Proceedings of the EuroGNC 2013, 2nd CEAS Specialist Conference
on Guidance, Navigation & Control, Delft University of Technology,
Delft, The Netherlands, April 10-12, 2013

Fault Tolerant Control of Octorotor Using Sliding Mode Control Allocation

Halim Alwi and Christopher Edwards

Abstract This paper presents a fault tolerant control scheme using sliding mode control allocation for an octorotor UAV. Compared to the existing literature on quadrotor or octorotor UAVs, the scheme in this paper takes full advantage of the redundant rotors to handle more than one rotor failure. A sliding mode approach is used as the core baseline controller, which is robust against uncertainty in the input channels – including faults to any of the rotors. Even when total failures occur, no reconfiguration is required to the baseline controller, and the control signals are simply re-allocated to the remaining healthy rotors using control allocation, to maintain nominal fault-free performance. To highlight the efficacy of the scheme, various types of rotor fault/failure scenarios have been tested on a nonlinear model. The results show no visible change in performance when compared to the fault-free case.

1 Introduction

There has been much research interest focussed on UAVs in recent years (see for example [6, 15, 5, 19]). The quadrotor UAV is one of the most popular choices due to its cheap cost, its simplicity of build and the availability of open source codes for the controller. Whilst most of the fault tolerant control (FTC) schemes for aerospace applications presented in the literature[1] cannot be tested easily due to cost and safety factors, for quadrotors this is possible in controlled laboratory environments. The

Halim Alwi
Control Systems Research, Dept. of Engineering, University of Leicester, LE1 7RH, UK e-mail: ha18@le.ac.uk

Christopher Edwards
Centre for Dynamics and Control, University of Exeter, EX4 4QF, UK e-mail: C.Edwards@exeter.ac.uk

[1] See [9] for a recent overview of the work in the field of FTC applied to aerospace systems, and detailed descriptions of certain state-of-the-art methods applied to a civil aircraft benchmark problem.

work described in [28] (and the references therein) catalogues leading work applying FTC schemes to quadrotors (see also [23, 14, 20]). Many different paradigms have been proposed including PID [5, 7, 20, 28], LQR [1, 5, 7], sliding mode control [18, 23, 26, 28], model predictive control (MPC) [14, 28], model reference adaptive control (MRAC) [28, 20], nonlinear dynamic inversion (NDI) [11, 15] and backstepping [7, 17, 16, 18, 28]. However because of the lack of redundancy in a quadrotor (due to its 4 rotor configuration), which is a critical factor for FTC design, almost all quadrotor FTC schemes in the literature only deal with partial faults on the rotors (or motors). A notable exception is recent work in [11] which allows one of the rotors to fail at the expense of yaw control, in order to maintain roll and pitch control.

Due to the lack of redundancy in quadrotors, it is natural to consider multirotor UAVs such as the hexrotor [21, 27] and octorotor [24, 2] for testing FTC schemes. However, despite the available redundancy, there has been almost no work for FTC on hexrotor or octorotor UAVs with the exception of [2, 22] (although [2] only allows for one of the rotors to fail). Despite discussing control allocation (CA) for octorotors (which offers the possibility of re-routing control signals among all 8 rotors), the work in [8] did not explicitly propose any FTC schemes.

Despite the robustness properties of sliding mode control (SMC), which can inherently deal with actuator faults, most of the SMC work in the area of multi rotor UAVs only focusses on quadrotors (see for example [7, 17, 16, 18, 26]), and only deals with partial faults due to lack of redundancy. The scheme proposed in this paper considers an octorotor and takes full advantage of the available redundant rotors (double redundancy for octorotor) and uses the fault tolerant sliding mode control allocation scheme originally developed in [4] to deal with faults and failures for generic over-actuated systems. Simulation results on a nonlinear model with various types of rotor fault/failure scenarios will be used to highlight the potential of the proposed scheme.

2 Octorotor

2.1 Equations of Motion

As in [6, 1], several simplifying assumptions will be introduced to create a (linear) model which can conveniently be used for control law design. Here it is assumed that:

- drag and thrust coefficients are assumed to be constant;
- the hub forces and rolling moments are neglected;
- the inertia matrix off-diagonal terms are zero (the octorotor is symmetric).

Remark The first assumption is a stringent one, but it facilitates the creation of a simplified linear model which can be used as the basis for the control law design. However it is possible to go to the next level of complication and allow the drag and thrust components to depend on measured parameters (such as speed) to create a LPV representation. In this situation a scheme such as the one recently proposed in [12] could be employed to create a gain scheduled version of the scheme which will be described in this paper.

Based on the above assumptions, the nonlinear equations of motion for an octorotor are the same as those for the quadrotor in [6], and can be written as:

$$\dot{X}(t) = \frac{d}{dt}\begin{bmatrix} x_b \\ y_b \\ z_b \\ \phi \\ \theta \\ \psi \\ \dot{x}_b \\ \dot{y}_b \\ \dot{z}_b \\ p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \\ \dot{z}_b \\ p + qsin(\phi)tan(\theta) + rcos(\phi)tan(\theta) \\ qcos(\phi) - rsin(\theta) \\ qsin(\phi)sec(\theta) + rcos(\phi)sec(\theta) \\ b_x \frac{1}{m}\tau_1(t) \\ b_y \frac{1}{m}\tau_1(t) \\ g - b_z \frac{1}{m}\tau_1(t) \\ \frac{I_{yy}-I_{zz}}{I_{xx}} qr + \frac{J_r}{I_{xx}} q\Omega_r + \frac{l}{I_{xx}}\tau_2(t) \\ \frac{I_{zz}-I_{xx}}{I_{yy}} pr - \frac{J_r}{I_{yy}} p\Omega_r + \frac{l}{I_{yy}}\tau_3(t) \\ \frac{I_{xx}-I_{yy}}{I_{zz}} qp + \frac{1}{I_{zz}}\tau_4(t) \end{bmatrix} \tag{1}$$

where the quantities $b_z = cos(\phi)cos(\theta)$, $b_y = cos(\phi)sin(\theta)cos(\psi) - sin(\phi)cos(\psi)$ and $b_x = cos(\phi)sin(\theta)cos(\psi) + sin(\phi)sin(\psi)$. The parameter m represents mass, I_{xx}, I_{yy}, I_{zz} are the inertia coefficients on the x,y,z axis, J_r is the rotor inertia, L is the arm length and l is the moment arm length (i.e. $l = L\cos(\pi/8)$ for the octorotor [1]). The states are

