Consequently, once u_{c_1}, u_{c_2}, and u_{c_3} are known, the desired roll and pitch angles can be obtained using the following equations:

$$\phi_d = sin^{-1}(u_{c_2})\left(\frac{m_{kg}}{T_h}\right) \tag{94}$$

and

$$\theta_d = tan^{-1}\left(\frac{u_{c_1}}{u_{c_3}}\right) \tag{95}$$

Table 1
Chosen control input vectors and their representing matrices.

	Fault-free case and remaining rotors ≥ 4	3 rotors remaining	Two rotors remaining
$\tau(t)$	$\begin{bmatrix} F_z & \mathcal{L} & \mathcal{M} & \mathcal{N} \end{bmatrix}^T$	$\begin{bmatrix} F_z & \mathcal{L} & \mathcal{M} \end{bmatrix}^T$	$\begin{bmatrix} F_z & \mathcal{L} \end{bmatrix}^T$

As shown in Fig. 2, these signals correspond to the desired roll and pitch angles, which are utilized by the inner-loop controller (labelled as ‘Attitude SMC’).

3.3. Overall forces and moments

Since the octorotor is an over-actuated system in the fault-free case, determining the angular rotational speeds of the individual rotors requires the CA unit (see Fig. 2). In (7), once the force F_z and moments $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{N}$ are determined by the controllers, the angular velocity of each rotor Ω_i can be provided using the CA unit. Considering that different controllers produce the values of F_z, $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{N}$, the final control law, which determines the individual rotor speeds is defined as:

$$u_\Omega(t) = \begin{bmatrix} \Omega_1^2 & \dots & \Omega_8^2 \end{bmatrix}^T = B^\dagger_{\Omega_{F_z}} F_z + B^\dagger_{\Omega_N} \mathcal{N} + u_p(t) \tag{96}$$

where $u_p(t)$ represents the control law for the inner loop control as given in (67). The term F_z is obtained from (89), and $\mathcal{N}$ is the desired yaw moment derived from the yaw control (PID control), which will be further elaborated on later. In (96), the different input matrices are defined by:

$$B_{\Omega_{F_z}} = \begin{bmatrix} -b & -b & -b & -b & -b & -b & -b & -b \end{bmatrix} \tag{97}$$

$$B_{\Omega_N} = \begin{bmatrix} d & -d & d & -d & d & -d & d & -d \end{bmatrix} \tag{98}$$

In (96), the term $B^\dagger_{\Omega_i}$ represents the weighted pseudo-inverse of B_{Ω_i} given by

$$B^\dagger_{\Omega_i} = W B^T_{\Omega_i} \left(B_{\Omega_i} W B^T_{\Omega_i} \right)^{-1} \tag{99}$$

where W is a diagonal matrix whose elements vary from 0 to 1, defined by the effectiveness level of each rotor. Note that the inner loop control law $u_p(t)$ in (96) defined in (67), is responsible for generating the desired roll and pitch moments $\mathcal{L}$ and $\mathcal{M}$.

As illustrated in Fig. 2, in the case of only three rotors remaining (five rotor failures), the yaw control will be eliminated and hence the terms related to the yaw moment $\mathcal{N}$ control in (96) will be removed utilizing an FDI switch as described at the start of Section 3). The pitch control will be eliminated in the most extreme scenario, where only two adjacent rotors remain (six rotor failures), the control law $u_p(t)$ in (67) and (96) are slightly modified. In this extreme scenario, the lower row of B_2 in (16) (related to the pitch moment) is eliminated, and the CA only depends on the pseudo-inverse of the top row of B_2 defined as in (100).

$$B_{21} = \frac{b}{I_{xx}} \begin{bmatrix} -\ell_1 & \ell_1 & \ell_2 & \ell_2 & \ell_1 & -\ell_1 & -\ell_2 & -\ell_2 \end{bmatrix} \tag{100}$$

The exclusion of the pitch moment control (through the use of an FDI switch) is possible since the system defined in (37), for which the inner loop control was designed, is decoupled in the roll and pitch axes, owing to the structure of the original systems in (10). As a summary, the variation of the final control law, depending on the fault/failure cases, is given in Table 1.

3.4. Summary of overall control structure

This subsection provides a summary of the control structure for the proposed scheme. This complements Fig. 2 and the high-level description of the proposed scheme given at the start of Section 3. In the fault-free case or in the event of faults/failures where the total number of failed rotors are less than 4, the system still has a sufficient number of rotors and therefore all the ‘virtual inputs’ $F_z, \mathcal{L}, \mathcal{M}, \mathcal{N}$ will be considered. When only 3 rotors are left, the system becomes under-actuated, and the yaw control is sacrificed and removed from the overall control (as shown in the FDI-activated switch in Fig. 2). Therefore, in this scenario, only $F_z, \mathcal{L}, \mathcal{M}$ will be utilized. When only 2 opposite motors are left, pitch control $\mathcal{M}$ is also sacrificed and removed from the overall control and the ‘virtual’ inputs, leaving only F_z and $\mathcal{L}$ as the remaining control (as shown in the FDI-activated switch in Fig. 2). This will be sufficient to control the octorotor in both roll and pitch rotation directions, and will still allow the octorotor to achieve some position tracking control.

Table 2
Storm Drone 8 octorotor physical parameters (Helipal, 2023).

Parameter	Value	Unit
m_{kg}	1.214	kg
I_{xx}	0.009565	kgm^2
I_{yy}	0.013746	kgm^2
I_{zz}	0.017866	kgm^2
b	7.8×10^{-8}	N/rpm^2
d	1.8065×10^{-9}	Nm/rpm^2
ℓ	0.21	m

From Fig. 2, the aircraft’s ‘desired acceleration’ signal, described in inertial axes (see Eq. (86)) on the left side of Fig. 2 is calculated using the desired position trajectory, the measured (actual) position and the aircraft inertial velocity (via SMC). In the ‘outer–inner loop relation’ block, the ‘desired acceleration’ signal is converted to the desired acceleration in the control axes (see Eq. (90)). Using the components of this signal, the desired total thrust F_{zd} can be obtained using Eq. (89) after the gravitational force has been extracted through Eqs. (87)–(88). Also using the desired acceleration in the control axes in (86), the desired attitude roll (ϕ) and pitch (θ) angles can be calculated using (94) and (95). The desired roll and pitch moments ($\mathcal{L}$ and $\mathcal{M}$) in the ‘attitude SMC’ block are computed using (67) and dependent on the desired and measured roll and pitch angles, as well as the roll rate p, pitch rate q, roll angle ϕ and pitch angle θ. The ‘yaw controller PID’ block generates the desired yaw moment ($\mathcal{N}$) using a typical PID based design.

From Fig. 2, the ‘Control Allocation’ block transforms the ‘virtual’ control law, which includes the desired force F_z, roll, pitch, and yaw moments $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{N}$ into the individual angular speeds of the rotors. Similar to the approach in Alwi and Edwards (2008), the CA process utilizes the rotor effectiveness levels so that the control signals can be redistributed among the remaining healthy rotors. In this paper, it is assumed that FDI is available to provide information on the effectiveness level of each rotor. Additionally, the FDI information is also used to disable the yaw moment $\mathcal{N}$ and pitch moment $\mathcal{M}$ controls when only three or two motors remain (underactuated cases).

4. Design

The octorotor physical parameters are summarized in Table 2, which are based on the commercially available UAVs called Storm Drone 8 (Helipal, 2023). These parameters are obtained from the physical UAV, and the moments of inertia are obtained using a bifilar pendulum experiment (Habeck & Seiler, 2016b; Jardin & Mueller, 2009).

Table 3
Design parameters.

System	Q	Φ	ρ	δ
Outer Loop	$diag(0.0001, 0.0001, 0.0001, 1, 1, 1, 1, 1, 1)$	$-2I_3$	0.1	0.1
Inner Loop	$diag(0.6, 0.6, 40, 40, 1, 1)$	$-20I_2$	100	0.01

4.1. Sliding mode control parameters

Integral action (see Sections 3.1.2 and 3.2.2) are incorporated to provide tracking capabilities of the SMC for the inner and outer loop control. The inner and outer loop controllers are designed using an LQR-like synthesis approach for the reduced-order system described in Section 3.1.6 (see also Alwi et al., 2011; Edwards & Spurgeon, 1998). The design parameters are provided in Table 3. It is important to note that the first two elements in the design matrix Q are related to the integral action states and therefore, least weighted. Additionally, the inner-loop controller is designed to be more aggressive compared to the outer loop since the inner-loop dynamics are faster and require quicker stabilization. This is shown in Table 3, where the chosen design values for Φ and modulation gains ρ for the inner loop control have larger values to provide faster attainment of sliding. In the simulations the discontinuity in the nonlinear control terms in (47) and (81) has been smoothed by using a sigmoidal approximation $(P_2 s(t)/\|P_2 s(t)\|+\delta)$ where the smoothing scalar δ for both the inner and out loop has been chosen as 0.01 and 0.1 respectively as shown in Table 3 (see for example Section 3.7 in Edwards and Spurgeon (1998)). This removes the discontinuity and introduces a further degree of tuning, especially during actuator fault or failure conditions. For the inner loop control, the smoothing scalar δ has a much smaller value to ensure close to signum function properties (less smoothing effect of the pseudo sigmoidal expression and therefore closer to ideal sliding motion, but big enough to avoid the chattering effect).

