Sliding Mode Fault Tolerant Control of an Octorotor Using LPV Based Schemes

Halim Alwi and Christopher Edwards

Abstract—This paper presents two fault tolerant control (FTC) schemes for an octorotor UAV. The FTC schemes are based on an LPV system representation and utilizes a combination of sliding mode ideas and control allocation in order to take full advantage of the available redundant rotors in the octorotor configuration. A detailed synthesis procedure for the design of the two FTC schemes in the presence of uncertainty, as well as faults/failures, is presented. The first scheme is based on an online control allocation methodology where knowledge of the rotor effectiveness level has been used to redistribute the control signals to the healthy rotors. The second scheme assumes that this information is not available and uses a fixed control allocation structure even in the event of faults/failures. Although the synthesis process for the two schemes is different and they use different strategies to redistribute the control signals when faults/failures occur, both schemes involved the same ‘baseline’ (sliding mode) controller which does not need to be reconfigured. The difference is in the final physical control law where the control allocation matrix is defined. Simulation results on the full nonlinear octorotor model are presented for the two different schemes in the presence of uncertainty, sensor noise as well as faults/failures. The simulation results for various fault/failure scenarios show no visible degradation in state tracking performance, highlighting the potential of the proposed schemes.

Keyword: Fault tolerant control (FTC), multirotor UAV

I. INTRODUCTION

In recent years, research on fault tolerant control (FTC) has received significant attention from the control research community: partly driven by the aerospace industry. However, despite this attention and the fact that some of the schemes are becoming reasonably ‘mature’, there are two factors which are significant obstacles to further testing of state-of-the-art schemes – especially for aerospace applications: these are cost and safety. In response to these two factors, there has been significant interest in applying state-of-the-art FTC schemes to multi-rotor UAVs – particularly to the so-called quadrotor (see for example [1], [2], [3], [4], [5]). This is largely due to their simple build, relatively low cost, and the convenience of undertaking tests in a controlled safe laboratory environment. Apart from being used as a testbed for FTC schemes, quadrotors have also been considered as a popular platform for testing other advanced control schemes such as sliding mode control (SMC) and backstepping (see for example [6], [7], [8], [9], [10], [11]).

However, despite all the attention, a quadrotor is a limited testing platform for FTC schemes (as implicitly shown in the literature – most FTC schemes can only deal with partial rotor faults). This is due to its four rotor configuration which lacks redundancy. As a consequence, it cannot even deal with

the scenario in which one of its rotors totally fails (the only exception is the work of [12] which sacrifices yaw control in order to maintain roll and pitch control). A six (hexrotor [13], [14]) or eight (octorotor [15], [16], [17]) rotor configuration is a much more suitable platform for testing FTC schemes due to the extra available redundancy, which provides greater flexibility for testing. However, there is still only limited work in the literature which considers FTC on hex/octorotor platforms. The work in [18], [19] is the only recent research in this area. Interestingly, the use of octorotors has begun to attract interest from the general public for various applications – especially due to its high payload capacity. The most high profile example is the Amazon Prime Air facility [20], [21] to deliver packages quickly. However, to ensure that the packages are delivered on time and safely, FTC is an important aspect to be considered.

In (fixed wing) aerospace applications, one of the ongoing challenges is to ensure stability and performance over a wide set of flight conditions. A popular method especially in industry to achieve this is to consider gain scheduling. However one of the drawbacks of gain scheduling as pointed out in [22] is its ad-hoc design procedure and the lack of a formal proof of stability. However an extension to linear based design using a so-called linear parameter varying (LPV) representation has attracted a good deal of attention as it provides proof of stability over a wide range of operating conditions. However, despite this advantage there is only limited work which considers LPV based methods for multirotor UAVs: most of the existing work in the literature is based on linear designs (see for example [17], [23], [24]). The two most significant exceptions are the work on quadrotors by [25] (which considers robust $\mathcal{H}_\infty$ self scheduling control) and [26] (which considers a discrete LPV representation). However these papers do not deal with faults/failures or consider any fault tolerance.

In this paper sliding mode FTC schemes for an octorotor will be presented. The proposed FTC schemes are based on the nonlinear equations of motion represented as an LPV system. It is also worth mentioning that there is work on multirotor UAVs which considers the nonlinear equations of motion directly. However in this paper, the nonlinear equations have been rewritten exactly as an LPV model without requiring any linearizations. The LPV model representation allows the nonlinear equations of motion to be written in a ‘linear’ way thus facilitating the simplicity of a linear based design, while at the same time still retaining a true representation of the nonlinear equations of motion. The schemes proposed in this paper combine the robustness of sliding modes to deal with uncertainty and faults, and control allocation (CA) in order to take full advantage of the available redundant rotors and to redistribute the control signals to the ‘healthy’ rotors in

the event of faults/failures. The use of an LPV model to represent the actual nonlinear equations of motion is different to existing work in terms of the combination of SMC and CA (see for example [27]). Furthermore, in this paper, uncertainty in the thrust and drag factors is also considered during the synthesis of the controller. This has scarcely been discussed in the literature on multirotor UAVs – especially in the ones which consider fault tolerance.

In this paper two FTC schemes are considered. The first scheme corresponds to online control allocation where the rotor effectiveness levels are explicitly used as part of the control allocation scheme in order to redistribute the control signals to the remaining healthy rotors in the event of faults/failures. This scheme requires a fault detection and isolation (FDI) unit to provide information about the ‘health’ or effectiveness levels of each rotor. The second scheme assumes that the rotor effectiveness level information is not available and therefore resorts to a fixed control allocation structure.

The objective of the paper is to develop FTC schemes that can retain close to nominal fault free performance despite actuator faults and failures, and to identify the class of faults/failures which the scheme can tolerate. The contribution of the paper is as follows:

- it represents the first application of a sliding mode control allocation scheme for FTC of an octorotor
- the FTC schemes are based on the actual equations of motion which are re-written in an LPV framework, which allows ‘linear based’ design to be used whilst retaining a wider operating envelope. The LPV model has only four varying parameters which is comparable to the one in [25], but with a much simpler representation and without requiring linearization of the nonlinear equations of motion.
- two new LPV based schemes are presented which are different to the existing work on combining SMC with CA for FTC.

The paper is structured as follows. Section II presents the equations of motion for the octorotor considered in this paper. Section III presents the synthesis of the online CA scheme which utilizes the information about rotor effectiveness to redistribute the control signals to the remaining healthy rotors. In Section IV, it is assumed that the rotor effectiveness level is not known and therefore a fixed control allocation structure is presented. Section VI presents the designs for both schemes on the octorotor model, and finally Section VII presents the simulation results.

II. OCTOROTOR

Figure 1 shows the configuration of the octorotor considered in this paper. Unlike the quadrotor which only has four rotors, the octorotor has eight rotors equally spaced at 22.5deg and connected to the center by arms.

A. Nonlinear Equation of Motion

Similar to the assumptions made in [28], [17], [18], for design purposes, the model is simplified as follows:

- the octorotor is rigid and symmetric (i.e. the inertia matrix off-diagonal terms are zero).

Fig. 1. Octorotor configuration

- the thrust and drag are proportional to the square of the rotor speed (any variations can be treated as bounded uncertainty which can be dealt with during design);
- the hub forces and rolling moments are neglected;
- the propellers are rigid;
- ground effect is neglected;
- each motor reacts immediately to the commanded rotor speed (i.e. motor dynamics are ignored).

Based on the above assumptions, the octorotor is governed by the following equations of motion:

$$\dot{X}(t) = \begin{bmatrix} \dot{x}_b(t) \\ \dot{y}_b(t) \\ \dot{z}_b(t) \\ a_\phi(X) \\ a_\theta(X) \\ a_\psi(X) \\ b_x(X) \frac{1}{m} \tau_1(t) \\ b_y(X) \frac{1}{m} \tau_1(t) \\ g - b_z(X) \frac{1}{m} \tau_1(t) \\ \frac{I_{yy} - I_{zz}}{I_{xx}} q(t) r(t) + \frac{J_r}{I_{xx}} q(t) \Omega_r(t) + \frac{l}{I_{xx}} \tau_2(t) \\ \frac{I_{zz} - I_{xx}}{I_{yy}} p(t) r(t) - \frac{J_r}{I_{yy}} p(t) \Omega_r(t) + \frac{l}{I_{yy}} \tau_3(t) \\ \frac{I_{xx} - I_{yy}}{I_{zz}} q(t) p(t) + \frac{1}{I_{zz}} \tau_4(t) \end{bmatrix} \quad (1)$$

where the terms

$$a_\phi(X) = p(t) + q(t) \sin(\phi) \tan(\theta) + r \cos(\phi) \tan(\theta) \quad (2)$$

$$a_\theta(X) = q(t) \cos(\phi) - r(t) \sin(\theta) \quad (3)$$

$$a_\psi(X) = q(t) \sin(\phi) \sec(\theta) + r(t) \cos(\phi) \sec(\theta) \quad (4)$$

and

$$b_x(X) = \cos(\phi) \sin(\theta) \cos(\psi) + \sin(\phi) \sin(\psi) \quad (5)$$

$$b_y(X) = \cos(\phi) \sin(\theta) \cos(\psi) - \sin(\phi) \cos(\psi) \quad (6)$$

$$b_z(X) = \cos(\phi) \cos(\theta) \quad (7)$$

The state $X(t)$ is given by

$$X(t) = \begin{bmatrix} x_b(t) & y_b(t) & z_b(t) & \phi(t) & \theta(t) & \psi(t) \\ \dot{x}_b(t) & \dot{y}_b(t) & \dot{z}_b(t) & p(t) & q(t) & r(t) \end{bmatrix}^T \quad (8)$$

which represents position in the body x, y, z axes; roll angle, pitch angle, yaw angle, velocity in the body x, y, z axes; and

roll rate, pitch rate and yaw rate respectively. The variable $\Omega_r(t)$ in (1) is defined by

$$\Omega_r(t) := -\Omega_1 - \Omega_2 + \Omega_3 + \Omega_4 - \Omega_5 - \Omega_6 + \Omega_7 + \Omega_8 \quad (9)$$

which represents the overall residual propeller angular speed [28], while $\Omega_1, \dots, \Omega_8$ are the individual propeller angular rates. The other system constant parameters are defined in Table I.