$$X = \begin{bmatrix} x_b \; y_b \; z_b \; \phi \; \theta \; \psi \; \dot{x}_b \; \dot{y}_b \; \dot{z}_b \; p \; q \; r \end{bmatrix}^{\mathrm{T}} \tag{2}$$

which represent position in the body x, y, z axis; roll angle, pitch angle, yaw angle; velocity in the body x, y, z axis; roll rate, pitch rate and yaw rate. The inputs are

$$\tau(t) = \begin{bmatrix} \tau_1(t) \; \tau_2(t) \; \tau_3(t) \; \tau_4(t) \end{bmatrix}^{\mathrm{T}} \tag{3}$$

which represent the total thrust, roll torque, pitch torque and yaw torque respectively on the vehicle. The variable Ω_r is the overall residual propeller speed from unbalanced rotor rotation and is given by

$$\Omega_r = -\Omega_1 - \Omega_2 + \Omega_3 + \Omega_4 - \Omega_5 - \Omega_6 + \Omega_7 + \Omega_8 \tag{4}$$

where $\Omega_1, \ldots \Omega_8$ are the individual propeller angular rates. The plant input torque and the forces $\tau_1(t), \ldots \tau_4(t)$ are mapped from the individual contributions of each of the eight rotors [1, 24] and are given by

$$\underbrace{\begin{bmatrix} \tau_1(t) \\ \tau_2(t) \\ \tau_3(t) \\ \tau_4(t) \end{bmatrix}}_{\tau(t)} = \underbrace{\begin{bmatrix} b & b & b & b & b & b & b & b \\ 0 & 0 & -bl & -bl & 0 & 0 & bl & bl \\ bl & bl & 0 & 0 & -bl & -bl & 0 & 0 \\ -d & -d & d & d & -d & -d & d & d \end{bmatrix}}_{B_\Omega} \underbrace{\begin{bmatrix} \Omega_1^2(t) \\ \vdots \\ \Omega_8^2(t) \end{bmatrix}}_{u(t)} \tag{5}$$

In (5) b and d are the thrust factor and drag factor respectively, which are assumed to be fixed [6]. Note that in terms of the equation of motion, (5) is the only source of difference when compared to the typical quadrotor.

2.2 Linearization

A linearization has been obtained at steady hover at an altitude of 10m. During hover $p = q = r = \phi = \theta = \psi = 0$. For design, only the following states are considered

$$x(t) = \begin{bmatrix} z_b \ \phi \ \theta \ \psi \ \dot{z}_b \ p \ q \ r \end{bmatrix}^{\mathrm{T}} \tag{6}$$

The linear model matrices given by [6] are:

$$\dot{x}(t) = Ax(t) + B_\tau \tau(t) + D\Omega_r(t) \tag{7}$$

where

$$A = \begin{bmatrix} 0 & I_4 \\ 0 & 0 \end{bmatrix}, \ B_\tau = \begin{bmatrix} 0 \\ B_{\tau,2} \end{bmatrix}, \ D = \begin{bmatrix} 0 \\ D_2 \end{bmatrix} = 0 \tag{8}$$

and $B_{\tau,2} = \mathrm{diag}(-\frac{1}{m}, \frac{1}{I_{xx}}, \frac{1}{I_{yy}}, \frac{1}{I_{zz}})$, $D_2 = [0 \ \frac{J_r}{I_{xx}} p \ \frac{J_r}{I_{yy}} q \ 0]^{\mathrm{T}}$. Note that the last term in (7) is considered as a disturbance term and is not considered during the controller synthesis. (Furthermore based on a hover condition about which the controller is designed, $D = 0$ since $p = q = 0$.) Also note that B_τ and D in (8) have a special structure which will be exploited during the controller design – especially in the control allocation scheme. The linear model which is used for synthesis is given by

$$\dot{x}(t) = Ax(t) + B_\tau \tau(t) \tag{9}$$

From (5), the torques and forces $\tau(t) = B_\Omega u(t)$, and therefore, the overall linear model can be written in terms of the contribution of each rotor as

$$\dot{x}(t) = Ax(t) + \underbrace{B_\tau B_\Omega}_{B} u(t) \tag{10}$$

The linear model given in (10) will be used for the controller design and the stability analysis. This representation is used in order to analyze the effect of rotor faults and failures on the performance and stability of the system, as well as to present the control allocation scheme used for FTC. Due to the structure of B_τ and B_Ω in (8) and (5), the B matrix in (10) can be factorized as

$$B = B_\tau B_\Omega = \underbrace{\begin{bmatrix} 0 \\ I_4 \end{bmatrix}}_{B_\nu} B_2 \tag{11}$$

where

$$B_2 = \begin{bmatrix} -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b & -\frac{1}{m}b \\ 0 & 0 & -\frac{1}{Ixx}bl & -\frac{1}{Ixx}bl & 0 & 0 & \frac{1}{Ixx}bl & \frac{1}{Ixx}bl \\ \frac{1}{Iyy}bl & \frac{1}{Iyy}bl & 0 & 0 & -\frac{1}{Iyy}bl & -\frac{1}{Iyy}bl & 0 & 0 \\ -\frac{1}{Izz}d & -\frac{1}{Izz}d & \frac{1}{Izz}d & \frac{1}{Izz}d & -\frac{1}{Izz}d & -\frac{1}{Izz}d & \frac{1}{Izz}d & \frac{1}{Izz}d \end{bmatrix} \tag{12}$$

Note that the decompositions in equations (10) and (11) are standard in control allocation problems [3, 13] and will be exploited here in order to achieve FTC. In particular, the structure in (11) allows ‘perfect factorization’ of the input distribution matrix.

3 Sliding Mode Control Synthesis

3.1 Control Allocation

In the event of a fault/failure occurring in any of the rotors, equation (10) can be written as

$$\dot{x}(t) = Ax(t) + BWu(t) + D\Omega_r(t) \tag{13}$$

where $W = diag(w_1 \ldots w_8)$ represent the effectiveness of the rotors. The scalars w_i model each individual rotor effectiveness level and satisfy $0 \leq w_i \leq 1$. In the fault free case $w_i = 1$, in the faulty case $w_i < 1$, and when $w_i = 0$ the rotor has failed totally. Note that Ω_r is ‘matched uncertainty’ [10] due to the structure of D in (8).