4.2. Yaw control

As illustrated in Fig. 2, a distinct yaw control is developed using a conventional PID controller given by

$$\dot{r}_d = K_d(\psi_d - \psi) - K_d r + K_i \int (\psi_d - \psi)dt \tag{101}$$

where $\dot{r}_d$ represents the desired angular acceleration and ψ_d represents the desired yaw angle. The PID controller coefficients, which are manually tuned, are chosen as $K_p = 4$, $K_d = 4$, and $K_i = 0.1$. These gains are selected to achieve a closed-loop natural frequency of $\omega_n = 2$ and a damping ratio of $\zeta_d = 1$. Finally, the desired yaw moment N_d can be obtained from

$$\mathcal{N}_d = I_{zz}\dot{r}_d \tag{102}$$

Note that the yaw control is only active under the condition that an adequate number of rotors is available (i.e., the remaining rotors are ≥ 4). When the number of remaining rotors decreases to 3 or lower, the system becomes underactuated, and the FDI switch is employed to deactivate the yaw control. In this scenario, the octorotor is allowed to rotate freely around the yaw axis, which is similar to the approach considered in Khattab et al. (2019b), Mueller and D. Andrea (2015), Mueller and D'Andrea (2014).

5. Results and discussion

During simulation, the desired position trajectory in the inertial axis x_{o_d} is defined as:

$$x_{o_d} = \begin{bmatrix} sin\left(2\omega_d t\right) cos\left(\omega_d t\right) \\ sin\left(2\omega_d t\right) sin\left(\omega_d t\right) \\ z_{des} \end{bmatrix} \tag{103}$$

where $\omega_d = 0.1 rad/sec$ and $z_{des} = -10m$. This leads to a distinctive trajectory, as illustrated in Fig. 4a. Here, the altitude increased from 0 to 10 m at the start of the simulation followed by the change in x and y positions while maintaining altitude. Note that the desired position trajectory x_{o_d} is chosen to be smooth up to the fourth order, satisfying the requirement for differential flatness. This manoeuvre will be implemented in all tests done in this section to ensure consistency and enable direct performance comparisons across the different fault and failure cases.

In this section, a total of five cases are examined, as summarized in Table 4. The simulations start with the fault-free case to highlight the nominal performance in the absence of rotor faults/failures. Subsequently, the remaining cases are introduced in increasing order of challenge, involving faults/failures in four, five, and six rotors, as indicated in Table 4. It is important to note that the effectiveness of each rotor is assumed to be known through an FDI system similar to the one described in Alwi and Edwards (2008).

In all simulation cases, to add realism to the simulation, wind/gusts have been considered using a continuous Dryden wind turbulence model (The MathWorks Inc., 2022). Since the focus application is for indoor inspection (e.g., nuclear power plant), the wind/gusts are set as 'light', and the wind speeds and resulting angular velocities across the axis are shown in Fig. 3. Here, it can be seen that the wind/gusts affecting the octorotor, mainly the speed in x, y and z axes (with a maximum speed of approximately 0.4 m/s), as well as the roll, pitch and yaw rates (with a maximum of approximately 4 deg/sec). Additionally, to simulate the effect of interference on sensor measurement (e.g., due to electromagnetic interference), sensor noise is also included in the simulation affecting all 12 state measurements. The noise is modelled as Gaussian noise with a variance of 3×10^{-7} for all sensor measurements. This is similar to the noise profile used in Alwi and Edwards (2015), Smaili, Breeman, Lombaerts, and Joosten (2010). Both wind/gusts and sensor noise are implemented from the start of the simulation.

5.1. Case 1 - fault-free

Fig. 4 presents the fault-free case in the presence of wind/gusts and sensor noise. The desired and actual trajectories of the x, y, z positions are illustrated in Fig. 4a. The desired and measured states of the inner loop are displayed in Fig. 4b. In this case, it is evident that the tracking performance of roll, pitch, and yaw angles closely follow the desired angles (overlap between measured angles (blue) and desired angles (red)) despite the presence of wind/gusts. This figure also shows the effect of sensor noise on the state measurements.

Fig. 4c shows the switching functions for both the outer loops $(s_x(t), s_y(t), s_z(t))$ and the inner loops $(s_\phi(t), s_\theta(\text{t}))$ are close to zero (which indicates good sliding motion), despite the presence of wind/gusts and sensor noise. In this case, the inner loop switching functions are kept very small, in the order of 1×10^{-4}, while the outer loop switching function is maintained below 1×10^{-1}. These values indicate that sliding motion is achieved and maintained throughout the simulation. Note that the inner loop has a tighter performance in comparison to the outer loop as intended by the design choice described in Section 4.1 and Table 3. Finally, the eight rotor speeds for the fault-free conditions in the presence of wind/gusts and sensor noise are shown in Fig. 4d with a value of approximately 4000 rpm for each rotor.

Table 4
Test cases.

Case	Test	No. of failed rotors	W_{diag}
1	Fault-Free	0	[1 1 1 1 1 1 1 1]
2	Faults/failures (Over-actuated)	2, (and 3 faulty)	[1 0 0.7 0.5 1 0.3 0 1]
3	Four Motors (X-Configuration)	4	[1 0 1 0 0 1 0 1]
4	Three Motors (Under-actuated)	5	[0 0 0 1 0 0 1 1]
5	Two Motors (Under-actuated)	6	[1 0 0 0 1 0 0 0]

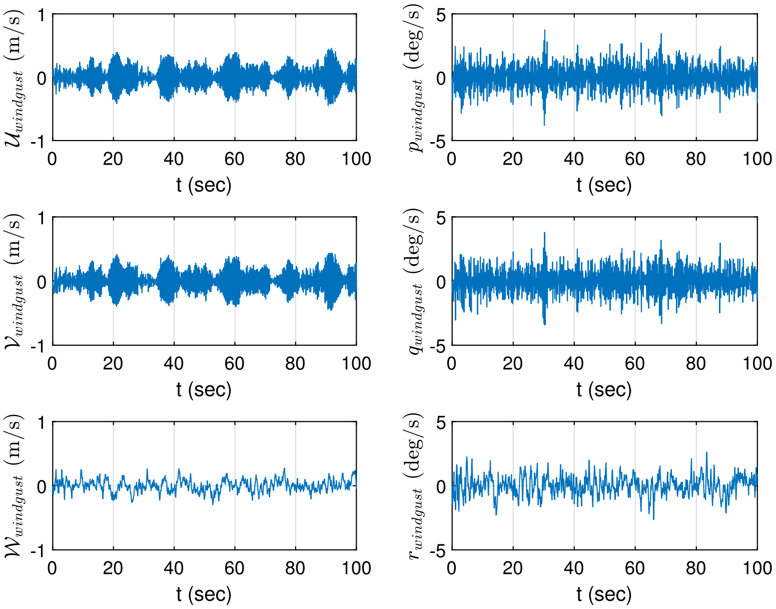


Fig. 3. Wind and gusts profile.

5.2. Case 2 - faults/failures (over-actuated)

In the second scenario, a combination of faults and failures is introduced while the system remains over-actuated, in the presence of wind/gusts and sensor noise. Specifically, two rotors (rotor 2 and 7) experience complete failure, while the remaining six rotors have different levels of effectiveness denoted by the vector

$W_{diag} = [1, 0, 0.7, 0.5, 1, 0.3, 0, 1]$.

Here only rotors 1, 5 and 8 are fault-free, while rotors 3, 4 and 6 experience some level of fault. The results, depicted in Fig. 5, demonstrate that the system maintains the same level of performance as in the fault-free case. The outer loop position tracking and the inner loop roll, pitch, and yaw angle tracking remain the same with an unnoticeable difference to the fault-free case (see Fig. 5a and 5b). Furthermore, the sliding motions are still maintained, as evidenced by the proximity of the switching functions to zero in Fig. 5c. The impact of faults and failures in the presence of wind/gusts and sensor noise on all the rotors is illustrated in Fig. 5d. In this case, rotors 2 and 7 have totally failed, while rotors 1, 5, and 8 remain fully effective, and rotor 3 operates at 70% effectiveness, rotor 4 at 50%, and rotor 6 at 30%. To compensate for the faults and failures in the other rotors, the rotor speeds for fault free rotors 1, 5, and 8 increase from their nominal values.