symbol	parameter
m	mass
I_{xx}, I_{yy}, I_{zz}	inertia on the x,y,z axis
b, d	thrust and drag factor
J_r	rotor inertia
L, l	arm and moment arm length ($l = L \cos(\pi/8)$)
g	gravitational constant

TABLE I
OCTOROTOR PARAMETER DEFINITIONS

For simplicity, equation (1) can be written as

$$\dot{X}(t) = \underbrace{\begin{bmatrix} \dot{x}_b(t) \\ \dot{y}_b(t) \\ \dot{z}_b(t) \\ a_\phi(X) \\ a_\theta(X) \\ a_\psi(X) \\ 0 \\ 0 \\ g \\ c_2 q(t)r(t) + c_3 q(t)\Omega_r(t) \\ c_5 p(t)r(t) - c_6 p(t)\Omega_r(t) \\ c_8 p(t)q(t) \end{bmatrix}}_{f(t,X)} + \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ b_x(X)c_1 & 0 & 0 & 0 & 0 \\ b_y(X)c_1 & 0 & 0 & 0 & 0 \\ -b_z(X)c_1 & 0 & 0 & 0 & 0 \\ 0 & c_4 & 0 & 0 & 0 \\ 0 & 0 & c_7 & 0 & 0 \\ 0 & 0 & 0 & c_9 & 0 \end{bmatrix}}_{g(t,X)} \underbrace{\begin{bmatrix} \tau_1(t) \\ \tau_2(t) \\ \tau_3(t) \\ \tau_4(t) \end{bmatrix}}_{\tau(t)} \quad (10)$$

where the constant parameters are given by

$$c_1 = 1/m \quad (11)$$

$$c_2 = (I_{yy} - I_{zz})/I_{xx}, c_3 = J_r/I_{xx}, c_4 = 1/I_{xx} \quad (12)$$

$$c_5 = (I_{zz} - I_{xx})/I_{yy}, c_6 = J_r/I_{yy}, c_7 = 1/I_{yy} \quad (13)$$

$$c_8 = (I_{xx} - I_{yy})/I_{zz}, c_9 = 1/I_{zz} \quad (14)$$

The inputs to the octorotor $\tau(t)$ defined in (10) represent the total thrust, roll torque, pitch torque and yaw torque respectively. Since it is assumed that the thrust and drag are proportional to the square of the rotor speed, the torque vector $\tau(t)$ can be mapped back into the individual rotor contributions given in (15) where b and d are the (fixed) thrust and drag factors respectively [28].

The scalars $\Delta_b(t)$ and $\Delta_d(t)$ represent the uncertainty/imperfections in the thrust and drag factors and are assumed to satisfy

$$|\Delta_b(t)| < \bar{\Delta}, \quad |\Delta_d(t)| < \bar{\Delta} \quad (16)$$

where $\bar{\Delta}$ is a positive scalar and therefore $\|\Delta(t)\| < \bar{\Delta}$. Note that nominally $\Delta_b(t) = \Delta_d(t) = 0$.

B. LPV form

Similar to [28], [17], for design purposes, only a subset of the states $X(t)$ in (8) will be considered, specifically

$$x(t) = [z_b(t) \quad \phi(t) \quad \theta(t) \quad \psi(t) \quad \dot{z}_b(t) \quad p(t) \quad q(t) \quad r(t)]^T \quad (17)$$

Note that the states x_b, y_b and their derivatives $\dot{x}_b, \dot{y}_b$ are not included in (17) for the design of the inner loop (rate) controller. This is due to the fact that typically the x_b and y_b position of the octorotor can be controlled using the pitch and roll angles as manipulated variables which can be controlled directly the pitch and roll torque $\tau_2(t), \tau_3(t)$ as shown in (1) [28], [17].

To further assist in the controller synthesis, it will be assumed that $(\phi, \theta, \psi) \approx (p, q, r)$, i.e. the perturbations from hover are small. Therefore as described in [28], the nonlinear equations in (10) associated with the states in (17) can be written in a ‘pseudo-linear’ structure as

$$\dot{x}(t) = A(\rho)x(t) + B_\tau(\rho)\tau(t) + E(\rho)\zeta(t) \quad (18)$$

where the time varying parameters $\rho(t)$ (which represent the LPV scheduling parameters) are defined as

$$\rho(t) := [\rho_1 \quad \rho_2 \quad \rho_3 \quad \rho_4] = [p(t) \quad q(t) \quad r(t) \quad b_z(X)] \quad (19)$$

The scheduling gain $\rho(t)$ is assumed to lie in a specific compact set, $\Theta \in \mathbb{R}^4$ while $b_z(X)$ is defined in (7). During the controller synthesis, the signal $\zeta(t)$ in (18) is given as

$$\zeta(t) = [\Omega_r(t) \quad g]^T \quad (20)$$

and will be considered as a disturbance¹. The system matrices

¹Of course if the $\Omega_i(t)$ are measured accurately, then the ‘uncertainty’ $\zeta(t)$ could be compensated for in a feedforward way – especially in terms of maintaining a sliding mode.

$$\underbrace{\begin{bmatrix} \tau_1(t) \\ \tau_2(t) \\ \tau_3(t) \\ \tau_4(t) \end{bmatrix}}_{\tau(t)} = \underbrace{\begin{bmatrix} 1 + \Delta_b(t) & 0 & 0 & 0 \\ 0 & 1 + \Delta_b(t) & 0 & 0 \\ 0 & 0 & 1 + \Delta_b(t) & 0 \\ 0 & 0 & 0 & 1 + \Delta_d(t) \end{bmatrix}}_{I + \Delta(t)} \underbrace{\begin{bmatrix} b & b & b & b & b & b & b & b \\ 0 & 0 & -bl & -bl & 0 & 0 & bl & bl \\ bl & bl & 0 & 0 & -bl & -bl & 0 & 0 \\ -d & -d & d & d & -d & -d & d & d \end{bmatrix}}_{B_\Omega} \underbrace{\begin{bmatrix} \Omega_1(t)^2 \\ \Omega_2(t)^2 \\ \vdots \\ \Omega_8(t)^2 \end{bmatrix}}_{u(t)} \quad (15)$$

in (18) are given by

$$\begin{aligned}
 A(\rho) &= \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{c_2}{2}\rho_3 & \frac{c_2}{2}\rho_2 \\ 0 & 0 & 0 & 0 & 0 & \frac{c_5}{2}\rho_3 & 0 & \frac{c_5}{2}\rho_1 \\ 0 & 0 & 0 & 0 & 0 & \frac{c_8}{2}\rho_2 & \frac{c_8}{2}\rho_1 & 0 \end{bmatrix}, \\
 B_\tau(\rho) &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\rho_4 c_1 & 0 & 0 & 0 \\ 0 & c_4 & 0 & 0 \\ 0 & 0 & c_7 & 0 \\ 0 & 0 & 0 & c_9 \end{bmatrix}, \\
 E(\rho) &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ c_3 \rho_2 & 0 \\ -c_6 \rho_1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ E_2(\rho) \end{bmatrix} \quad (21)
 \end{aligned}$$

Note that due to the order of the states in (17), the system matrices in (21) are already in regular form [29], [27] which will assist in the sliding mode design as well as in the control allocation scheme. Using (19), the matrices in (21) can be represented in LPV form given by

$$\begin{aligned}
 A(\rho) &= A_0 + A_1 \rho_1 + A_2 \rho_2 + A_3 \rho_3 \\
 B_\tau(\rho) &= B_{\tau,0} + B_{\tau,4} \rho_4 \\
 E(\rho) &= E_0 + E_1 \rho_1 + E_2 \rho_2 \quad (22)
 \end{aligned}$$

Using the force/torque-rotor speed mapping given in (15), equation (18) can also be represented by

$$\begin{aligned}
 \dot{x}(t) &= A(\rho)x(t) + B_\tau(\rho)(I + \Delta(t))B_\Omega u(t) + E(\rho)\zeta(t) \\
 &= A(\rho)x(t) + (B(\rho) + B^\delta(\rho))u(t) + E(\rho)\zeta(t) \quad (23)
 \end{aligned}$$

where $B(\rho) = B_\tau(\rho)B_\Omega$ and $B^\delta(\rho) = B_\tau(\rho)\Delta(t)B_\Omega$. The system in (23) has 8 inputs $u(t)$, corresponding to the signals sent to each of the individual rotors. Equation (23) represents a system with redundancy – which is essential for fault tolerant control – and will be exploited for the controller synthesis. Based on the structure of B_Ω , and $B_\tau(\rho)$ in (15) and (21), $B(\rho)$ and $B^\delta(\rho)$ in (23) can be perfectly factorized into fixed and varying components given by

$$B(\rho) = B_\nu B_2(\rho), \quad B^\delta(\rho) = B_\nu \Delta(t) B_2(\rho) \quad (24)$$

where B_ν and B_2 are given in (25).

Due to the perfect factorization in (24), the system in (23) has a standard structure which can be utilized for FTC using the concept of control allocation [30], [31].

Remark: The factorization of $B^\delta(\rho)$ in (24) exploits the diagonal nature of $\Delta(t)$ in (15) and the diagonal form of the lower sub-block of $B_\tau(\rho)$ in (21).

Remark: Similar to the assumption made in [9], [8], [10], here it is assumed that the octorotor is limited from engaging in a sustained $\pi/2$ pitch or roll angle orientation i.e. $-\pi/2 < \phi < \pi/2$ and $-\pi/2 < \theta < \pi/2$ (rad). This assumption is made to ensure that $\det(B_2(\rho)B_2^T(\rho)) \neq 0$, since from (25), $B_2(\rho)$ depends on the LPV parameter $\rho_4 = \cos(\phi)\cos(\theta)$.