First define

$$\nu(t) := B_2 u(t) \tag{14}$$

From (14) the control signal $u(t)$ can be written as

$$u(t) = B_2^\dagger \nu(t) \tag{15}$$

where $B_2^\dagger$ is the right pseudo inverse of B_2 defined as

$$B_2^\dagger := WB_2^{\mathrm{T}}(B_2 W B_2^{\mathrm{T}})^{-1} \tag{16}$$

Using the fact that $B = B_\nu B_2$, from (11) and using (14)-(16), equation (13) can be written as

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B_\nu B_2 W B_2^\dagger \nu(t) + D\Omega_r(t) \\ &= Ax(t) + B_\nu \hat{\nu}(t) + D\Omega_r(t)\end{aligned} \tag{17}$$

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

$$\hat{\nu}(t) := B_2 W^2 B_2^{\mathrm{T}} (B_2 W B_2^{\mathrm{T}})^{-1} \nu(t) \tag{18}$$

Note that in the sliding mode literature, faults and uncertainty with the structure of Ω_r in (17), are classified as 'matched uncertainty' [10, 4, 25]. Furthermore sliding modes are inherently robust against such a class of uncertainty [10, 25]. Therefore in the case when a fault occurs in any of the rotors, sliding modes will reject such effects. In the case when a total failure occurs (provided enough redundancy still exists in the system so that $\det(B_2 W B_2^{\mathrm{T}}) \neq 0$), control allocation can be used to redistribute the control signals to the remaining healthy rotors.

3.2 Sliding Mode

Equation (17) can be written in detail as

$$\underbrace{\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}}_{A} \underbrace{\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix}0\\ I_4\end{bmatrix}}_{B_\nu} \hat{\nu}(t) + \underbrace{\begin{bmatrix}0\\ D_2\end{bmatrix}}_{D} \Omega_r(t) \tag{19}$$

Note that (19) is naturally in 'regular form' [10, 4] due to the structure of B_ν. Therefore any standard sliding mode scheme in the literature (e.g. [10, 4]) can be used to directly design the virtual controller.

To synthesize the 'virtual' control $\nu(t)$, define a switching function $s(t)$ to be

$$s(t) = Sx(t) \tag{20}$$

where $S \in \mathbb{R}^{4\times 8}$ and $SB_\nu = I$. Let $\mathscr{S}$ be the hyperplane $\mathscr{S} = \{x \in \mathbb{R}^8 \ : \ Sx = 0\}$. If a control law can be developed which forces the closed-loop trajectories onto the surface $\mathscr{S}$ in finite time and constrains the states to remain there, then an ideal sliding motion is said to have been attained [10, 4]. The selection of the sliding surface is the first part of any sliding mode design and defines the system's closed-loop performance. The second aspect of the control design, is the synthesis of a control law to guarantee that the surface is reached in finite time and a sliding mode is maintained.

In the regular form coordinates as given in (refeq:lin4a), a suitable choice for the sliding surface matrix is

$$S = \begin{bmatrix} M & I_4 \end{bmatrix} \tag{21}$$

where $M \in \mathbb{R}^{4\times 4}$ represents design freedom. Introduce a transformation so that $(x_1, x_2) \mapsto Tx = (x_1, s)$ associated with the nonsingular matrix

$$T_s = \begin{bmatrix} I_4 & 0 \\ M & I_4 \end{bmatrix} \tag{22}$$

In the new coordinates, equation (19) then becomes

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{s}(t) \end{bmatrix} = \begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} x_1(t) \\ s(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I_4 \end{bmatrix} \hat{v}(t) + \begin{bmatrix} 0 \\ D_2 \end{bmatrix} \Omega_r(t) \tag{23}$$

where $\hat{A}_{11} := A_{11} - A_{12}M$, $\hat{A}_{21} := M\hat{A}_{11} + A_{21} - MA_{22}$ and $\hat{A}_{22} = MA_{12} + A_{22}$. If a control law can be designed to induce sliding, then during sliding $\dot{s}(t) = s(t) = 0$, and the reduced order sliding motion is given by the top partition of (23):

$$\dot{x}(t) = \hat{A}_{11}\hat{x}_1(t) \tag{24}$$

In (24), the matrix $\hat{A}_{11} := A_{11} - A_{12}M$ can be made stable by choice of M in (21). The selected control law comprises linear and nonlinear components given by

$$\hat{v}(t) = \hat{v}_l(t) + \hat{v}_n(t) \tag{25}$$

The linear component is defined as

$$\hat{v}_l(t) = -\hat{A}_{21}\hat{x}_1(t) - (\hat{A}_{22} - \Phi)s(t) \tag{26}$$

where $\Phi \in \mathbb{R}^{4\times 4}$ is any stable design matrix and the nonlinear component

$$\hat{v}_n(t) = -\rho(t,x)\frac{P_2 s(t)}{\|P_2 s(t)\|} \qquad \text{if } s(t) \neq 0 \tag{27}$$

where $P_2 \in \mathbb{R}^{4\times 4}$ is a s.p.d matrix satisfying the Lyapunov equation

$$P_2\Phi + \Phi^{\mathrm{T}}P_2 = -I_4 \tag{28}$$

The problem of determining the stability of the closed-loop system under the influence of matched uncertainty becomes the problem of ensuring that sliding occurs despite the presence of uncertainty or faults.
Assumption: the signal Ω_r from (4) is considered as uncertainty and is assumed to be bounded and satisfies

$$\|\Omega_r(t)\| \leq \gamma\|\hat{v}(t)\| + \alpha(t,x) \tag{29}$$

where $\alpha(\cdot)$ is a known function while the gain γ satisfies $\gamma\|D_2\| < 1$.

The following proposition shows that if (29) is satisfied, the controller in (25) will still induce sliding in the presence of the 'matched uncertainty'.

Proposition 1. *If the matrix M has been chosen so that $\hat{A}_{11} = A_{11} - A_{12}M$ is stable, then choosing*

$$\rho(t,x) \geq \frac{\|D_2\|\,(\gamma\|\hat{v}_l\| + \alpha(t,x)) + \eta}{(1 - \gamma\|D_2\|)} \tag{30}$$

where η is a positive scalar and γ is a known constant, ensures a sliding motion takes place on $\mathscr{S}$ in finite time.