5.3. Case 3 - four motors (X-configuration) remaining

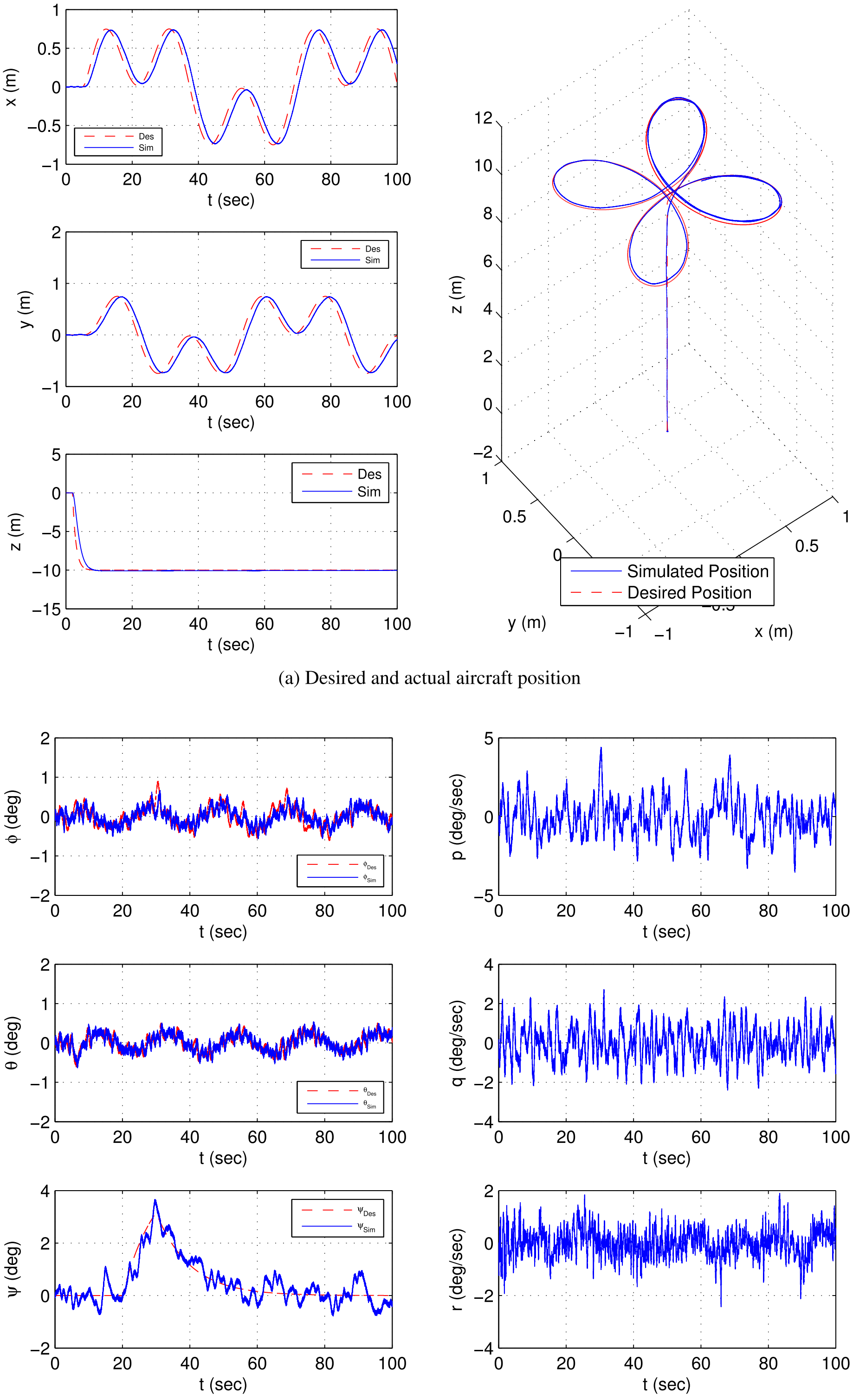
Fig. 6 presents the results when rotors 2, 4, 5, and 7 experience failure, resulting in zero rotor speeds, while the remaining rotors remain fully effective (Fig. 6). It is important to note that the hover speeds of the remaining rotors are higher than the nominal case to compensate for the failed rotors. As in the previous cases, the simulation is conducted in the presence of wind/gusts and sensor noise. In this failure case, the octorotor became an equivalent X-shaped quadrotor (although it is not a symmetric quadcopter), as shown in Fig. 6d. Similar to the previous cases, Fig. 6a and 6b demonstrate no visible degradation in the position and angle tracking performance. The switching functions also remain close to zero (see Fig. 6c), similar to the fault-free case.

5.4. Under-actuated scenarios

The last two cases, depicted in Figs. 7 and 8, are considerably more challenging compared to the previous cases. In these cases, the remaining rotors drop below the minimum required 4 failure-free rotors to maintain nominal flight, making the system underactuated. The simulations are conducted in the presence of wind/gusts and sensor noise.

5.4.1. Case 4 - three motors remaining

In Fig. 7, a case is presented where five rotors (rotors 1, 2, 3, 5, and 6) have failed (in the presence of wind/gusts and sensor noise), leaving only three functional rotors (see Fig. 7d). As a result, the system becomes underactuated, and as mentioned earlier, the control of the yaw angle (ψ) is sacrificed. This causes the octorotor to rotate in the yaw axis, leading to the fluctuation in the yaw angle between ±180 degrees (see Fig. 7b). The yaw rate (r) also reaches approximately -1100 degrees per second (equivalent to roughly -3 revolutions per second).



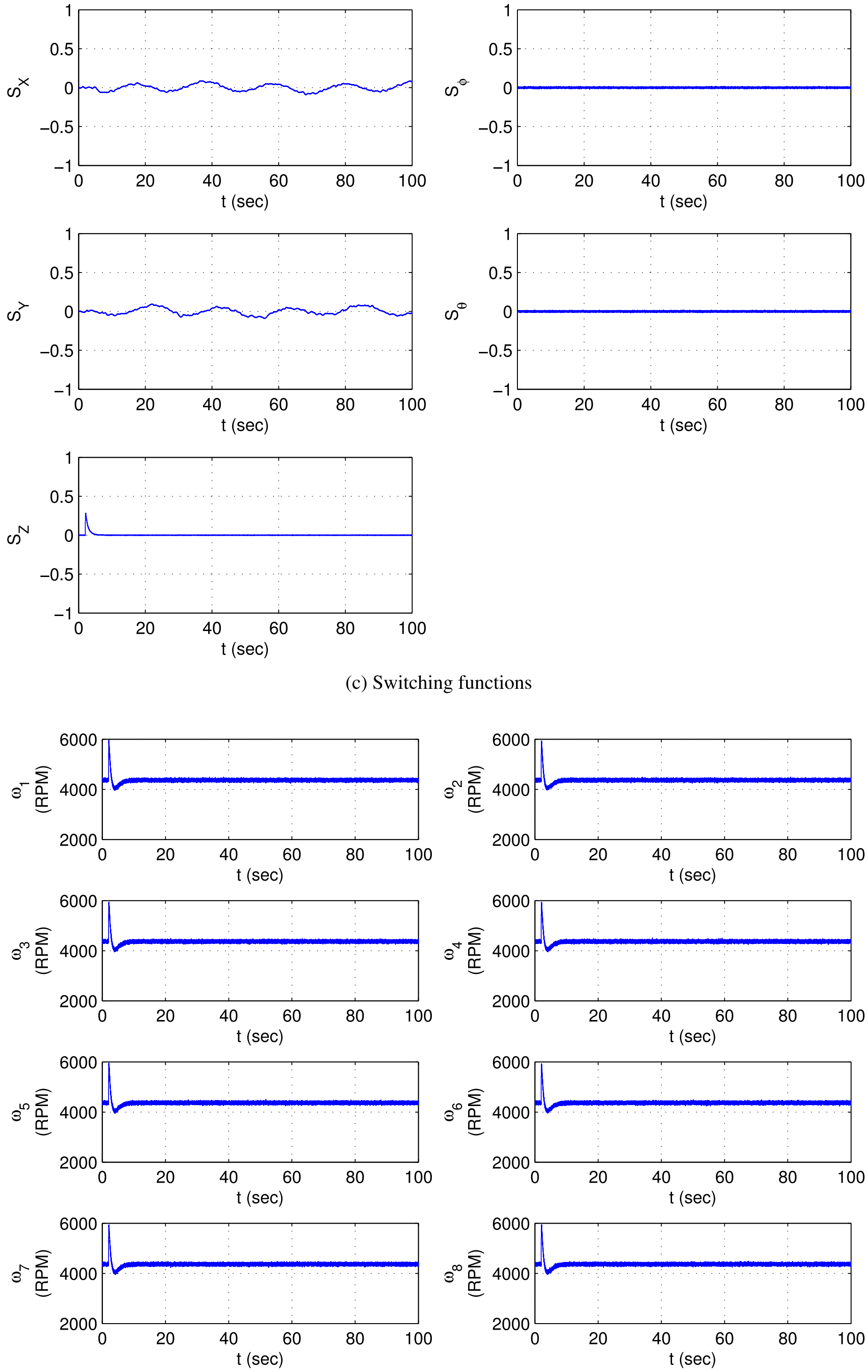
(a) Desired and actual aircraft position

(b) Desired, actual Euler angles and actual angular rates

Fig. 4. Case 1 - Fault-free.

Despite having only three remaining rotors, Fig. 7a shows no visible impact on the position tracking performance compared to the fault free case. The octorotor is still able to execute the manoeuvre successfully and has no visible difference compared to fault-free performance when maintaining altitude at 10 m during x and y position change. Fig. 7c shows the switching function remains close to zero for both inner loop and outer loop controller. There are small deviations from zero for the outer loop control switching function in comparison to previous cases, although the deviations remain small.