Remark: Assuming $\det(B_2(\rho)B_2^T(\rho)) \neq 0$ is equivalent to $\text{rank}(B_2(\rho)B_2^T(\rho)) = 4$ and relates to the physical requirement of maintaining 4 independent rotors (out of the original 8 rotors), so that 4 independent outputs can be controlled.

III. ONLINE CONTROL ALLOCATION

The objective of this section is to develop controllers which can maintain close to nominal performance in the event of actuator faults. The control objective is to control altitude, roll, pitch and yaw angles. In this system at least four rotors need to be fault-free, and a condition will be developed capturing the class of faults/failures for which the control laws are applicable. Throughout the paper it is assumed that full state information is available to the controller.

A. Actuator faults/failures

When a fault/failure occurs in any of the rotors, equation (23) can be represented as

$$\begin{aligned}
 \dot{x}(t) &= A(\rho)x(t) + (B(\rho) + B^\delta(\rho))(I_8 - K(t))u(t) \\
 &\quad + E(\rho)\zeta(t) \quad (26)
 \end{aligned}$$

where the diagonal matrix $K(t) = \text{diag}(k_1(t) \dots k_8(t))$ models the reduction of effectiveness level in the individual rotors. The scalars $k_i(t)$ for $i = 1 \dots 8$ satisfy $0 \leq k_i(t) \leq 1$ where $k_i(t) = 0$ represents the fault free situation, $k_i(t) > 0$ indicates that the i th rotor contains some faults and $k_i(t) = 1$ indicates that the i th rotor has totally failed.

Note that due to the structure of $B(\rho)$, $B^\delta(\rho)$ and $E(\rho)$ in (24), (25) and (21), rotor faults, uncertainty and disturbances which are represented by $K(t)u(t)$, $\Delta(t)$ and $\zeta(t)$ in (26) are classified as ‘matched uncertainty’ [29], [27] in the sliding mode literature. One of the well known properties of sliding modes is its inherent robustness against ‘matched uncertainty’: and in fact such effects in theory are completely rejected by sliding mode controllers. In the event of a total rotor failure, provided that enough redundancy is still available in the system i.e.

$$\det(B_2(\rho)(I - K(t))B_2^T(\rho)) \neq 0 \quad \text{for all } \rho \in \Theta \quad (27)$$

$$B_\nu = \begin{bmatrix} 0 \\ I_4 \end{bmatrix}, \quad B_2(\rho) = \begin{bmatrix} -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b & -\rho_4 c_1 b \\ 0 & 0 & -c_4 b l & -c_4 b l & 0 & 0 & c_4 b l & c_4 b l \\ c_7 b l & c_7 b l & 0 & 0 & -c_7 b l & -c_7 b l & 0 & 0 \\ -c_9 d & -c_9 d & c_9 d & c_9 d & -c_9 d & -c_9 d & c_9 d & c_9 d \end{bmatrix} \quad (25)$$

control allocation can be used to redistribute the control signals to the remaining healthy rotors in order to maintain the same level of performance as compared to the fault free case.

For the control allocation analysis which follows, define $\nu(t) \in \mathbb{R}^4$ as a virtual control signal (to control altitude, roll, pitch and yaw angles) given by

$$\nu(t) := B_2(\rho)u(t) \quad (28)$$

where $B_2(\rho)$ is defined in (25). The actual control signal $u(t) \in \mathbb{R}^8$ which will be sent to all eight rotors is obtained by manipulating (28) to give

$$u(t) = B_2^\dagger(\rho)\nu(t) \quad (29)$$

In (29), the term $B_2^\dagger(\rho)$ represents the right pseudo inverse of $B_2(\rho)$ which is defined as

$$B_2^\dagger(\rho) := \Lambda(t)B_2^T(\rho)(B_2(\rho)\Lambda(t)B_2^T(\rho))^{-1} \quad (30)$$

where $\Lambda(t) = \text{diag}(\Lambda_1(t) \dots \Lambda_8(t))$ is a diagonal matrix with positive elements.

In order to deal with total rotor failure, the following analysis will look at two different control allocation strategies: namely online and fixed control allocation. The difference between the two strategies will determine by how $\Lambda(t)$ in (30) is chosen.

The idea for the online control allocation is to incorporate the information available from an FDI unit as part of the control signal redistribution in the control allocation matrix in (30). To achieve this, define

$$W(t) := I_8 - K(t) \quad (31)$$

For online control allocation, the weight $\Lambda(t)$ in (30) is chosen as

$$\Lambda(t) := W(t) \quad (32)$$

Using (31) and (32) plus the fact that $B(\rho) = B_\nu B_2(\rho)$ and $B^\delta(\rho) = B_\nu \Delta(t) B_2(\rho)$ from (24), and using (28)-(29), equation (26) can be written as

$$\begin{aligned} \dot{x}(t) &= A(\rho)x(t) + (B(\rho) + B^\delta(\rho))Wu(t) + E(\rho)\zeta(t) \\ &= A(\rho)x(t) + B_\nu B_2(\rho)W(t)B_2^\dagger(\rho)\nu(t) \\ &\quad + B^\delta(\rho)Wu(t) + E(\rho)\zeta(t) \\ &= A(\rho)x(t) + B_\nu \hat{\nu}(t) + \underbrace{\begin{bmatrix} B_\nu \Delta(t) & E(\rho) \end{bmatrix}}_{D(\rho)} \underbrace{\begin{bmatrix} \hat{\nu}(t) \\ \zeta(t) \end{bmatrix}}_{\xi(t)} \end{aligned} \quad (33)$$

where $\hat{\nu}(t)$ is defined as

$$\hat{\nu}(t) := B_2(\rho)W^2(t)B_2^T(\rho)(B_2(\rho)W(t)B_2^T(\rho))^{-1}\nu(t) \quad (34)$$

Remark: Note that the first component of $\xi(t)$ in (33) is associated with input multiplicative uncertainty. A limitation on $\|\Delta(t)\|$ must therefore be assumed so that this uncertainty is not destabilizing [32] (see equation (39)).

B. Sliding Mode

For the synthesis of the control law, equation (33) will be considered. Due to the structure of $B_\nu(\rho)$ and $E(\rho)$ in (25) and (21) and the uncertainty distribution matrix

$$D(\rho) = \begin{bmatrix} 0 \\ D_2(\rho) \end{bmatrix} \quad (35)$$

where $D_2(\rho) = \begin{bmatrix} \Delta(t) & E_2(\rho) \end{bmatrix}$. Equation (33) can be written in detail as

$$\underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} A_{11}(\rho) & A_{12}(\rho) \\ A_{21}(\rho) & A_{22}(\rho) \end{bmatrix}}_{A(\rho)} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} 0 \\ I_4 \end{bmatrix}}_{B_\nu} \hat{\nu}(t) + \underbrace{\begin{bmatrix} 0 \\ D_2(\rho) \end{bmatrix}}_{D(\rho)} \xi(t) \quad (36)$$

where the states have been partitioned as

$$\begin{aligned} x_1(t) &= \begin{bmatrix} z_b(t) & \phi(t) & \theta(t) & \psi(t) \end{bmatrix}^T, \\ x_2(t) &= \begin{bmatrix} \dot{z}_b(t) & p(t) & q(t) & r(t) \end{bmatrix}^T \end{aligned} \quad (37)$$

The signal $\xi(t)$ is assumed to be bounded and satisfies

$$\|\xi(t)\| \leq \gamma \|\hat{\nu}(t)\| + \alpha(t, x) \quad (38)$$

where γ is a known positive constant, $\alpha(\cdot)$ is a known function, and $\hat{\nu}(t)$ is defined in (34). Define

$$d_2 := \max_{\rho \in \Theta, \|\Delta\| < \bar{\Delta}} \|D_2(\rho)\| \quad (39)$$

and assume that $\gamma d_2 < 1$. Here the known quantities γ and $\alpha(t, x)$ would need to be established using engineering intuition and judgement regarding the particular octorotor system². Also note that due to the structure of B_ν from (25), equation (36) is already in 'regular form' [29], [27] which is suitable for sliding mode controller synthesis.

The first step in designing a sliding mode controller is to design the sliding hyperplane which will determine the closed-loop performance of the system. First define a switching function $s(t)$ for the fault free condition as

$$s(t) = Sx(t) \quad (40)$$

where

$$S = \begin{bmatrix} M & I_4 \end{bmatrix} \quad (41)$$

and $M \in \mathbb{R}^{4 \times 4}$ represents the design freedom which will be used to determine the system performance. Define the hyperplane as $\mathcal{S} := \{x \in \mathbb{R}^8 : Sx = 0\}$. If a control law can be designed such that the closed-loop trajectories of the system reach the hyperplane and subsequently remain there, an ideal sliding mode is said to have been achieved [29], [27]. During an ideal sliding motion, the closed loop system will be insensitive to matched uncertainties and behaves like a reduced order system [29], [27]. To analyse the performance of the closed-loop reduced order sliding motion, a coordinate transformation $(x_1, x_2) \mapsto (x_1, s)$ is introduced. In this new coordinate system, equation (36) becomes

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{s}(t) \end{bmatrix} = \begin{bmatrix} \hat{A}_{11}(\rho) & \hat{A}_{12}(\rho) \\ \hat{A}_{21}(\rho) & \hat{A}_{22}(\rho) \end{bmatrix} \begin{bmatrix} x_1(t) \\ s(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I_4 \end{bmatrix} \hat{\nu}(t) + \begin{bmatrix} 0 \\ D_2(\rho) \end{bmatrix} \xi(t) \quad (42)$$

where

$$\begin{aligned} \hat{A}_{11}(\rho) &:= A_{11}(\rho) - A_{12}(\rho)M, \\ \hat{A}_{21}(\rho) &:= M\hat{A}_{11}(\rho) + A_{21}(\rho) - A_{22}(\rho)M, \\ \hat{A}_{22}(\rho) &:= MA_{12}(\rho) + A_{22}(\rho) \end{aligned} \quad (43)$$

²That said, in the simulation in Section VI, very crude bounds on the uncertainty will be adopted.

and $\hat{A}_{21}(\rho) = A_{21}(\rho)$. During ideal sliding $\dot{s}(t) = s(t) = 0$ and from the top partition of (42), the reduced order sliding motion is governed by

$$\dot{x}_1(t) = \hat{A}_{11}(\rho)x_1(t) \quad (44)$$

Therefore $\hat{A}_{11}(\rho) := A_{11}(\rho) - A_{12}(\rho)M$ must be made stable by choice of M .