Proof: Substituting the control law in (25)-(27) into system (23) gives:

$$\dot{\hat{x}}_1(t) = \hat{A}_{11}\hat{x}_1(t) + \hat{A}_{12}s(t) \tag{31}$$

$$\dot{s}(t) = \Phi s(t) - \rho(t,x)\frac{P_2 s}{\|P_2 s\|} + D_2\Omega_r(t) \tag{32}$$

Consider a Lyapunov function $V(s) = s^{\mathrm{T}}P_2 s$ for (32). Differentiating the Lyapunov function yields:

$$\begin{aligned}
\dot{V} &= \dot{s}^{\mathrm{T}}P_2 s + s^{\mathrm{T}}P_2\dot{s}\\
&= \left(\Phi s - \rho\frac{P_2 s}{\|P_2 s\|} + D_2\Omega_r(t)\right)^{\mathrm{T}} P_2 s + s^{\mathrm{T}}P_2\left(\Phi s - \rho\frac{P_2 s}{\|P_2 s\|} + D_2\Omega_r(t)\right)\\
&= s^{\mathrm{T}}\left(\Phi^{\mathrm{T}}P_2 + P_2\Phi\right)s - 2\rho\frac{1}{\|P_2 s\|}\left(s^{\mathrm{T}}P_2P_2 s\right) + 2s^{\mathrm{T}}P_2D_2\Omega_r(t)\\
&= -\|s\|^2 - 2\rho\,\|P_2 s\| + 2s^{\mathrm{T}}P_2D_2\Omega_r(t)
\end{aligned} \tag{33}$$

since $s^{\mathrm{T}}s = \|s\|^2$, $s^{\mathrm{T}}P_2P_2 s = \|P_2 s\|^2$ and $\Phi^{\mathrm{T}}P_2 + P_2\Phi = -I$. Furthermore since

$$\|s^{\mathrm{T}}P_2D_2\Omega_r(t)\| < \|P_2 s\|\|D_2\|\|\Omega_r(t)\|$$

from the Cauchy-Schwartz inequality,

$$\dot{V} \leq -\|s\|^2 - 2\|P_2 s\|\left(\rho - \|D_2\|\|\Omega_r(t)\|\right) \tag{34}$$

The idea is to represent ρ in (34) in terms of the uncertainty Ω_r using the definition of ρ given in (30). From (25) and (27), and using the triangle inequality property of norms

$$\|\hat{v}(t)\| \leq \|\hat{v}_l(t)\| + \|\hat{v}_n(t)\| \leq \|\hat{v}_l(t)\| + \rho \tag{35}$$

Equation (30) can be written as

$$\rho(t,x)\left(1 - \gamma\|D_2\|\right) \geq \|D_2\|\left(\gamma\|\hat{v}_l\| + \alpha(t,x)\right) + \eta \tag{36}$$

Rearranging this equation yields

$$\begin{aligned}
\rho(t,x) &\geq \|D_2\|\big(\gamma\|\hat{v}_l\| + \alpha(t,x)\big) + \eta + \rho(t,x)\gamma\|D_2\|\\
&\geq \|D_2\|\big(\gamma\|\hat{v}_l\| + \rho(t,x)\gamma + \alpha(t,x)\big) + \eta
\end{aligned}$$

Using (35) and (29), the above can be written as

$$\rho(t,x) \geq \|D_2\|\left(\gamma\|\hat{v}\| + \alpha(t,x)\right) + \eta \geq \|D_2\|\|\Omega_r\| + \eta \tag{37}$$

Substituting for (37) in (34) yields

$$\begin{aligned}\dot{V} &\leq -\|s\|^2 - 2\|P_2 s\|\left(\|D_2\|\|\Omega_r\| + \eta\right) + 2\|P_2 s\|\,\|D_2\|\|\Omega_r\| \\ &\leq -\|s\|^2 - 2\eta\|P_2 s\| \end{aligned} \tag{38}$$

Equation (38) shows that the controller in the form (25)-(27), induces ideal sliding in the presence of matched uncertainty. This inequality will be used to show that sliding on $\mathscr{S}$ takes place in finite time. From the Rayleigh principle

$$\|P_2 s\|^2 = (P_2^{1/2}s)^{\mathrm{T}}P_2(P_2^{1/2}s) \geq \lambda_{min}(P_2)\|P_2^{1/2}s\|^2 = \lambda_{min}(P_2)V(s) \tag{39}$$

which together with (38) gives

$$\dot{V} \leq -2\eta\sqrt{\lambda_{min}(P_2)}\sqrt{V} \tag{40}$$

Integrating (40) implies that the time taken to reach the sliding surface $\mathscr{S}$ denoted by t_s satisfies

$$t_s \leq \eta^{-1}\sqrt{V(s_0)/\lambda_{min}(P_2)} \tag{41}$$

where s_0 represents the initial value of $s(t)$ at $t = 0$ [10]. ■

The final control law is obtained using (15)-(16) and (18) which establishes

$$u(t) = WB_2(B_2W^2B_2^{\mathrm{T}})^{-1}\hat{v}(t) \tag{42}$$

Note that the control which is sent to the actuators is dependent on the effectiveness gains w_i (from the diagonal weighting matrix W) and therefore w_i must be estimated from a fault detection and isolation (FDI) scheme.