In the scenario where only three rotors remain, it is possible to reconfigure the controller to deactivate one of the rotors and maintain only the two opposite rotors while keeping the roll and pitch angles at zero degrees. However, this configuration would impose a higher load and rotational speed on the two remaining rotors. Under these



(c) Switching functions

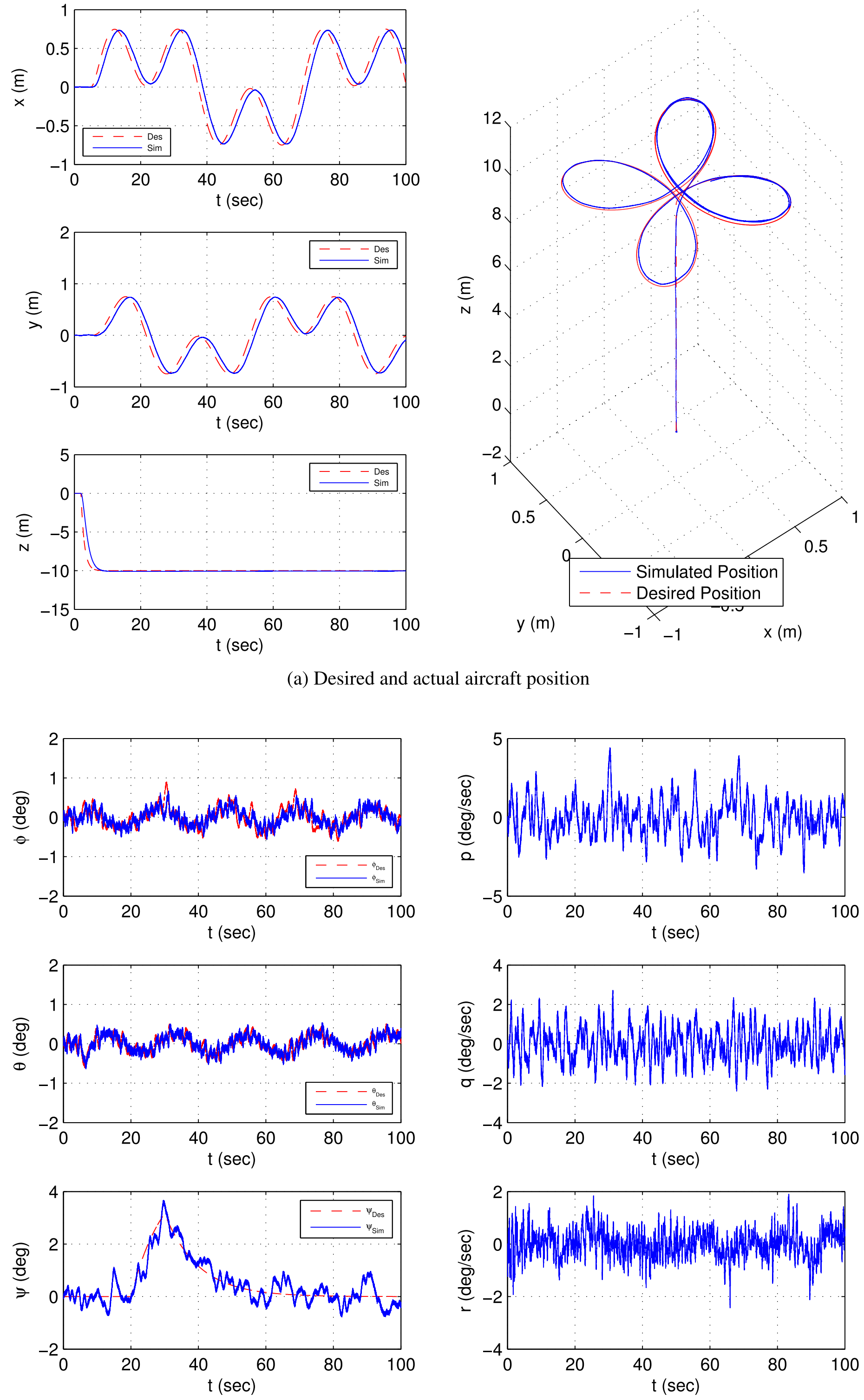
(d) Motors angular velocities

Fig. 4. (*continued*).

conditions, all three motors remain operational, although one rotor (rotor 7) operates slower than the other two rotors, as illustrated in Fig. 7d. Fig. 7b demonstrates that both the roll angle (ϕ_d) and pitch angle (θ_d) have desired values of about 1 degree to compensate for the odd number of rotors operating. These non-zero trim conditions enable rotor 7 to maintain a lower operational speed of approximately 2000 rpm, compared to rotors 4 and 8 which are close to 7000 rpm (Fig. 7d). Fig. 7b also shows a small deviation between the desired and actual pitch and roll angle, although the overall position tracking performance remains unaffected.

5.4.2. *Case 5 - two motors remaining*

The most challenging scenario is the final case, where only two of all rotors are operational, resulting from failures of the other six rotors. In this situation, the control of yaw and pitch angles (ψ and θ) are sacrificed, and only altitude and roll control are used for control.

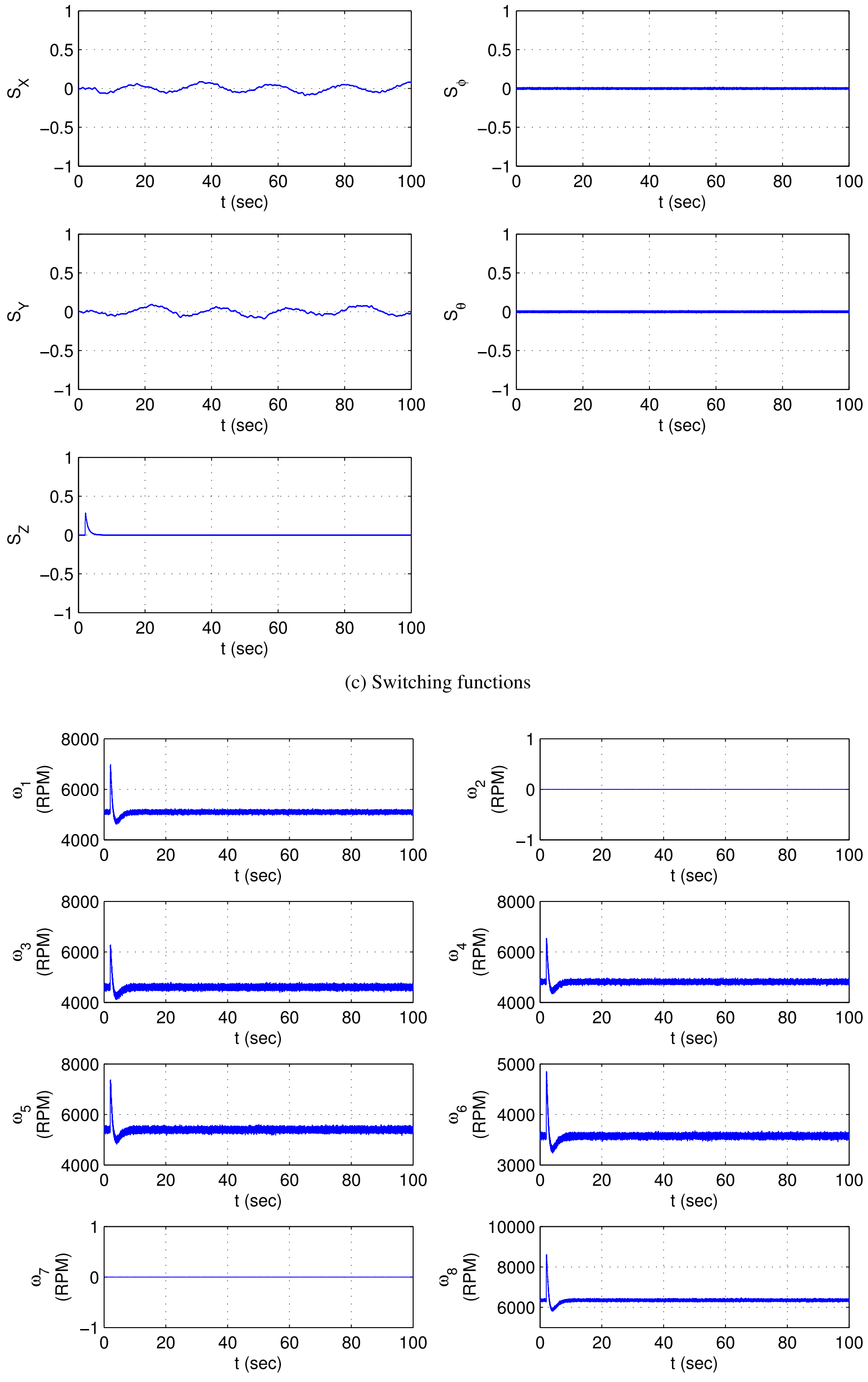


(a) Desired and actual aircraft position

(b) Desired, actual Euler angles and actual angular rates

Fig. 5. Case 2 - Faulty Scenario, $W_{diag} = [1 \quad 0 \quad 0.7 \quad 0.5 \quad 1 \quad 0.3 \quad 0 \quad 1]$.