C. Control law

The next step after designing the sliding hyperplane is to choose a control law in such a way that the hyperplane is reached in finite time and sliding is maintained. In this paper the proposed control law contains two distinct components: a 'linear' and 'nonlinear' component defined as

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

The 'linear' component is akin to a standard state feedback law and here is given by

$$\hat{v}_l(\rho) = -\hat{A}_{21}(\rho)x_1(t) - (\hat{A}_{22}(\rho) - \Phi)s(t) \quad (46)$$

where the matrix $\Phi \in \mathbb{R}^{4 \times 4}$ is chosen to be Hurwitz. The 'nonlinear' component is

$$\hat{v}_n(t) = -\mathcal{K}(t, x) \frac{P_2 s(t)}{\|P_2 s(t)\|} \quad \text{if } s(t) \neq 0 \quad (47)$$

where the symmetric positive definite matrix (s.p.d) $P_2 \in \mathbb{R}^{4 \times 4}$ satisfies the following Lyapunov equation

$$P_2 \Phi + \Phi^T P_2 = -I_4 \quad (48)$$

Based on the analysis above, instead of analysing the stability of the closed system with matched uncertainty, the problem becomes one of ensuring that sliding occurs despite the presence of uncertainty. This will be justified by the following proposition and its proof:

Proposition 1: If the design matrix M has been chosen such that $\hat{A}_{11}(\rho)$ in (43) is stable, then choosing

$$\mathcal{K}(t, x) \geq \frac{d_2 (\gamma \|\hat{v}_l(\rho)\| + \alpha(t, x)) + \eta}{(1 - \gamma d_2)} \quad (49)$$

ensures a sliding motion takes place on $\mathcal{S}$ in finite time. In (49) the scalar d_2 is defined in (39), the variable η is a positive design scalar and γ is a known constant from (38).

Proof:

Substituting the control law in (45)-(47) into (42) yields

$$\dot{x}_1(t) = \hat{A}_{11}(\rho)x_1(t) + \hat{A}_{12}(\rho)s(t) \quad (50)$$

$$\dot{s}(t) = \Phi s(t) - \mathcal{K}(t, x) \frac{P_2 s}{\|P_2 s\|} + D_2(\rho)\xi(t) \quad (51)$$

For equation (51), consider a single quadratic Lyapunov function [33] given by $V(s) = s^T P_2 s$. Using the fact that $s^T P_2 P_2 s = \|P_2 s\|^2$ and $\Phi^T P_2 + P_2 \Phi = -I$, differentiating the Lyapunov function yields:

$$\begin{aligned} \dot{V} &= s^T (\Phi^T P_2 + P_2 \Phi) s - 2\mathcal{K} \frac{1}{\|P_2 s\|} (s^T P_2 P_2 s) \\ &\quad + 2s^T P_2 D_2(\rho)\xi(t) \\ &= -\|s\|^2 - 2\mathcal{K} \|P_2 s\| + 2s^T P_2 D_2(\rho)\xi(t) \end{aligned} \quad (52)$$

Furthermore, since

$$\|s^T P_2 D_2(\rho)\xi(t)\| < \|P_2 s\| d_2 \|\xi(t)\|$$

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

$$\dot{V} \leq -\|s\|^2 - 2\|P_2 s\| (\mathcal{K} - d_2 \|\xi(t)\|) \quad (53)$$

To assist in the following analysis, the gain $\mathcal{K}$ in (53) will be expressed in terms of the uncertainty ξ . First rewrite equation (49) as

$$\mathcal{K}(t, x) (1 - \gamma d_2) \geq d_2 (\gamma \|\hat{v}_l(\rho)\| + \alpha(t, x)) + \eta \quad (54)$$

Rearranging this equation in terms of $\mathcal{K}(t, x)$ gives the expression

$$\begin{aligned} \mathcal{K}(t, x) &\geq d_2 (\gamma \|\hat{v}_l(\rho)\| + \alpha(t, x)) + \eta + \mathcal{K}(t, x) \gamma d_2 \\ &\geq d_2 (\gamma \|\hat{v}_l(\rho)\| + \mathcal{K}(t, x) \gamma + \alpha(t, x)) + \eta \\ &\geq d_2 (\gamma \|\hat{v}(t)\| + \alpha(t, x)) + \eta \\ &\geq d_2 \|\xi(t)\| + \eta \end{aligned} \quad (55)$$

Substituting equation (55) in (53) yields

$$\dot{V} \leq -\|s\|^2 - 2\eta \|P_2 s\| \quad (56)$$

Equation (56) 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

$$\begin{aligned} \|P_2 s\|^2 &= (P_2^{1/2} s)^T P_2 (P_2^{1/2} s) \geq \lambda_{\min}(P_2) \|P_2^{1/2} s\|^2 \\ &= \lambda_{\min}(P_2) V(s) \end{aligned} \quad (57)$$

Substituting (57) into (56) yields

$$\dot{V} \leq -2\eta \sqrt{\lambda_{\min}(P_2)} \sqrt{V} \quad (58)$$

By integrating (58), 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

$$t_s \leq \eta^{-1} \sqrt{V(s_0)/\lambda_{\min}(P_2)} \quad (59)$$

where s_0 represents the initial value of switching function $s(t)$ at time $t = 0$ [29]. ■

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

D. Design of matrix M

Generically, the parameter M , can be designed using an LMI based approach based on a single quadratic Lyapunov function [33] according to

$$\hat{A}_{11}^T P_1 + P_1 \hat{A}_{11} < 0 \quad (60)$$

$$P_1 > 0 \quad (61)$$

where $\hat{A}_{11}(\rho) = A_{11}(\rho) - A_{12}(\rho)M$ as defined in (43). The standard MATLAB LMI multi-model state feedback synthesis code 'msfsyn' [34] can be used to obtain the solution for P_1 and the design matrix M . In the specific octorotor example considered in this paper, since the system has been partition according to (36) and (21), the sub-partition matrices $A_{11}(\rho) = 0$ and $A_{12}(\rho) = I_4$. Since both $A_{11}(\rho)$ and $A_{12}(\rho)$ are fixed, any simple linear based design can be used to choose M in order to stabilize $\hat{A}_{11}(\rho)$. In the design example which will be shown later, simple pole placement will be used.

E. Final Control Law

From (34) the virtual control

$$\nu(t) = B_2(\rho)W(t)B_2^T(\rho)(B_2(\rho)W^2(t)B_2^T(\rho))^{-1}\hat{\nu}(t) \quad (62)$$

Manipulating (29), (30), (31)-(32) and (45)-(47) yields an expression for $u(t)$ (which is the actual control signal sent to all the rotors) of the form

$$u(t) = W(t)B_2^T(\rho)(B_2(\rho)W^2(t)B_2^T(\rho))^{-1} \times \left(-\hat{A}_{21}(\rho)x_1(t) - (\hat{A}_{22}(\rho) - \Phi)s(t) - \mathcal{K}(\cdot) \frac{P_2 s(t)}{\|P_2 s(t)\|} \right) \quad (63)$$

where the switching function $s(t)$ is defined in (40). For the online control allocation scheme described in this section, the final control law (63) requires an FDI scheme because (63) contains information about each of the rotor effectiveness levels $w_i(t)$ through the diagonal matrix $W(t)$.

Remark: Equation (63) shows that for all combinations of faults/failures which satisfy $\det(B_2(\rho)W(t)B_2^T(\rho)) \neq 0$ for all $\rho \in \Theta$ (i.e. there is enough redundancy in the system), the control law in (63) can redistribute the control signals to the healthy rotors and maintain the same level of performance as the fault-free case.

Remark: In practice $W(t)$ in (63) must be replaced by an estimate $\hat{W}(t)$ from an FDI scheme and therefore

$$u(t) = \hat{W}(t)B_2^T(\rho)(B_2(\rho)\hat{W}^2(t)B_2^T(\rho))^{-1}(\hat{\nu}_l(t) + \hat{\nu}_n(t)) \quad (64)$$

where $\hat{\nu}_l(t)$ and $\hat{\nu}_n(t)$ are defined in (46) and (47). Suppose $W(t) = \hat{W}(t)(I + \delta(t))$ where $\delta(t)$ is a diagonal matrix capturing the difference between $W(t)$ and $\hat{W}(t)$, then using (64) it follows

$$\begin{aligned} \dot{s}(t) &= \hat{A}_{21}(\rho)x_1(t) + \hat{A}_{22}(\rho)s(t) \\ &\quad + B_2(\rho)W(t)u(t) + D_2(\rho)\xi(t) \\ &= \hat{A}_{21}(\rho)x_1(t) + \hat{A}_{22}(\rho)s(t) + \hat{\nu}_l(t) + \hat{\nu}_n(t) \\ &\quad + \tilde{\delta}(t)(\hat{\nu}_l(t) + \hat{\nu}_n(t)) + D_2(\rho)\xi(t) \\ &= \Phi s(t) - \mathcal{K}(t, x) \frac{P_2 s}{\|P_2 s\|} + D_2(\rho)\xi(t) \\ &\quad + \tilde{\delta}(t)(\hat{\nu}_l(t) + \hat{\nu}_n(t)) \\ &\leq \Phi s(t) - \mathcal{K}(t, x) \left(1 - \|\tilde{\delta}(t)\|\right) \frac{P_2 s}{\|P_2 s\|} \\ &\quad + D_2(\rho)\xi(t) + \tilde{\delta}(t)\hat{\nu}_l(t) \end{aligned} \quad (65)$$

where

$$\tilde{\delta}(t) = B_2(\rho)\hat{W}(t)\delta(t)\hat{W}(t)B_2^T(\rho)(B_2(\rho)\hat{W}^2(t)B_2^T(\rho))^{-1} \quad (66)$$

And so provided $\delta(t)$ is small enough so that $\|\tilde{\delta}(t)\| < 1$ sliding can still be maintained.