4 Design and Simulations

The octorotor parameters are inspired by the quadrotor parameters taken from [6]. Most of the parameters such as weight and inertia, are double those from the quadrotor described in [6]. Specifically the parameters are: $m = 1.3(kg)$, $I_{xx} = I_{yy} = 0.0150(kgm^2)$, $I_{xx} = 0.0026(kgm^2)$, $b = 3.13\times10^{-5}(Ns^2)$, $d = 7.5\times10^{-7}(Nms^2)$, $J_r = 6.0\times10^{-5}(Kgm^2)$ and $L = 0.23(m)$. For design, the trim condition used for linearization is steady hover at an altitude of 10m. Solving for Ω_1 during a hover condition gives each rotor the initial condition $\Omega_i(0) = \sqrt{mg/8b} = 225.5628$ (rpm) which translates to $\tau(0) = \begin{bmatrix}12.74\ 0\ 0\ 0\end{bmatrix}^{\mathrm{T}}$. The linear matrices used for design are given by

$$A = \begin{bmatrix} 0_{4\times4} & I_4 \\ 0_{4\times4} & 0_{4\times4} \end{bmatrix}, \quad B_v = \begin{bmatrix} 0_{4\times4} \\ I_4 \end{bmatrix} \tag{43}$$

To include a tracking facility, integral action [10, 4] has been included in the design. Let $x_c(t)$ represent integral action states given by

$$\dot{x}_c(t) = y_c(t) - C_c x(t) \tag{44}$$

where $C_c = \begin{bmatrix} I_4 & 0_{4\times4} \end{bmatrix}$ is the distribution matrix associated with the controlled outputs, and $y_c(t)$ is the (differentiable) reference signal [10] which represents the z axis position, roll, pitch and yaw angle commands. The differentiable (filtered reference) signal $y_c(t)$ is assumed to satisfy

$$\dot{y}_c(t) = \Gamma\,(y_c(t) - Y_c) \tag{45}$$

where $\Gamma \in \mathbb{R}^{4\times4}$ is a stable design matrix and Y_c is a constant demand vector [10]. Augmenting the states from (43) with the integral action states and then defining $x_a(t) = \mathrm{col}(x_c(t), x(t))$, it follows that

$$\dot{x}_a(t) = A_a x_a(t) + B_a u(t) + B_c y_c(t) \tag{46}$$

where

$$A_a = \begin{bmatrix} 0 & -C_c \\ 0 & A \end{bmatrix} \quad B_a = \begin{bmatrix} 0 \\ B_v \end{bmatrix} \quad B_c = \begin{bmatrix} I_4 \\ 0 \end{bmatrix} \tag{47}$$

It is easy to check (A_a, B_a) is controllable [10]. Define a switching function

$$s_a(t) = S_a x_a(t) = \begin{bmatrix} M_a & I_4 \end{bmatrix} \tag{48}$$

where $M_a \in \mathbb{R}^{4\times8}$. As in (25) the proposed 'virtual control' law comprises two components $\hat{v}(t) = \hat{v}_l(t) + \hat{v}_n(t)$. Now because of the reference signal $y_c(t)$, the linear component has a feed-forward reference term and $\hat{v}_l(t) = L_a x_a(t) + L_c y_c(t)$ where $L_a = -(S_a A_a - \Phi_a S_a)$ and $L_c = -M_a B_c$. Here A_a, B_a and S_a are the matrices from (47) and (48) which are already in regular form while Φ_a is the design freedom (analogous to Φ in (28)). Note that an extra term L_c has appeared in this tracking formulation compared to the one in (26). The nonlinear component is defined as

$$\hat{v}_n(t) = -\rho(t, x_a)\tfrac{s_a(t)}{\|s_a(t)\|} \quad \text{for } s_a(t) \neq 0 \tag{49}$$

A quadratic optimal design has been used to obtain the sliding surface matrix S_a (see for example [25, 10]). The symmetric positive definite weighting matrix has been chosen as $Q = diag(10, 100, 100, 100, 1, 1, 1, 1, 1, 1, 1, 1)$ while the parameter $\Phi_a = \mathrm{diag}(-10, -10, -10, -10)$. The poles associated with the reduced order sliding motion are

$$\{-1.3532 \pm 1.1537i, -2.2913 \pm 2.1794i, -2.2913 \pm 2.1794i, -2.2913 \pm 2.1794i\}$$

The pre-filter from (45) has been chosen as $\Gamma = -20I_4$. In the simulations the discontinuity in the nonlinear control term in (49) has been smoothed by using a sigmoidal approximation $\frac{s_a}{\|s_a\| + \delta_a}$, where the scalar δ_a has been chosen as $\delta_a = 0.01$ (see for example §3.7 in [10]). This removes the discontinuity and introduces a further degree of tuning to accommodate the motor limits – especially during actuator fault or failure conditions. Finally the nonlinear modulation gain ρ from (27) has been chosen as $\rho = 10$.