As in the previous cases, the simulation is conducted in the presence of wind/gusts and sensor noise. Fig. 8b illustrates that the yaw and pitch movements (ψ and θ) are sacrificed, and only the roll angle (ϕ) is utilized to track the position coordinates x and y (Fig. 8a). The yaw rate r attains a maximum value of approximately $3rev/s$ (1100 deg/s). Due to the continuous rotation of the aircraft around the yaw axis, controlling the roll angle (inner loop) alone is adequate to regulate the positions x and y. In comparison to the previous cases, there is a slight degradation seen in the performance of position tracking, as depicted in Fig. 8a. Despite this minor degradation, the octorotor still managed to



(c) Switching functions

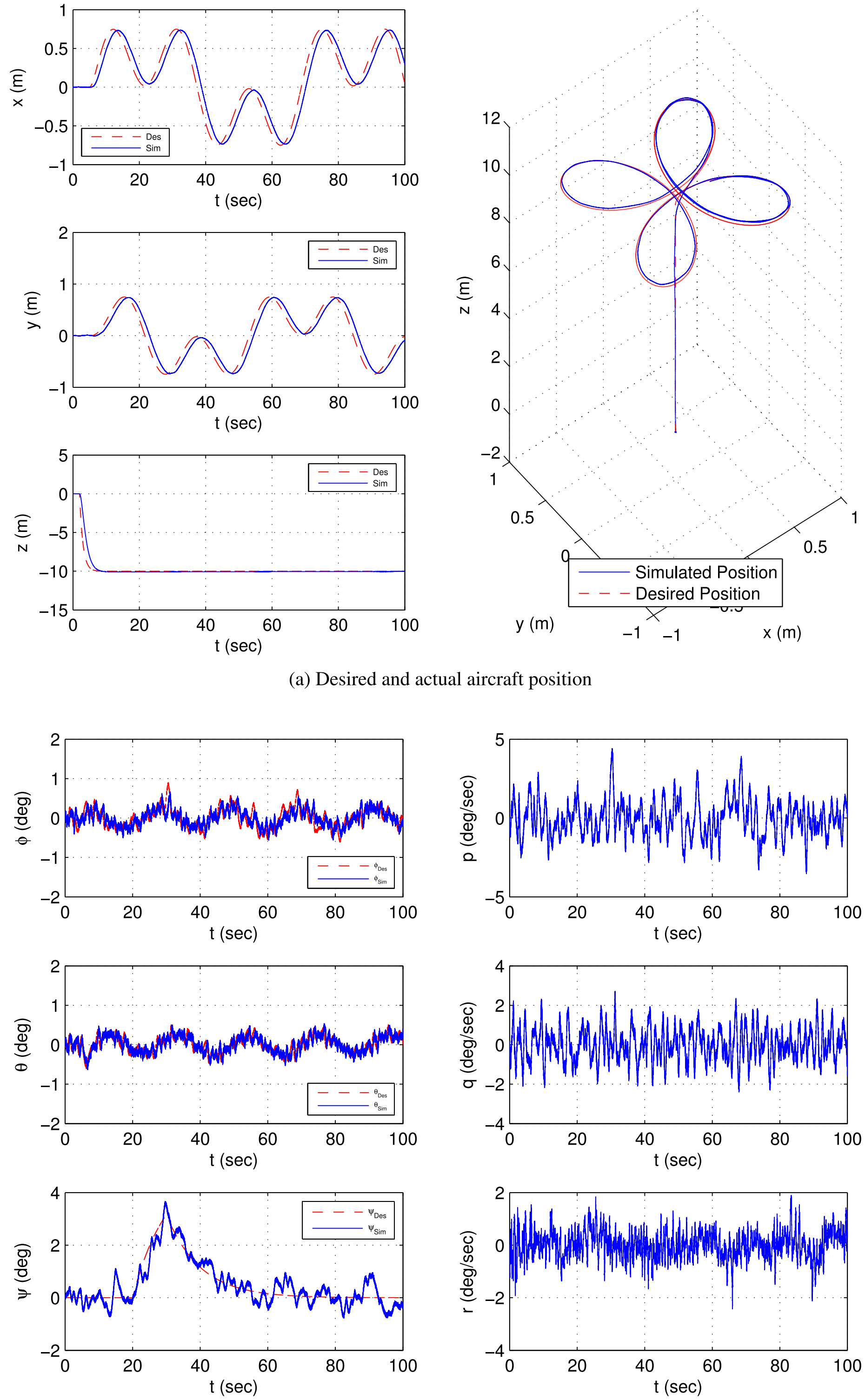
(d) Motors angular velocities

Fig. 5. (*continued*).

complete the manoeuvre. The switching functions are slightly noisier than in the previous cases (see Fig. 8c), with a maximum of ±0.2 in the X and Y axes and ±0.3 in the inner loop pitch and roll channel, which is acceptable. To compensate for the loss of the other six rotors, the remaining rotors (1 and 5) operate at a higher speed of 8000 rpm, which is significantly higher than the speed in the fault-free case (around 4000 rpm). This increased speed is necessary to generate the additional thrust required to maintain hover with only two operational rotors.

6. Conclusion

This paper has presented an FTC scheme for octorotors, offering control capabilities for fault-free, as well as over and under-actuated scenarios. Unlike existing literature on octorotors, this scheme utilizes SMC for both the outer and inner loop control. This is achieved by leveraging the differential flatness properties of the octorotor to construct appropriate state-space models and employing nonlinear feedback linearization to mitigate system nonlinearities. For inner loop

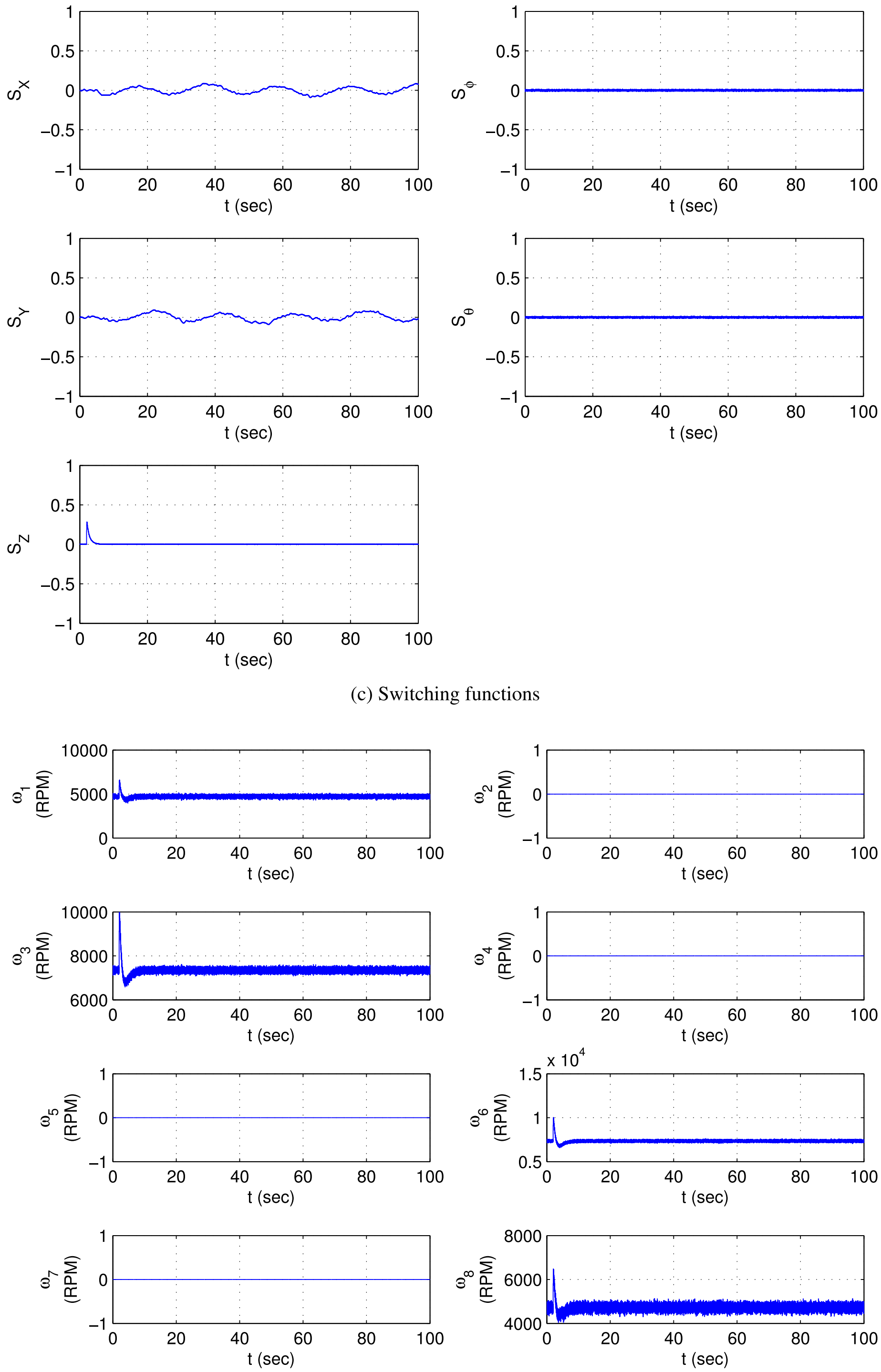


(a) Desired and actual aircraft position

(b) Desired, actual Euler angles and actual angular rates

Fig. 6. Case 3 - Four motor (X configuration scenario).

control, online control allocation is combined with SMC to dynamically redistribute control signals based on the health and effectiveness of the rotors. The paper considers five cases to evaluate the proposed scheme in the presence of wind/gusts and sensor noise. The first case represents a fault-free condition, demonstrating the controller's capabilities under nominal over-actuated conditions. Additional cases involve an over-actuated faults/failures case with two failed rotors and different effectiveness levels among the remaining six rotors. A sufficiently actuated scenario is also examined, where four rotors remain and the UAV behaves similarly to a standard quadrotor, albeit with nonsymmetrical



(c) Switching functions

(d) Motors angular velocities

Fig. 6. (*continued*).