IV. FIXED CONTROL ALLOCATION

The online control allocation scheme in Section III assumed that an FDI scheme is available and took advantage of this information as part of the control allocation unit. In this section, it will be assumed that an FDI unit is not available and therefore the control allocation will have a 'fixed' structure, and the final control law expression will be independent of any knowledge of the rotor effectiveness level.

In this section, for the potentially faulty system in (26)-(30), the weight $\Lambda(t)$ in the pseudo inverse $B_2^\dagger(30)$ is chosen as

$$\Lambda := I_8 \quad (67)$$

and therefore (30) becomes

$$B_2^\dagger(\rho) := B_2^T(\rho)(B_2(\rho)B_2^T(\rho))^{-1} \quad (68)$$

Using (24), (28)-(29) and (31), equation (26) can be rewritten as

$$\begin{aligned} \dot{x}(t) &= A(\rho)x(t) + B(\rho)(I - K(t))u(t) \\ &\quad + \underbrace{\left[B_\nu \Delta(t)B_2(\rho)W(t)B_2^\dagger(\rho) \quad E(\rho) \right]}_{\check{D}(\rho)} \underbrace{\begin{bmatrix} \nu(t) \\ \xi(t) \end{bmatrix}}_{\check{\xi}(t)} \\ &= A(\rho)x(t) + B_\nu \left(I - B_2(\rho)K(t)B_2^\dagger(\rho) \right) \nu(t) \\ &\quad + \check{D}(\rho)\check{\xi}(t) \end{aligned} \quad (69)$$

Note that due to the structure of B_ν and $E(\rho)$ in (25) and (21), the term $\check{D}(\rho)$ in (69) can be written as

$$\check{D}(\rho) = \begin{bmatrix} 0 \\ \check{D}_2(\rho) \end{bmatrix} \quad (70)$$

where $\check{D}_2(\rho) = \begin{bmatrix} \Delta(t)B_2(\rho)W(t)B_2^\dagger(\rho) & E_2(\rho) \end{bmatrix}$.

A. Sliding Mode

For the synthesis of the control law, equation (69) can be written in detail as

$$\begin{aligned} \underbrace{\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix}}_{\dot{x}(t)} &= \underbrace{\begin{bmatrix} A_{11}(\rho) & A_{12}(\rho) \\ A_{21}(\rho) & A_{22}(\rho) \end{bmatrix}}_{A(\rho)} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} 0 \\ I_4 \end{bmatrix}}_{B_\nu} \nu(t) \\ &\quad - \begin{bmatrix} 0 \\ B_2(\rho)K(t)B_2^\dagger(\rho) \end{bmatrix} \nu(t) + \begin{bmatrix} 0 \\ \check{D}_2(\rho) \end{bmatrix} \check{\xi}(t) \end{aligned} \quad (71)$$

Also note that (71) has the same state partition as in (36) as it is based on the same partition in (21). Using the same coordinate transformation as in (42), equation (71) can be written in the 'regular form' coordinates as

$$\begin{aligned} \begin{bmatrix} \dot{x}_1(t) \\ \dot{s}(t) \end{bmatrix} &= \begin{bmatrix} \hat{A}_{11}(\rho) & \hat{A}_{12}(\rho) \\ \hat{A}_{21}(\rho) & \hat{A}_{22}(\rho) \end{bmatrix} \begin{bmatrix} x_1(t) \\ s(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I_4 \end{bmatrix} \nu(t) \\ &\quad - \begin{bmatrix} 0 \\ B_2(\rho)K(t)B_2^\dagger(\rho) \end{bmatrix} \nu(t) + \begin{bmatrix} 0 \\ \check{D}_2(\rho) \end{bmatrix} \check{\xi}(t) \end{aligned} \quad (72)$$

where $\hat{A}_{11}(\rho)$, $\hat{A}_{21}(\rho)$ and $\hat{A}_{22}(\rho)$ are defined in (43) and $\hat{A}_{11}(\rho)$ is stable by choice of M . Since (72) is in the same coordinate system as in (42), the approach described in Section III-D can be also used to design the matrix M for the fixed control allocation. The controller synthesis and a stability analysis for the faulty system using the fixed control allocation scheme will be discussed in the following subsection.

B. Control law

Using (68), the lower equations of (72) can be written as

$$\begin{aligned} \dot{s}(t) &= \hat{A}_{21}(\rho)x_1(t) + \hat{A}_{22}(\rho)s(t) + \check{D}_2(\rho)\check{\xi}(t) \\ &\quad + \nu + B_2(\rho)K(t)B_2^\dagger(\rho)(B_2(\rho)B_2^T(\rho))^{-1}\nu(t) \end{aligned} \quad (73)$$

where the uncertainty $\check{\xi}(t)$ is assumed to satisfy a bound (similar to the one in (38)):

$$\|\check{\xi}(t)\| \leq \tilde{\gamma}\|\nu(t)\| + \check{\alpha}(t, x) \quad (74)$$

where $\tilde{\gamma}$ is a known positive scalar and $\check{\alpha}(\cdot)$ is a known function.

As in the previous section, the control law consists of 'linear' and 'nonlinear' components of the form

$$\nu(t) = \nu_l(\rho) + \nu_n(t) \quad (75)$$

The 'linear' component is given by

$$\nu_l(\rho) = -\hat{A}_{21}(\rho)x_1(t) - (\hat{A}_{22}(\rho) - \Phi)s(t) \quad (76)$$

where the design matrix $\Phi \in \mathbb{R}^{4 \times 4}$ is chosen to be stable. The 'nonlinear' component is given by

$$\nu_n(t) = -\check{K}(t, x) \frac{P_2 s(t)}{\|P_2 s(t)\|} \quad \text{if } s(t) \neq 0 \quad (77)$$

where $P_2 \in \mathbb{R}^{4 \times 4}$ satisfies the Lyapunov equation (48).

The following proposition shows that the control law (75) will induce sliding in the presence of matched uncertainty.

Proposition 2: Suppose the combination of faults and failures satisfies the following condition

$$\begin{aligned} & (B_2(\rho)K(t)B_2^T(\rho))^T (B_2(\rho)K(t)B_2^T(\rho)) \\ & < (1 - \varepsilon)^2 (B_2(\rho)B_2^T(\rho))^T (B_2(\rho)B_2^T(\rho)) \end{aligned} \quad (78)$$

where ε is a positive scalar satisfying $1 > \varepsilon > \check{d}_2\tilde{\gamma}$, the scalar $\tilde{\gamma}$ is the bound from (74) and $\check{d}_2$ is a scalar defined as

$$\check{d}_2 := \max_{\rho \in \Theta, 0 < W(t) < I, \|\Delta\| < \bar{\Delta}} \check{D}_2(\rho, W) \quad (79)$$

Then if the matrix M has been chosen so that $\hat{A}_{11}(\rho) = A_{11}(\rho) - A_{12}(\rho)M$ is stable, a sliding motion is guaranteed to take place on $\mathcal{S}$ in finite time if the gain $\check{K}(t, x)$ in (77) is chosen as

$$\check{K}(t, x) \geq \frac{\check{d}_2(\tilde{\gamma}\|\nu_l(\rho)\| + \check{\alpha}(t, x)) + (1 - \varepsilon)\|\nu_l(t)\| + \check{\eta}}{(\varepsilon - \tilde{\gamma}\check{d}_2)} \quad (80)$$

where $\check{\eta}$ is a positive scalar.

Proof:

Substituting the control law in (75)-(77) into (73) yields

$$\begin{aligned} \dot{s}(t) &= \Phi s(t) - \check{K}(\cdot) \frac{P_2 s}{\|P_2 s\|} - (B_2(\rho)K(t)B_2^T(\rho)) \\ &\quad \times (B_2(\rho)B_2^T(\rho))^{-1} \nu(t) + \check{D}_2(\rho)\check{\xi}(t) \end{aligned} \quad (81)$$

For equation (81), consider a single quadratic Lyapunov function [33] given by $V(s) = s^T P_2 s$. Differentiating the Lyapunov function and using the Lyapunov equation defined in (48) gives

$$\begin{aligned} \dot{V} &= s^T (\Phi^T P_2 + P_2 \Phi) s - 2\check{K}\|P_2 s\| + 2s^T P_2 \check{D}_2(\rho)\check{\xi}(t) \\ &\quad - 2s^T P_2 (B_2(\rho)K(t)B_2^T(\rho)) (B_2(\rho)B_2^T(\rho))^{-1} \nu(t) \\ &= -\|s\|^2 - 2\check{K}\|P_2 s\| + 2s^T P_2 \check{D}_2(\rho)\check{\xi}(t) \\ &\quad - 2s^T P_2 (B_2(\rho)K(t)B_2^T(\rho)) (B_2(\rho)B_2^T(\rho))^{-1} \nu(t) \end{aligned} \quad (82)$$

It is easy to see from (78) that the following inequality holds

$$\|(B_2(\rho)K(t)B_2^T(\rho)) (B_2(\rho)B_2^T(\rho))^{-1}\| < (1 - \varepsilon) \quad (83)$$

If the condition in (78) is satisfied for all combinations of rotor faults and failures, using (83), it follows that

$$\begin{aligned} & -2s^T P_2 (B_2(\rho)K(t)B_2^T(\rho)) (B_2(\rho)B_2^T(\rho))^{-1} \nu(t) \\ & < 2\|P_2 s\| \|(B_2(\rho)K(t)B_2^T(\rho)) \\ & \quad \times (B_2(\rho)B_2^T(\rho))^{-1} (\|\nu_l(t)\| + \|\nu_n(t)\|)\| \\ & < 2\|P_2 s\|(1 - \varepsilon)(\|\nu_l(t)\| + \check{K}(t, x)) \end{aligned} \quad (84)$$

Using the Cauchy-Schwarz inequality the following inequality can be obtained

$$\|s^T P_2 \check{D}_2(\rho)\check{\xi}(t)\| < \|P_2 s\| \check{d}_2 \|\check{\xi}(t)\| \quad (85)$$