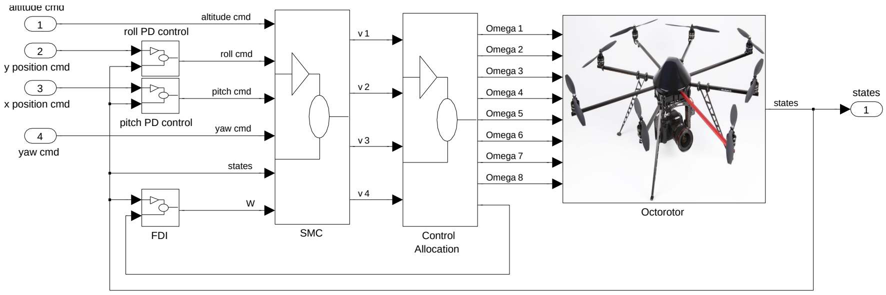


Fig. 1 Controller Architecture

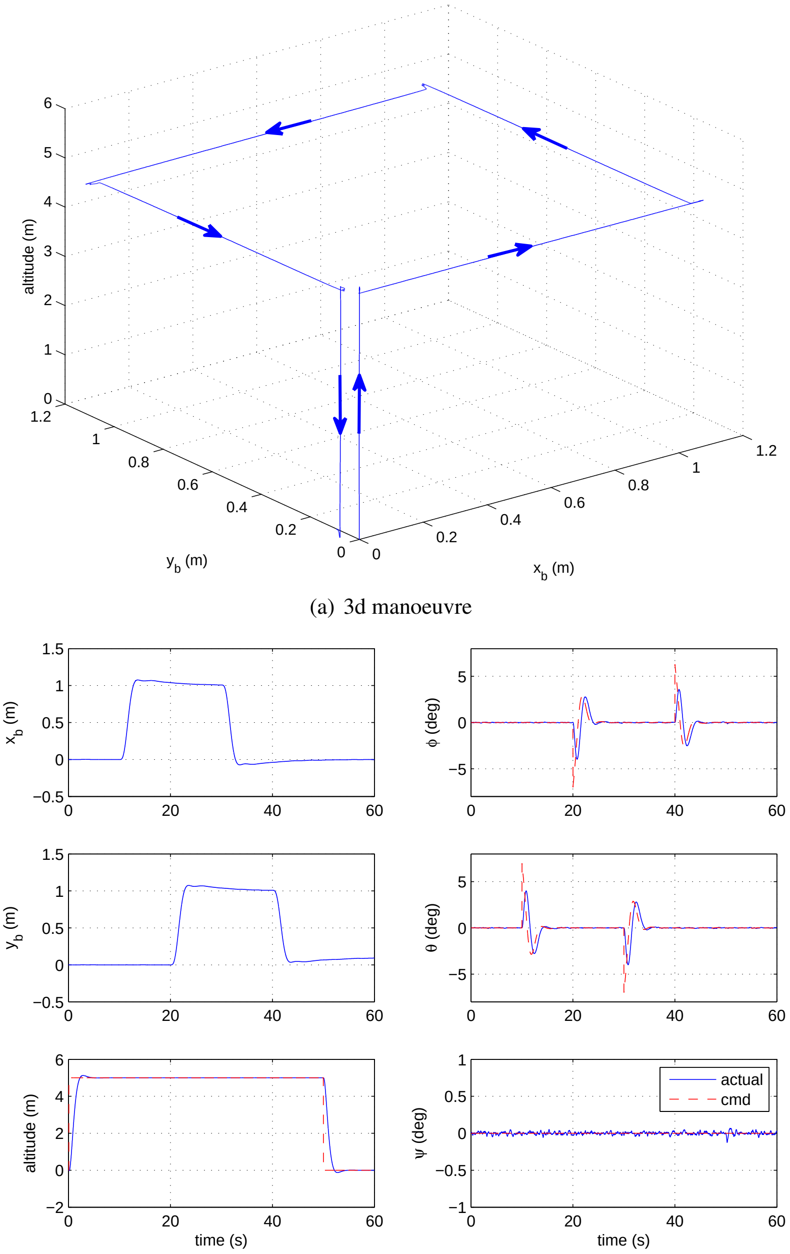
4.1 Simulation Results

Note that although the design is based on a linear model, the *simulation tests have been conducted on the nonlinear model* in (1). A gust condition is also included in the simulation, where band limited white noise with a variance of 1×10^{-3} is applied to the p, q and r channels.

Note that during the simulations, the motor speed is limited to a range $\begin{bmatrix} 0 & 600 \end{bmatrix}$ (rpm). For position control in the x and y direction, a PD control has been used to provide roll and pitch commands while the yaw command is set to zero (see Figure 1 for the architecture of the overall control scheme). The gains for the x and y position PD controller are $k_{p,x} = 1, k_{d,x} = 7$ and $k_{p,y} = 1, k_{d,y} = 7$ respectively. Note that the scheme requires the effectiveness levels $w_1 \dots w_8$ of each rotor – which are assumed to be available from an FDI unit. In the simulations, this is obtained by directly comparing the demanded and the actual rotor speed. This is possible in practice, since modern motors have a built-in encoder, which allows motor speed to be measured in real time. Alternatively, any FDI scheme such as the fault reconstruction methodology described in [4] can also be used.

4.1.1 Fault Free

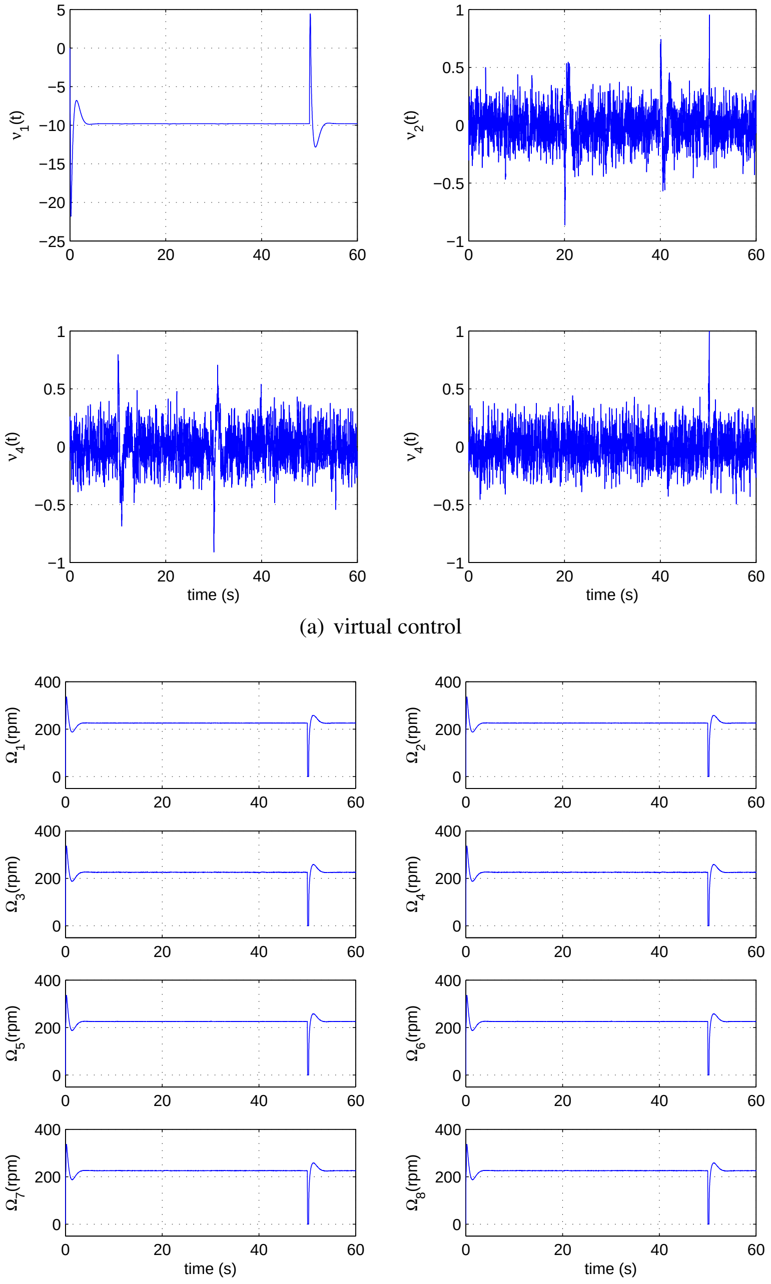
Figures 2-3 show the results for a fault free scenario. Figure 2(a) shows the overall manoeuvres. These manoeuvres will also be used in the fault/failure scenarios to make a performance comparison to the fault free case. The simulation starts with an immediate increase in altitude followed by a series of position changes and finally a landing close to the original starting point. Figure 2(b) shows good state tracking performance and the roll, pitch, yaw and altitude closely track the command signals. Figure 3(a) shows the virtual control signals produced by the controller while Figure 3(b) shows the speed for each rotor.