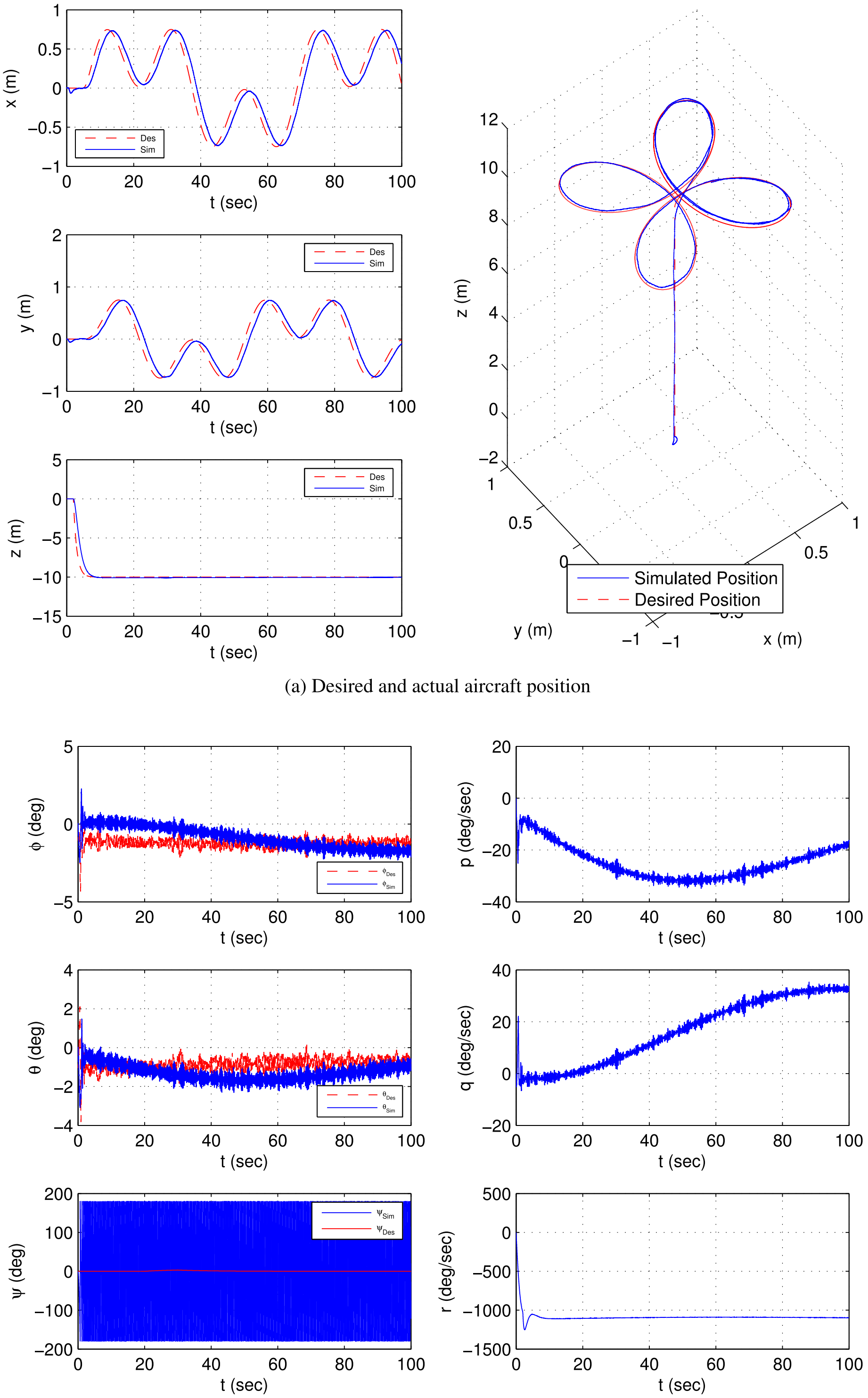
characteristics. Lastly, two extreme scenarios are investigated: one with only three remaining rotors (sacrificing yaw control to maintain position tracking) and another with only two remaining rotors (sacrificing both yaw and pitch control while preserving altitude and roll tracking). In the latter case, continuous rotation in the yaw axis is utilized to facilitate roll tracking and control over the x and y positions. Simulation results demonstrate satisfactory tracking performance across all tested scenarios, despite the presence of wind/gusts and sensor noise, affirming the effectiveness of the proposed scheme.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

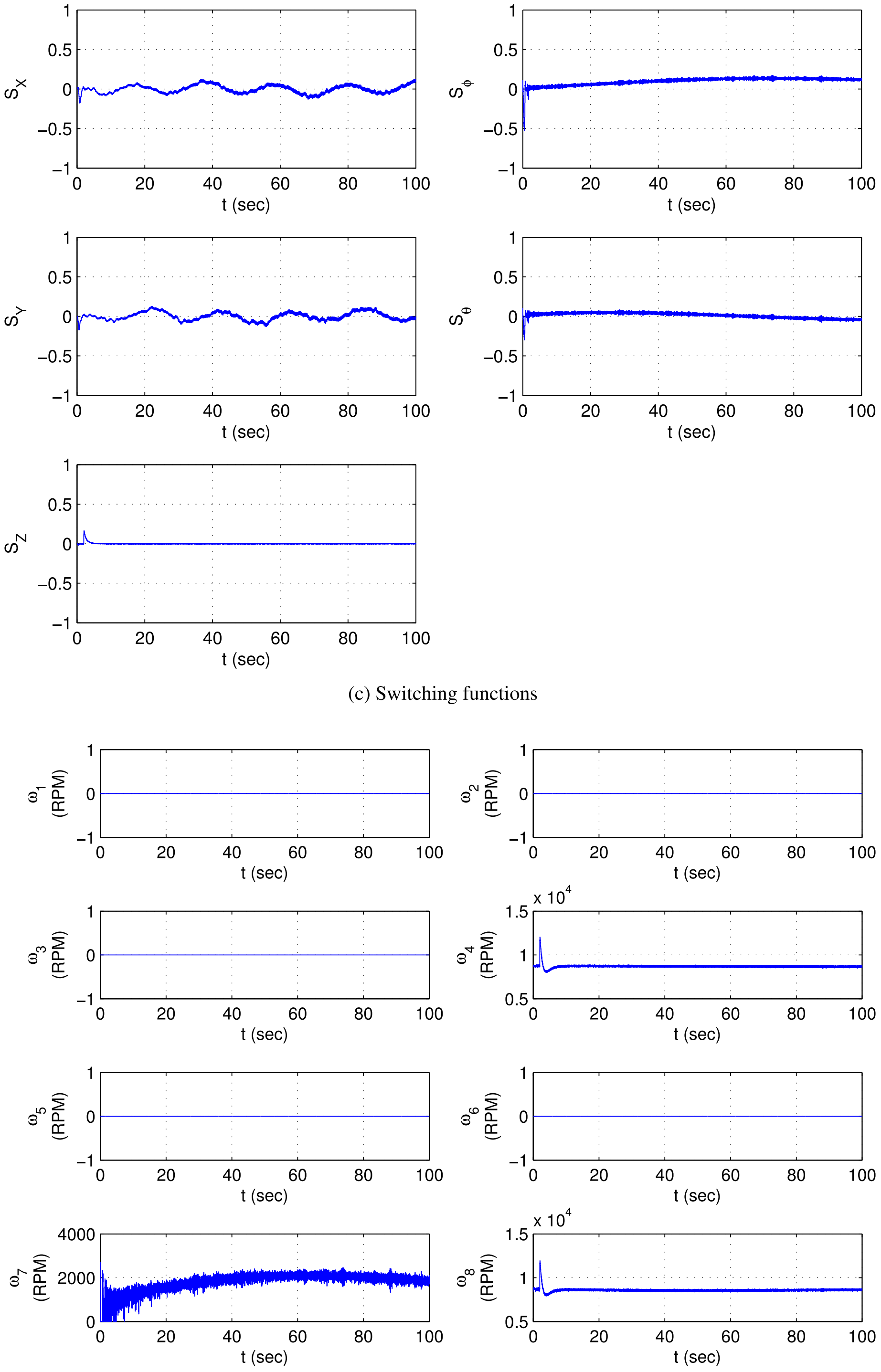
Data will be made available on request.