Therefore, using (83) and substituting (84)-(85) into (82), yields

$$\dot{V} \leq -\|s\|^2 - 2\|P_2 s\| (\varepsilon\check{K}(\cdot) - \check{d}_2\|\check{\xi}(t)\| - (1 - \varepsilon)\|\nu_l\|) \quad (86)$$

Equation (80) can be represented by

$$\check{K}(\cdot) (\varepsilon - \tilde{\gamma}\check{d}_2) \geq \check{d}_2 (\tilde{\gamma}\|\nu_l(\rho)\| + \check{\alpha}(t, x)) + (1 - \varepsilon)\|\nu_l\| + \check{\eta} \quad (87)$$

which upon rearrangement yields

$$\begin{aligned} \varepsilon\check{K}(\cdot) &\geq \check{d}_2 (\tilde{\gamma}\|\nu_l(\rho)\| + \check{K}(\cdot)\tilde{\gamma} + \check{\alpha}(\cdot)) + \check{\eta} + (1 - \varepsilon)\|\nu_l\| \\ &\geq \check{d}_2 (\tilde{\gamma}\|\nu(t)\| + \check{\alpha}(\cdot)) + \check{\eta} + (1 - \varepsilon)\|\nu_l\| \\ &\geq \check{d}_2 \|\check{\xi}(t)\| + \check{\eta} + (1 - \varepsilon)\|\nu_l\| \end{aligned} \quad (88)$$

Finally substituting equation (88) into (86) gives

$$\dot{V} \leq -\|s\|^2 - 2\check{\eta}\|P_2 s\| \quad (89)$$

Equation (89) indicates that an ideal sliding mode is achieved despite the presence of matched uncertainty. Note that equation (89) is the same as the one in (56) and therefore using similar arguments as in (57)-(59), it can be shown that the sliding surface will be reached in finite time. ■

Remark: The condition in (78) represents a smaller fault set than the one associated with (27), since any $K(t)$ satisfying (78) also satisfies (27). To see this, suppose $K_0(t)$ satisfies (78) and assume for a contradiction that $W_0(t) := I - K_0(t)$ does not satisfy (27) i.e. $\det(B_2(\rho)W_0(t)B_2^T(\rho)) = 0$. If $\det(B_2(\rho)W_0(t)B_2^T(\rho)) = 0$, then there exists a vector $\varsigma \neq 0$ such that $B_2(\rho)W_0(t)B_2^T(\rho)\varsigma = 0$ or equivalently $B_2(\rho)B_2^T(\rho)\varsigma = B_2(\rho)K_0(t)B_2^T(\rho)\varsigma$ from the definition of W_0 . Consequently

$$\begin{aligned} & \varsigma^T (B_2(\rho)K_0(t)B_2^T(\rho))^T (B_2(\rho)K_0(t)B_2^T(\rho)) \varsigma \\ &= \varsigma^T (B_2(\rho)B_2^T(\rho))^T (B_2(\rho)B_2^T(\rho)) \varsigma \end{aligned}$$

Consequently multiplying (78) on the left by ς^T and on the right by ς yields $1 < (1 - \varepsilon)^2$ which is a contradiction since $0 < \varepsilon < 1$. As $\varepsilon \rightarrow 0$, the discrepancy between (27) and (78) becomes smaller, but the constraint in (78) will always be more conservative. This conservatism must be traded-off against the benefit of greater simplicity and the fact that an FDI scheme is not required.

C. Final Control Law

From (29), (67)-(68) and (75)-(77) the final control signal $u(t)$ actually sent to all the eight rotors is given by

$$u(t) = B_2^T(\rho) \left(B_2(\rho) B_2^T(\rho) \right)^{-1} \times \left(-\hat{A}_{21}(\rho)x_1(t) - (\hat{A}_{22}(\rho) - \Phi)s(t) - \tilde{K}(\cdot) \frac{P_2 s(t)}{\|P_2 s(t)\|} \right) \quad (90)$$

where $s(t)$ is defined in (40). Note that due to the ‘fixed’ control allocation structure chosen in (68), the control signal (90) does not contain any information about the rotor effectiveness level $W(t)$ as compared to the online control allocation in (63). Therefore, the scheme proposed in this section is simpler since no FDI is required, albeit at the expense of a slightly more restrictive set of rotor faults/failures which satisfy condition (78).

V. ONLINE VS FIXED CONTROL ALLOCATION

The synthesis of a baseline controller for the case of online control allocation given in Section III is a direct extension of a classical sliding mode controller applied to an LPV system. As discussed in Section III-A, due to the control allocation formulation, the rotor faults/failures in the octorotor represent ‘matched uncertainty’ [29], [27] which sliding mode control is inherently robust against. Although theoretically it is a direct extension of classical sliding mode theory, the scheme requires an FDI unit to continuously monitor the effectiveness levels of all the eight rotors individually. This can be quite computationally expensive for the onboard microcontroller (depending on the complexity of the FDI scheme used).

This is the advantage of using the fixed control allocation based scheme which sends the control signals to all rotors regardless of any faults/failures. The scheme does not require any FDI information and the rotor effectiveness level is not required. As such the fixed control allocation scheme in Section IV would be considered a passive approach [35], [36] (relying on the robustness of controller). Whereas the online control allocation scheme in Section III would be termed an active approach [35], [36] since it relies on information from the FDI scheme. The controller synthesis however is not as direct as in the case of online CA and uses a more conservative analysis and requires a larger modulation gain in the event of faults/failures.

Note that despite the differences between the two schemes, the baseline controller for online and the fixed control allocation are exactly the same. The difference can be seen in the final control law given in (63) and (90) due to the different control allocation matrix used. In the simulation results which follow, the same baseline control will be used.

VI. DESIGNS

Symbol	Value	Unit
m	1.3	kg
I_{xx}, I_{yy}, I_{zz}	0.0150, 0.0150, 0.0026	kgm^2
b, d	$3.13 \times 10^{-5}, 7.5 \times 10^{-7}$	Ns^2, Nms^2
J_r	6.0×10^{-5}	Kgm^2
L	0.23	m

TABLE II
OCTOROTOR PARAMETERS

Table II describes the parameters of the octorotor which are used for simulation in this paper. Rotor speed at steady hover can be obtained using (10) where

$$\ddot{z}_b = g - b_z c_1 \tau_1(t) = g - b_z c_1 8b\Omega_1^2$$

Using the fact that at steady hover $p = q = r = \ddot{z}_b = \dot{\phi} = \theta = 0$ and assuming all eight rotor speeds are the same, then from (15), $\tau_1(t) = 8b\Omega_1^2$, and solving for one of the rotors yields

$$\Omega_i(0) = \sqrt{mg/8b} = 448.2187 \text{ (rpm)}$$

as the speed for each rotor during hover. This translates to $\tau(0) = [16.072 \ 0 \ 0 \ 0]^T$, i.e. the total thrust required for hover is 16.072 while the resultant roll, pitch and yaw torques from the rotors are zero.

The control objective is to track altitude, roll, pitch and yaw commands given by

$$y_c(t) = [z_{bcmd}(t) \ \phi_{cmd}(t) \ \theta_{cmd}(t) \ \psi_{cmd}(t)]^T$$

To achieve this objective, integral action [27], [29] is employed by augmenting new states given by

$$\dot{x}_c(t) = y_c(t) - C_c x(t) \quad (91)$$

to yield $x_a(t) = \text{col}(x_c(t), x(t))$. Here $C_c = [I_4 \ 0_{4 \times 4}]$ is the distribution matrix for the controlled outputs. It is assumed that the command signal $y_c(t)$ is differentiable and satisfies

$$\dot{y}_c(t) = \Gamma(y_c(t) - Y_c) \quad (92)$$

where Y_c is a constant demand vector and $\Gamma \in \mathbb{R}^{4 \times 4}$ is a stable design matrix [29]. For design purposes, the augmented system for developing the control law (ignoring the uncertainty associated with Δ and $\zeta(t)$ in Equation (23)) is given by

$$\dot{x}_a(t) = A_a(\rho)x_a(t) + B_{\nu,a}u(t) + B_c y_c(t) \quad (93)$$

where

$$A_a(\rho) = \begin{bmatrix} 0 & -C_c \\ 0 & A(\rho) \end{bmatrix}, B_{\nu,a} = \begin{bmatrix} 0 \\ B_\nu \end{bmatrix}, B_c = \begin{bmatrix} I_4 \\ 0 \end{bmatrix} \quad (94)$$

Note that the system in (93) represents the nominal fault-free system on which the baseline controller will be based. Finally the switching function associated with the augmented system is given by

$$s_a(t) = S_a x_a(t) = [M_a \ I_4] x_a(t) \quad (95)$$

where $s_a : \mathbb{R}^{12} \rightarrow \mathbb{R}^4$ and $M_a \in \mathbb{R}^{4 \times 8}$.