(a) 3d manoeuvre

(b) states

Fig. 2 Fault free: position and states



(a) virtual control

(b) rotor speed

Fig. 3 Fault free: virtual control and rotor speed

4.1.2 Rotor 2,4,6,8 failure

Figure 4(b) shows the rotor speeds when rotors 2, 4, 6 and 8 are totally failed after 20sec during a change in the x_b position. Figure 4(a) shows the states of both the fault free and failure cases, to show the difference between the two cases. It can be seen that despite the failure to 4 rotors, there is no visible change in terms of the performance (lines overlap), thus highlighting the efficacy of the proposed scheme. Note that compared to [2], the scheme proposed here can deal with more than one single rotor failure, by taking advantage of the available redundancy through control allocation.

4.1.3 Rotor 1,4,6,7 failure

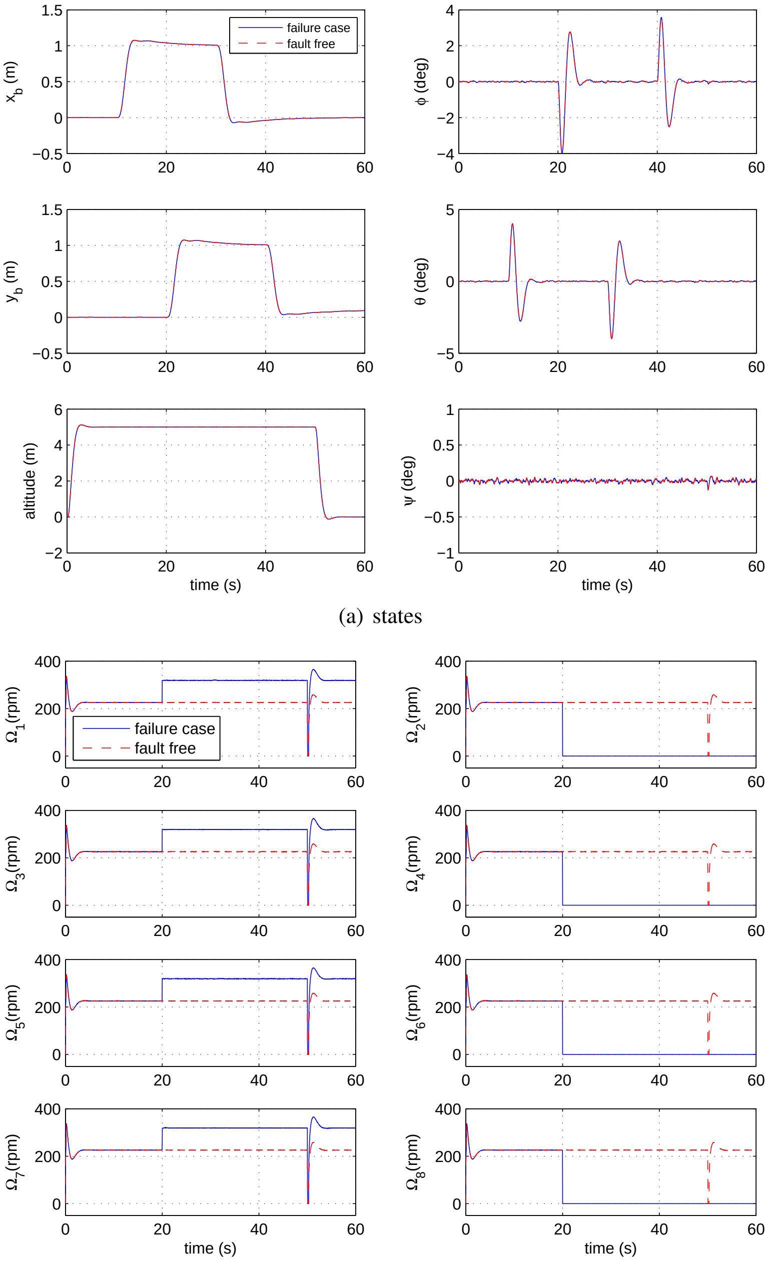
Figure 5 shows simulation results when a different set of rotors (rotor 1, 4, 6 and 7) fail at 20 sec. Figure 5(b) clearly shows the speed of rotors 1, 4, 6 and 7 has dropped to zero at 20 sec – which simulates the failure. The control signal is redistributed to rotors 2, 3, 5 and 7 which show an increase in speed in order to compensate for the failed rotors. Again Figure 5(a) shows no visible degradation in performance when compared to the fault free case (the lines overlap) which highlights the efficacy of the proposed scheme.

4.1.4 Rotor 1,3,5,7 failure

Figure 6 shows the effect when rotors 1, 3, 5 and 7 fail at different times during the manoeuvre. Figure 6(b) shows rotors 1, 3, 5, 7 fail at 20, 30, 40 and 50 sec respectively. It can be seen that the rotor speed for rotors 2, 4, 6, 8 increases to compensate for the effect of rotor failures. Figure 6(a) shows no visible difference in terms of state tracking performance (the lines overlap) despite the failure of 4 of the rotors.