(a) Desired and actual aircraft position

(b) Desired, actual Euler angles and actual angular rates

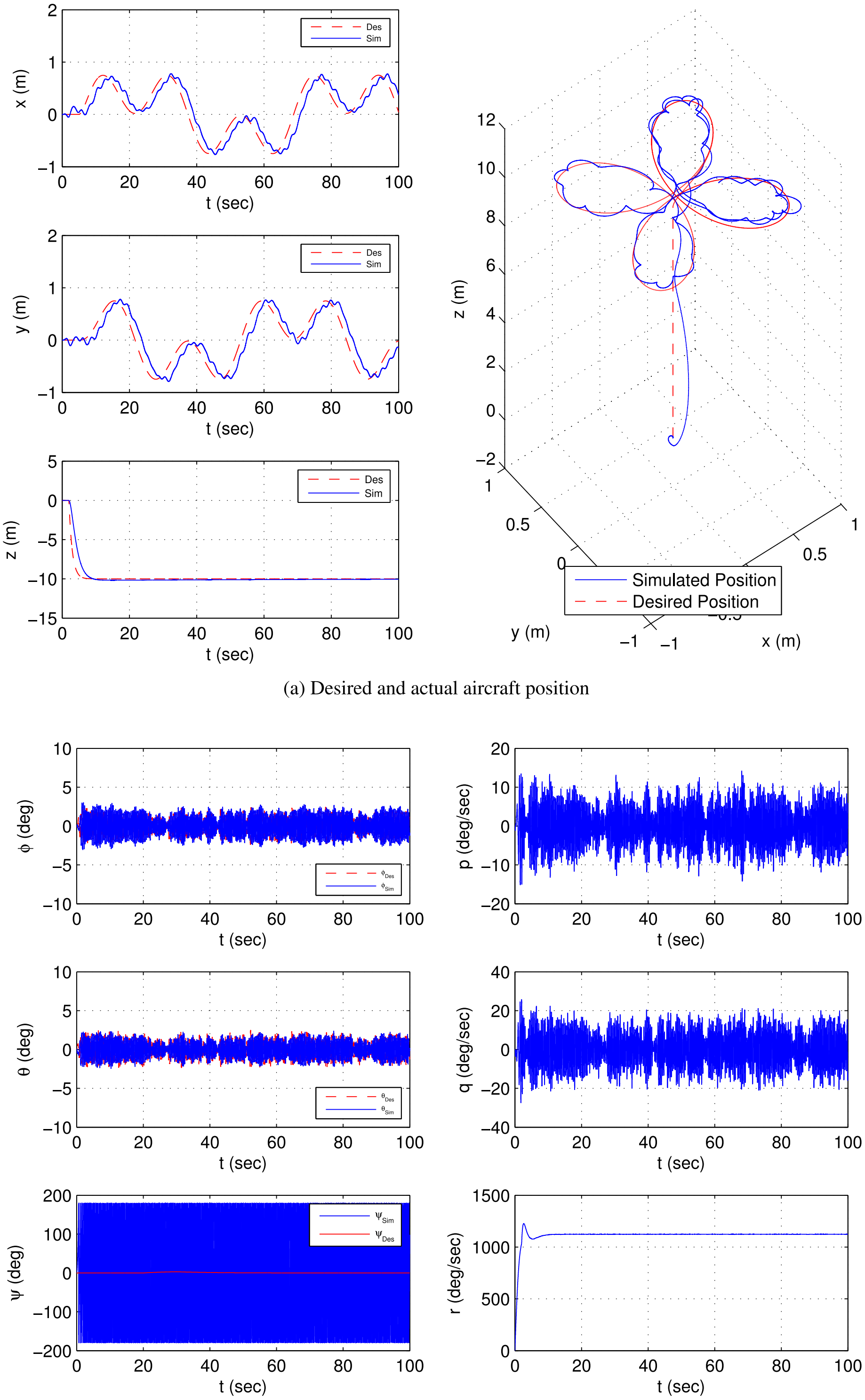
Fig. 7. Case 4 - Three motor (under-actuated) scenario.



(c) Switching functions

(d) Rotor angular velocities

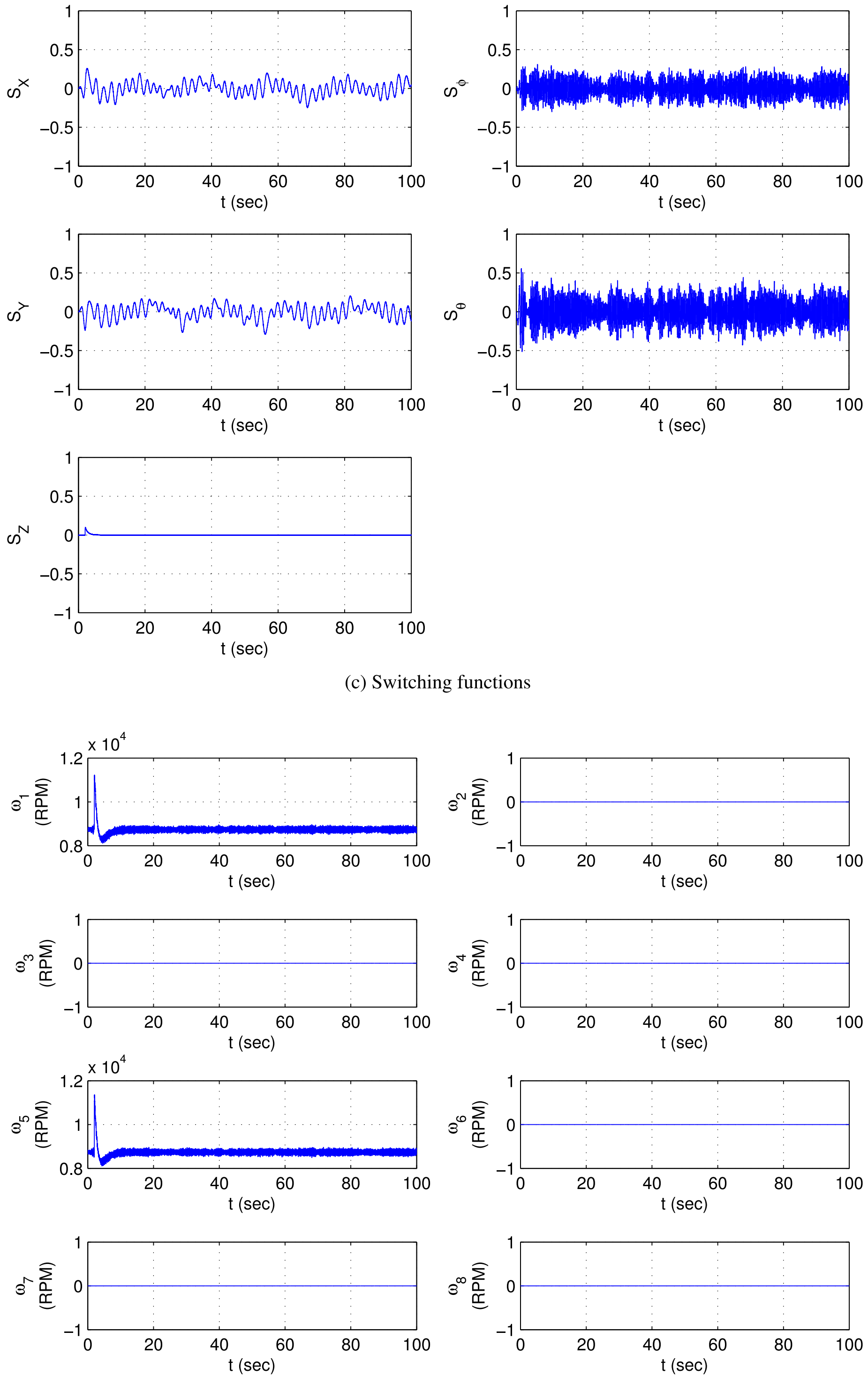
Fig. 7. (*continued*).



(a) Desired and actual aircraft position

(b) Desired, actual Euler angles and actual angular rates

Fig. 8. Case 5 - Two motors (under-actuated) scenario.



(c) Switching functions

(d) Motors angular velocities

Fig. 8. (*continued*).

Acknowledgment

The work is part of the project CRUISE: fault tolerant Control for highly Redundant multirotor Unmanned aerIal vehicle using Sliding modEs funded by EPSRC (Project Reference: EP/P015409/1).

References

Ahmadi, K., Asadi, D., Merheb, A., Nabavi-Chashmi, S.-Y., & Tutsoy, O. (2023). Active fault-tolerant control of quadrotor UAVs with nonlinear observer-based sliding mode control validated through hardware in the loop experiments. *Control Engineering Practice*, *137*, Article 105557.

Alwi, H., Chen, L., & Edwards, C. (2014). Reconstruction of simultaneous actuator and sensor faults for the reconfigure benchmark using a sliding mode observer. *IFAC Proceedings Volumes*, *47*(3), 3497–3502.

Alwi, H., & Edwards, C. (2008). Fault tolerant control using sliding modes with on-line control allocation. *Automatica*, *44*(7), 1859–1866.

Alwi, H., & Edwards, C. (2015). Sliding mode fault-tolerant control of an octorotor using linear parameter varying-based schemes. *IET Control Theory & Applications*, *9*(4), 618–636.

Alwi, H., Edwards, C., & Tan, C. (2011). *Advances in industrial control, Fault detection and fault-tolerant control using sliding modes*. Springer London.

Annual safety review 2021 (2022). *Technical report*, Air Accidents Investigation Branch.

Avram, R. C. (2016). *Fault diagnosis and fault-tolerant control of quadrotor UAVs* (Ph.D. dissertation), Wright State University.

Beard, R. W., & McLain, T. W. (2012). *Small unmanned aircraft theory and practice*. Princeton University Press.

Boyd, S., Ghaoui, L. E., Feron, E., & Balakrishnan, V. (1994). *Linear matrix inequalities in systems and control theory*. SIAM: Philadelphia.

Chamseddine, A., Zhang, Y., Rabbath, C.-A., Apkarian, J., & Fulford, C. (2016). Model reference adaptive fault tolerant control of a quadrotor UAV. In *Infotech@Aerospace 2011*.

Chen, W., Fekih, A., & Mao, Z. (2016). Fault Detection, Estimation/Reconstruction, and Fault-Tolerant Control: Theory and Practice. *Mathematical Problems in Engineering, 2016*.

Durham, W., Bordignon, K. A., & Beck, R. (2017). *Aircraft control allocation*. Chichester: John Wiley & Sons.

Edwards., C. (2004). A practical method for the design of sliding mode controllers using linear matrix inequalities. *Automatica*, *40*(10), 1761–1769.