A. Baseline Controller

As shown in equations (45)-(47) and (75)-(77), the proposed baseline ‘virtual control’ law for the augmented system in (93) is given by

$$\nu_a(\rho) = \nu_{a,l}(\rho) + \nu_{a,n}(t) \quad (96)$$

i.e. it comprises two components: the ‘linear’ component $\nu_{a,l}(\rho)$ and the ‘nonlinear’ component $\nu_{a,n}(t)$. Due to the presence of the reference signal $y_c(t)$, the ‘linear’ component of the control law for the augmented system is given by

$$\nu_{a,l}(\rho) = L_a(\rho)x_a(t) + L_c y_c(t) \quad (97)$$

where $L_c y_c(t)$ represents a feed-forward term. Here

$$L_a(\rho) = -(S_a A_a(\rho) - \Phi_a S_a) \text{ and } L_c = -M_a B_c \quad (98)$$

Note that due to the ordering of the states given in (1), the augmented system (94) is already in regular form [27], [29]. The matrix S_a defines the switching function matrix in (95) while Φ_a is the stable design matrix which is analogous to Φ in (48). The ‘nonlinear’ component of the baseline controller for the augmented system (94) is given by

$$\nu_{a,n}(t) = -\mathcal{K}(t, x_a) \frac{s_a(t)}{\|s_a(t)\|} \text{ for } s_a(t) \neq 0 \quad (99)$$

The final control laws for the augmented system (94) are:

$$u_o(t) = W B_2^T(\rho) \left(B_2(\rho) W^2(t) B_2^T(\rho) \right)^{-1} \nu_a(\rho) \text{ for online CA} \quad (100)$$

$$u_f(t) = B_2^T(\rho) \left(B_2(\rho) B_2^T(\rho) \right)^{-1} \nu_a(\rho) \text{ for fixed CA} \quad (101)$$

In this paper, the design parameter Φ_a (which can be used to determine how fast sliding is attained [27], [29]) has been chosen as $\Phi_a = \text{diag}(-10, -20, -50, -1)$. Note the poles of the sliding motion associated with Φ_a have been chosen faster than the ones associated with the reduced order sliding motion. Here roll and pitch channel (the second and third components of Φ_a) all weighted heavily compared to altitude and yaw (the first and fourth components of Φ_a) to reflect the importance of roll and pitch control in this particular design. Meanwhile the pre-filter matrix from (92) has been chosen as $\Gamma = \text{diag}(-2, -20, -100, -1)$. Again note that roll and pitch channel (the second and third components of Γ) have higher bandwidth compared to altitude and yaw (the first and fourth components of Γ) to reflect the requirement of needing faster responses in the roll and pitch channels (which serve as manipulated variables for the outer loop x_b, y_b position loop).

As shown in Section III-B, the controller gains for both the online and fixed CA schemes are dependant on the design matrix M . For the octorotor system presented in this paper, since $A_{11}(\rho) = 0$ and $A_{12}(\rho) = I_4$ (i.e. fixed), simple pole placement by choosing M as a Hurwitz matrix has been used. For the augmented system in (93), the closed-loop poles have been chosen as $\{-8.8286 \pm 3.5157i, -8.8286 \pm 3.5157i, -8.8286 \pm 3.5157i, -8.8286 \pm 3.5157i\}$ which yields $M_a = \begin{bmatrix} -90.3048 I_4 & 17.6573 I_4 \end{bmatrix}$.

In the simulations and for practical applications, a simple sigmoidal approximation [29] $((P_2 s_a)/(\|P_2 s_a\| + \delta_a))$ has been used to smooth the unit vector function in (99) where δ_a is a small positive scalar. This approximation reduces the chattering effect while also providing some degree of tuning – especially when dealing with rotor faults/failures. The results shown in this paper have been obtained using $\delta_a = 0.1$. Finally, the nonlinear modulation gain $\mathcal{K}$ in (99) has been chosen as $\mathcal{K} = 20$ for both the online and fixed CA schemes. Both gains are chosen to be identical in order to compare the performance between the two schemes.

B. LPV vs. nonlinear model

Figure 2 shows a comparison between the LPV model used for design and the actual nonlinear model used for simulation. Note that there is no uncertainty present in either the LPV or the nonlinear plant. The same baseline controller has been

used on both the LPV and nonlinear plant to provide a tracking command on altitude, roll, pitch and yaw angles. Figure 2 shows no visible difference between the LPV and the nonlinear tracking performance. This shows that the LPV model discussed in Section II-B, which is used for designing the baseline controller, captures the dynamics of the formal nonlinear model discussed in Section II-A.

Remark: Note that the results in Figure 2(a) do not include the x_b and y_b positions since the LPV plant used for design does not contain these two states.

VII. SIMULATION RESULTS

The simulation results presented in the following subsections were obtained using the nonlinear model in (10) using SIMULINK with a fixed time solver ODE14x with a step size of 0.01s.

To provide a position tracking capability in the x and y direction, an outer loop PID structure has been considered. Figure 3 shows the overall configuration of the outer loop PID and the baseline sliding mode scheme with control allocation. The output of the PID is the roll and pitch command signal, while for simplicity the yaw angle command is set to zero. In the simulations that follow, the PID gains for both x and y position control are set as $k_p = 1, k_i = 0.05, k_d = 7$.

During the simulation, motor rotational speed is limited between $[\Omega_{min} \ \Omega_{max}] = [0 \ 1500]$ (rpm). In the simulation, sensor noise is also included. This is modeled as Gaussian noise with variance 3×10^{-8} for all the sensors. This is similar to the noise profile used in [37]. Finally, for the simulation results that follow, the uncertainty level in the thrust and drag factor (Δ_b, Δ_d) are set to 10% of their actual values (b, d) (given in Table II).

Note that for the online CA approach, it is assumed that perfect FDI information is available to provide the rotor effectiveness levels $w_1(t) \dots w_8(t)$ for the matrix $W(t)$. In the simulation, it is assumed that the actual rotor speed is measurable and can be compared directly with the commanded speed to provide the effectiveness level. Alternatively any FDI scheme in the literature can be used to provide this information (see for example the scheme proposed in [27] which has the capability to reconstruct unknown faulty actuators).

Remark: Note that the overall structure of the online CA scheme is shown in Figure 3. In this figure, the FDI unit is required as the scheme requires knowledge of the rotor effectiveness levels. For the case of fixed CA, it will be assumed that this FDI unit is not present.

A. Fault Free

Figures 4-5 show the results for the fault free case. Figure 4 shows the overall trajectory of the octorotor. This manoeuvre will be repeated for the fault/failure cases in order to compare it with the fault free performance. The trajectory consists of a change of direction in the x and y position as well as a change in altitude at the same time (which can be seen in Figure 5(a)). Figure 5(a) also shows good roll, pitch and yaw tracking performance during fault free conditions. Figure 5(a) also shows the switching function (represented in terms of its norm) which demonstrates that despite the presence of 10% uncertainty in the thrust and drag factors, sliding still occurs. Note that for the fault free case, $W = I$ (as shown in Figure

Fig. 2. LPV vs non linear model: states and rotor speed

5(c)) and therefore equation (63) is the same as the one in equation (90), i.e. there is no difference between online and fixed CA during the fault free case.

B. Online CA

Figures 6-8 show the results for the online CA scheme when faults/failures have occurred. Figure 6 shows the results when rotors 1,4,6 and 7 fail at 20sec while Figure 7 shows a more challenging condition when rotors 1,4,5,8 are faulty (i.e. 70% efficient) from 20sec, and rotors 2 and 3 totally fail at 25sec. Figure 8 shows the same challenging condition as in Figure 7, but with imperfection and delays in the FDI.

1) *Rotor 1,4,6,7 failure - online CA*: Figure 6(b) shows rotors 1,4,6 and 7 failing at 20sec as the rotor speed drops to zero. At the same time the speed of rotors 2,3,5,8 all increase compared to the fault free case (in red), after the control signals have been redistributed, in order to maintain nominal performance. Figure 6(c) clearly shows the drop in rotor effectiveness level to zero at 20sec for rotors 1,4,6 and 7 which indicates total failure. Figure 6(a) shows that despite the presence of uncertainty as well as four rotors failing at 20sec, there is no visible degradation in tracking performance between the fault free and failure scenarios (as

the lines overlap). Finally Figure 6(a) shows the effect of four rotors failing, on the switching function. There is a slight deviation from the sliding surface as compared to the fault free condition, but it is relatively small.

2) *Rotor 1,4,5,8 fault and rotor 2,3 failure - online CA*: Figure 7 shows a challenging condition when a total of six rotors contain some faults/failures in the presence of uncertainty. Figure 7(b) shows the comparison between the fault free case and when faults/failures occur. Figure 7(b) shows that faults occur to rotors 1,4,5, and 8 at 20sec followed by rotors 2 and 3 failure at 25sec. Despite a combination of faults and failures to six of the rotors as well as the uncertainty, there is no significant change in the overall tracking performance (Figure 7(a)). There is a small transient at 25sec in the pitch angle when rotors 2 and 3 fail, but the effect on altitude and x, y position tracking is negligible. The other difference can be seen in the switching function when during fault/failure conditions, the switching function is slightly further away from zero compared to the fault free case. Note that Figure 7(c) shows perfect estimation of rotor effectiveness levels from an FDI.

3) *Rotor 1,4,5,8 fault and rotor 2,3 failure - online CA with imperfect FDI information*: Figure 8 shows the same condition

Fig. 3. Online Control Allocation Architecture

Fig. 4. Fault free: 3d manoeuvre

as the one in Figure 7, except here there is imperfection and delays in the FDI. Figure 8(c) shows imperfect estimation of all rotor effectiveness level from the start of the simulation by 10%. There is also a detection delay of 0.5sec when faults and failures occur at 20sec and 25sec. Figure 7(a) shows that despite a combination of faults and failures to six of the rotors, as well as uncertainty in the system and imperfection and delays in the FDI, there is no significant effect on the overall tracking performance as compared to the fault free condition. As in Figure 7(a), there are small transients when rotors 2 and 3 fail at 25sec in the roll, pitch and yaw angles, but the effect on altitude and x, y position tracking is small. Finally Figure 8(a) also shows that the switching function is slightly further away from zero compared to the fault free case.

C. Fixed CA

The same tests have been repeated for the fixed CA scheme. Figure 9 shows the results when rotors 1,4,6 and 7 fail, while Figure 10 shows the results when rotors 1,4,5,8 are only 70% efficient and rotors 2 and 3 totally fail.

1) *Rotor 1,4,6,7 failure - fixed CA:* Figure 9(b) shows that the rotational speed for rotors 1,4,6 and 7 becomes zero at

20sec when the failures occur. At the same time the speed on 2,3,5 and 8 increase to compensate for the failed rotors. Figure 9(c) shows the actual effectiveness levels of each rotor. As in the previous tests, Figure 9(a) shows no visible degradation in terms of tracking performance as compared to the fault free case. The plot of $\|s(t)\|$ shows the switching function as compared to the fault free case. As in previous tests, there is a slight increase in the switching function when the failures occur. The switching function for the fixed CA in Figure 9(a) is slightly further away from zero as compared to the online CA in Figure 6(a) although in both cases the same nonlinear modulation gain $K = 20$ is used. This is expected because as for the case of fixed CA, a larger modulation is required to obtain the same level of switching function activity. This is the down-side of using the simpler fixed CA scheme without the knowledge of rotor effectiveness level.

2) *Rotor 1,4,5,8 fault and rotor 2,3 failure - fixed CA:* Figure 10(b) shows the results when faults occur to rotors 1,4,5 and 8 at 20sec and failures to rotors 2 and 3 at 25 sec using fixed CA. Again, as in the previous tests, Figure 10(a) shows a very small change in the overall tracking performance compared to the fault free case (the lines overlap). The plot

Fig. 5. Fault free

of $\|s(t)\|$ shows a slightly larger deviation from the sliding surface as compared to the online CA results shown in Figure 7(a). Again this is expected as a larger gain is required to maintain the same level of switching when online CA is used. Also note that as in the previous test (Figure 7(a)), there is a small transient at 25sec in the roll and pitch angle when rotors 2 and 3 fail as seen in Figure 10(a), but the effect on altitude and x, y position tracking are small.

VIII. CONCLUSIONS

This paper has demonstrated new LPV sliding mode fault tolerant control schemes with control allocation capability, applied to an octorotor UAV in the presence of uncertainty. Two sliding mode control allocation FTC schemes have been presented: an online scheme and a fixed control allocation scheme. The first scheme corresponds to an online control allocation framework where the rotor effectiveness level (from an FDI unit) is explicitly used as part of the control allocation scheme. In this scheme, the control signals are redistributed

to the remaining healthy rotors. The second scheme assumes that the rotor effectiveness level information is not available and therefore uses a fixed control allocation structure. From an implementation point of view, this is simpler as there is no FDI scheme required. However this is achieved at the expense of a larger nonlinear modulation gain. Simulation results from the two schemes show good state tracking performance in the presence of uncertainties in the drag and thrust factors, sensor noise, as well as simultaneous faults/failures to multiple rotors. The results highlight the capability of the proposed schemes by taking full advantage of the available redundant rotors in the octorotor configuration.

REFERENCES

- [1] Y. Zhang and A. Chamseddine, "Fault tolerant flight control techniques with application to a quadrotor UAV testbed," in *Automatic Flight Control Systems - Latest Developments*, T. Lombaerts, Ed. InTech, 2012, pp. 119–150.
- [2] F. Sharifi, M. Mirzaei, B. Gordon, and Y. Zhang, "Fault tolerant control of a quadrotor UAV using sliding mode control," in *Conference on Control and Fault Tolerant Systems*, Nice, France, 2010.

Fig. 6. Rotor 1,4,6,7 failure - online CA

- [3] H. A. Izadi, Y. Zhang, and B. W. Gordon, "Fault tolerant model predictive control of quad-rotor helicopters with actuator fault estimation," in *Proceedings of the 18th IFAC World Congress, Milan*, 2011.
- [4] I. Sadeghzadeh, A. Mehta, Y. Zhang, and C. Rabbath, "Fault-tolerant trajectory tracking control of a quadrotor helicopter using gain-scheduled PID and model reference adaptive control," in *Conference of the Prognostics and Health Management Society*, 2011.
- [5] A. Chamseddine, D. Theilliol, Y. Zhang, C. Join, and C. Rabbath, "Active fault-tolerant control system design with trajectory re-planning against actuator faults and saturation: Application to a quadrotor unmanned aerial vehicle," *International Journal of Adaptive Control and Signal Processing*, vol. 2013, no. DOI: 10.1002/acs.2451, 2013.
- [6] L. Besnarda, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," *Journal of the Franklin Institute*, vol. 349, pp. 658–684, 2012.
- [7] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *International Conference on Robotics and Automation, Barcelona, Spain*, 2005.
- [8] T. Madani and A. Benallegue, "Control of a quadrotor mini-helicopter via full state backstepping technique," in *45th IEEE Conference on Decision and Control, San Diego, CA, USA*, 2006.
- [9] —, "Backstepping sliding mode control applied to a miniature quadrotor flying robot," in *32nd IEEE Conference on Industrial Electronics, IECON*, 2006.
- [10] —, "Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles," in *American Control Conference, New York City, USA*, 2007.
- [11] R. Xu and U. Ozguner, "Sliding mode control of a quadrotor helicopter," in *45th IEEE Conference on Decision and Control, San Diego, CA, USA*, 2006.
- [12] A. Freddi, A. Lanzon, and A. Longhi, "A feedback linearization approach to fault tolerance in quadrotor vehicles," in *Proceedings of the 18th IFAC World Congress, Milan*, 2011.
- [13] A. Sanca, P. Alsina, and J. Cerqueira, "Dynamic modeling with nonlinear inputs and backstepping control for a hexarotor micro-aerial vehicle," in *Latin American Robotics Symposium and Intelligent Robotics Meeting*, 2010.
- [14] L. Yin, J. Shi, and Y. Huang, "Modeling and control for a six-rotor aerial vehicle," in *International Conference on Electrical and Control Engineering*, 2010.
- [15] B. R. Trilaksono, R. Triadhitama, W. Adiprawita, A. Wibowo, and A. Sreenatha, "Hardware-in-the-loop simulation for visual target tracking of octorotor UAV," *Aircraft Engineering and Aerospace Technology: An International Journal*, vol. 83, no. 6, pp. 407–419, 2011.
- [16] A. Budiyo, W. Adiprawita, and B. Riyanto, "Modeling and control of unmanned flying robots," in *2011 International Conference on Electrical Engineering and Informatics, Bandung, Indonesia*, 2011.
- [17] V. Adir and A. Stoica, "Integral LQR control of a star-shaped octorotor," *INCAS Bulletin*, vol. 4, no. 2, pp. 3–18, 2012.
- [18] A. Marks, J. Whidborne, and I. Yamamoto, "Control allocation for

Fig. 7. Rotor 1,4,5,8 fault and rotor 2,3 failure - online CA

- fault tolerant control of a VTOL octocopter,” in *UKACC International Conference on Control, Cardiff, UK*, 2012, pp. 357–362.
- [19] V. Adir, A. Stoica, A. Marks, and J. Whidborne, “Modelling, stabilization and single motor failure recovery of a 4Y octocopter,” in *13th IASTED International Conference on Intelligent Systems and Control (ISC 2011)*, Cambridge, U.K., 2011.
- [20] Anon. (2013) Amazon testing drones for deliveries. [Online]. Available: <http://www.bbc.co.uk/news/technology-25180906>
- [21] —. (2013) Amazon prime air. [Online]. Available: <http://www.amazon.com/b?node=8037720011>
- [22] G. J. Balas, “Linear, parameter-varying control and its application to a turbofan engine,” *International Journal of Robust and Nonlinear Control*, vol. 12, pp. 763–796, 2002.
- [23] C. Balas, “Modelling and linear control of a quadrotor,” School of Engineering, Cranfield, Master of Science Thesis, 2007.
- [24] S. Bouabdallah, A. Noth, and R. Siegwart, “PID vs LQ control techniques applied to an indoor micro quadrotor,” in *IEEE/RSJ International Conference On Intelligent Robots and Systems, Sendai, Japan*, 2004.
- [25] S. Rangajeeva and J. Whidborne, “Linear parameter varying control of a quadrotor,” in *6th International Conference on Industrial and Information Systems, ICIS*, 2011.
- [26] A. Budiyo and H. Sutarto, “Linear parameter varying model identification for control of rotorcraft-based UAV,” in *5th Taiwan-Indonesia Workshop on Aeronautical Science, Technology and Industry, Tainan, Taiwan*, 2006.
- [27] H. Alwi, C. Edwards, and C. P. Tan, *Fault Detection and Fault-Tolerant Control Using Sliding Modes*, ser. Advances in Industrial Control. Springer-Verlag, 2011.
- [28] S. Bouabdallah, “Design and control of quadrotors with application to autonomous flying,” Ph.D. dissertation, École Polytechnique Fédérale De Lausanne, 2007.
- [29] C. Edwards and S. K. Spurgeon, *Sliding Mode Control: Theory and Applications*. Taylor & Francis, 1998.
- [30] H. Alwi and C. Edwards, “Fault tolerant control using sliding modes with on-line control allocation,” *Automatica*, vol. 44, no. 7, pp. 1859–1866, 2008.
- [31] O. Härkegård and S. T. Glad, “Resolving actuator redundancy - optimal control vs. control allocation,” *Automatica*, vol. 41, no. 1, pp. 137–144, 2005.
- [32] P. Frank, E. A. Garcia, and B. Kppen-Seliger, “Modelling for fault detection and isolation versus modelling for control,” *Mathematical and Computer Modelling of Dynamical Systems*, vol. 7, no. 1, pp. 1–46, 2001.
- [33] G. Becker, A. Packard, D. Philbrick, and G. Balas, “Control of parametrically-dependent linear systems a single quadratic lyapunov approach,” in *American Control Conference*, 1993, pp. 2795–99.
- [34] P. Gahinet, A. Nemirovski, A. Laub, and M. Chilali, *LMI Control Toolbox, User Guide*. MathWorks, Inc., 1995.
- [35] R. Patton, “Robustness in model-based fault diagnosis: the 1997 situation,” *IFAC Annual Reviews*, vol. 21, pp. 101–121, 1997.
- [36] Y. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-

Fig. 8. Rotor 1,4,5,8 fault and rotor 2,3 failure - online CA, with imperfect and delayed FDI

tolerant control systems,” *Annual Reviews in Control*, vol. 32, pp. 229–252, 2008.

- [37] H. Smaili, J. Breeman, T. Lombaerts, and D. Joosten, “Recover: A benchmark for integrated fault tolerant flight control evaluation,” in *Fault Tolerant Flight Control*, ser. Lecture Notes in Control and Information Sciences, C. Edwards, T. Lombaerts, and H. Smaili, Eds. Springer Berlin / Heidelberg, 2010, vol. 399, pp. 171–221.

Fig. 9. Rotor 1,4,6,7 failure - fixed CA

Fig. 10. Rotor 1,4,5,8 fault and rotor 2,3 failure - fixed CA