Edwards, C., Lombaerts, T., & Smaili, H. (2010). *Fault tolerant flight control: A benchmark challenge*. Springer Berlin Heidelberg.

Edwards, C., & Spurgeon, S. (1998). *Sliding mode control: Theory and applications*. CRC Press.

Ferrin, J., Leishman, R., Beard, R., & McLain, T. (2011). Differential flatness based control of a rotorcraft for aggressive maneuvers. In *2011 IEEE/RSJ international conference on intelligent robots and systems* (pp. 2688–2693). IEEE.

Gao, B., Liu, Y.-J., & Liu, L. (2022). Adaptive neural fault-tolerant control of a quadrotor UAV via fast terminal sliding mode. *Aerospace Science and Technology*, *129*, Article 107818.

Garcia, M. R. (2021). Inspecting nuclear facilities with unmanned aerial systems. *NuclearNewswire*, *1*(1), [Online] Available: www.ans.org/news/article-3362/inspecting-nuclear-facilities-with-unmanned-aerial-systems/.

Goupil, P., & Marcos, A. (2012). Industrial benchmarking and evaluation of ADDSAFE FDD designs. *IFAC Proceedings Volumes*, *45*(20), 1131–1136.

Guzmán-Rabasa, J. A., López-Estrada, F. R., González-Contreras, B. M., Valencia-Palomo, G., Chadli, M., & Pérez-Patricio, M. (2019). Actuator fault detection and isolation on a quadrotor unmanned aerial vehicle modeled as a linear parameter-varying system. *Measurement and Control*, *52*(9–10), 1228–1239.

Habeck, J., & Seiler, P. (2016). *Moment of inertia estimation using a bifilar pendulum*: *UMTC undergraduate research presentations and papers (UROP) [1817]*, The University of Minnesota, Dept. of Aerospace Engineering and Mechanics.

Habeck, J., & Seiler, P. (2016b). *Moment of inertia estimation using a bilar pendulum*: *Technical report*, University of Minnesota.

Helipal (2023). Storm drone 8 GPS flying platform. [Online] Available: https://www.helipal.com/collections/storm-drones/products/storm-drone-8-gps-flying-platform. (Accessed 13 July 2023).

Ijaz, S., Fuyang, C., & Hamayun, M. T. (2020). Adaptive non-linear integral sliding mode fault-tolerant control allocation scheme for octorotor UAV system. *IET Control Theory & Applications*, *14*(19), 3139–3156.

Izadi, H. A., Zhang, Y., & Gordon, B. W. (2011). Fault tolerant model predictive control of quad-rotor helicopters with actuator fault estimation. *IFAC Proceedings Volumes*, *44*(1), 6343–6348.

Jardin, M. R., & Mueller, E. R. (2009). Optimized measurements of unmanned-air-vehicle mass moment of inertia with a bifilar pendulum. *Journal of Aircraft*, *46*(3), 763–775.

Johansen, T. A., & Fossen, T. I. (2013). Control allocation – A survey. *Automatica*, *49*, 1087–1103.

Jordan, S., Moore, J., Hovet, S., Box, J., Perry, J., Kirsche, K., et al. (2018). State-of-the-art technologies for UAV inspections. *IET Radar, Sonar & Navigation*, *12*(12).

Khattab, A., Alwi, H., & Edwards, C. (2018). Implementation of sliding mode fault tolerant control on the IRIS+ quadrotor. In *IEEE conference on control technology and applications* (pp. 1724–1729). IEEE.

Khattab, A., Alwi, H., & Edwards, C. (2019a). Fault tolerant control of a spherical UAV. In *4th conference on control and fault tolerant systems* (pp. 92–97). IEEE.

Khattab, A., Alwi, H., & Edwards, C. (2019b). Mitigating total rotor failure in quadrotor using LPV based sliding mode control scheme. In *4th conference on control and fault tolerant systems* (pp. 98–103). IEEE.

Kim, D. H., Lee, B. K., & Sohn, S. Y. (2016). Quantifying technology–indsustry spillover effects based on patent citation network analysis of Unmanned Aerial Vehicle (UAV). *Technological Forecasting and Social Change*, *105*, 140–157.

Li, T., Zhang, Y., & Gordon, B. W. (2013). Passive and active nonlinear fault-tolerant control of a quadrotor unmanned aerial vehicle based on the sliding mode control technique. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *227*(1), 12–23.

Liang, X., Wang, Q., Hu, C., & Dong, C. (2020). Fixed-time observer based fault tolerant attitude control for reusable launch vehicle with actuator faults. *Aerospace Science and Technology*, *107*, Article 106314.

Liu, Z., Yuan, C., Zhang, Y., & Luo, J. (2016). A learning-based fault tolerant tracking control of an unmanned quadrotor helicopter. *Journal of Intelligent and Robotic Systems*, *84*(1–4), 145–162.

Lucey, D., & Davis, B. (2016). *The first 1000 commercial UAS exemptions*: *Technical report*, AUVSI.

Mellinger, D., & Kumar, V. (2011). Minimum snap trajectory generation and control for quadrotors. In *IEEE international conference on robotics and automation* (pp. 2520–2525).

Merheb, A.-R., Noura, H., & Bateman, F. (2015). Design of passive fault–tolerant controllers of a quadrotor based on sliding mode theory. *International Journal of Applied Mathematics and Computer Science*, *25*(3), 561–576.

Mohsan, S. A. H., Khan, M. A., Fazal, N., Ullah, I., & Alsharif, M. H. (2022). Towards the unmanned aerial vehicles (UAVs): A comprehensive review. *Drones*, *6*(6).

Moormann, D. (2015). DHL parcelcopter research flight campaign 2014 for emergency delivery of medication. In *Proceedings of the ICAO RPAS symposium*.

Mueller, M. W., & D. Andrea, R. (2015). Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles. *International Journal of Robotics Research*, *35*(8), 873–889.

Mueller, M. W., & D'Andrea, R. (2014). Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers. In *Robotics and automation (ICRA), 2014 IEEE international conference on* (pp. 45–52). IEEE.

Nelson, R. C. (1998). *Flight stability and automatic control*: *Vol. 2*, WCB McGraw Hill New York.

Nguyen, N.-P., & Pitakwatchara, P. (2023). Attitude fault-tolerant control of aerial robots with sensor faults and disturbances. *Drones*, *7*(3), 156.

Saied, M., Lussier, B., Fantoni, I., Francis, C., & Shraim, H. (2015). Fault tolerant control for multiple successive failures in an octorotor: Architecture and experiments. In *IEEE international conference on intelligent robots and systems* (pp. 40–45).

Saied, M., Lussier, B., Fantoni, I., Shraim, H., & Francis, C. (2017). Fault diagnosis and fault-tolerant control of an octorotor UAV using motors speeds measurements. *IFAC-PapersOnLine*, *50*(1), 5263–5268.

Saied, M., Lussier, B., Fantoni, I., Shraim, H., & Francis, C. (2020). Active versus passive fault-tolerant control of a redundant multirotor UAV. *Aeronautical Journal*, *124*(1273), 385–408.

Sanwale, J., Dahiya, S., Trivedi, P., & Kothari, M. (2023). Robust fault-tolerant adaptive integral dynamic sliding mode control using finite-time disturbance observer for coaxial octorotor UAVs. *Control Engineering Practice*, *135*, Article 105495.

Sato, M., Hardier, G., Ferreres, G., Edwards, C., Alwi, H., Chen, L., et al. (2018). Fault tolerance - background and recent trends. *Journal of the Society of Instrument and Control Engineers*, *57*(4), 279–286.

Schneider, T., Ducard, G., Rudin, K., & Strupler, P. (2012). Fault-tolerant control allocation for multirotor helicopters using parametric programming. In *International micro air vehicle conference and flight competition*.

Smaili, H., Breeman, J., Lombaerts, T., & Joosten, D. (2010). RECOVER: A benchmark for integrated fault tolerant flight control evaluation. In C. Edwards, T. Lombaerts, & H. Smaili (Eds.), *Fault tolerant flight control: A benchmark challenge* (pp. 171–221). Berlin, Heidelberg: Springer Berlin Heidelberg.

Tabor, A. (2022). *Volcano-observing drone flights open door to routine hazard monitoring*: *Technical report*, NASA Ames Research Center.

The MathWorks Inc. (2022). Dryden wind turbulence model (continuous), MATLAB reference pages. [Online] Available: https://uk.mathworks.com/help/aeroblks/drydenwindturbulencemodelcontinuous.html.

Vile, L., Alwi, H., Edwards, C., & Yates, N. (2021). Fault detection and fault tolerant control for a Prandtl wing UAV. In *Conf. control and fault-tolerant systems*.

Xiong, J.-J., Guo, N.-H., Mao, J., & Wang, H.-D. (2023). Self-tuning sliding mode control for an uncertain coaxial octorotor UAV. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *53*(2).

Yu, Y., & Dong, Y. (2019). Global fault-tolerant control of underactuated aerial vehicles with redundant actuators. *International Journal of Aerospace Engineering*, *2019*, 12.

Zeghlache, S., Mekki, H., Bouguerra, A., & Djerioui, A. (2018). Actuator fault tolerant control using adaptive RBFNN fuzzy sliding mode controller for coaxial octorotor UAV. *ISA Transactions*, *80*, 267–278.

Zhang, Y., & Jiang, J. (2008). Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, *32*(2), 229–252.