4.1.5 Rotor 2,4,6,8 fault and rotor 1,3 failure

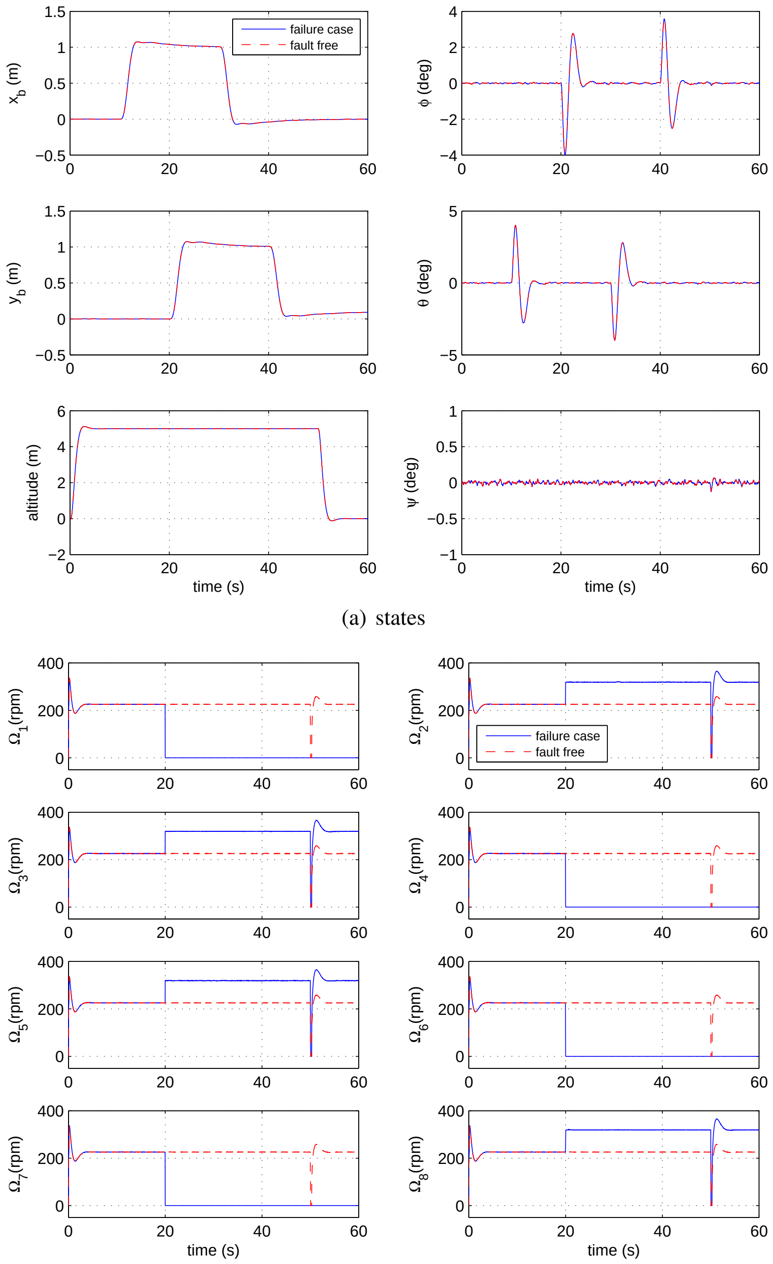
Figure 7(b) shows a more severe fault/failure scenario where rotors 2, 4, 6 and 8 are only 50% effective from 20 sec onwards, and then rotor 1 and 3 fail totally at 30sec. Despite the fact that 4 rotors are only 50% effective and 2 rotors have totally failed, Figure 7(a) shows no visible difference in terms of state performance compared to the fault free case (lines overlap), and thus the proposed scheme has managed to maintain fault free performance.



(a) states

(b) rotor speed

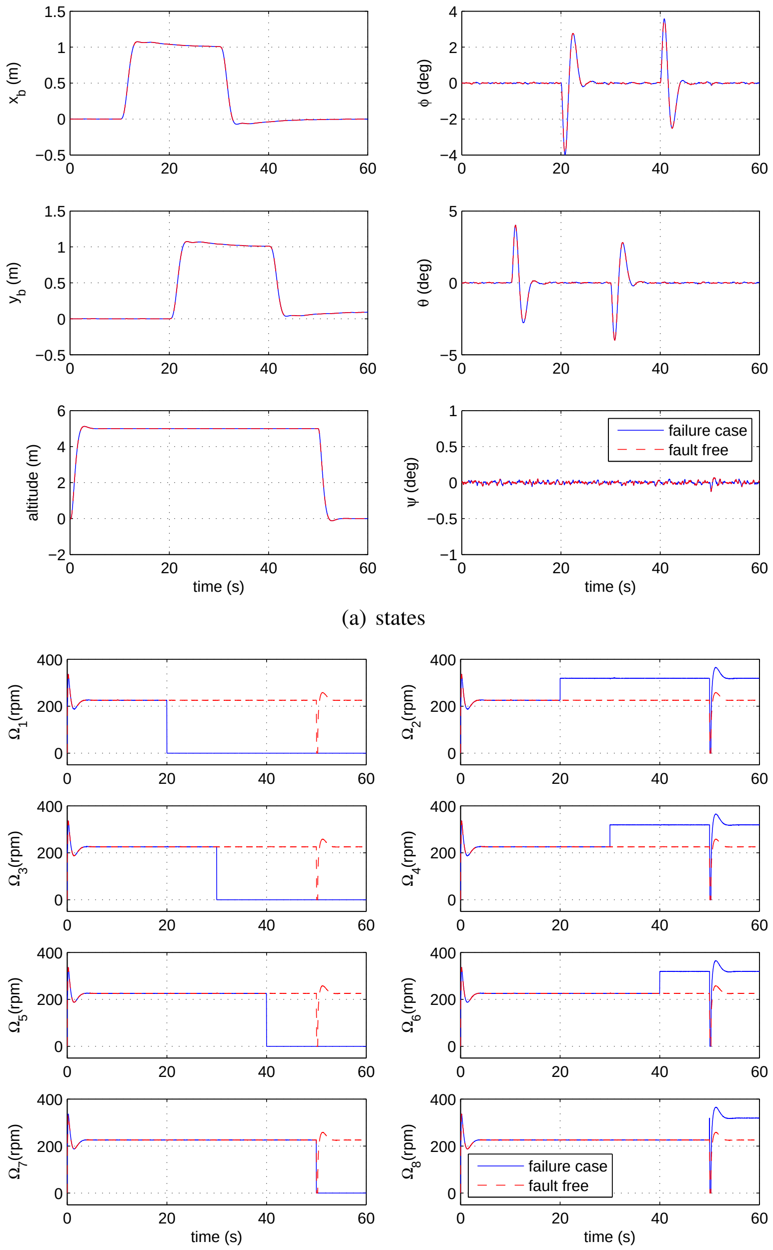
Fig. 4 Rotor 2,4,6,8 failure



(a) states

(b) rotor speed

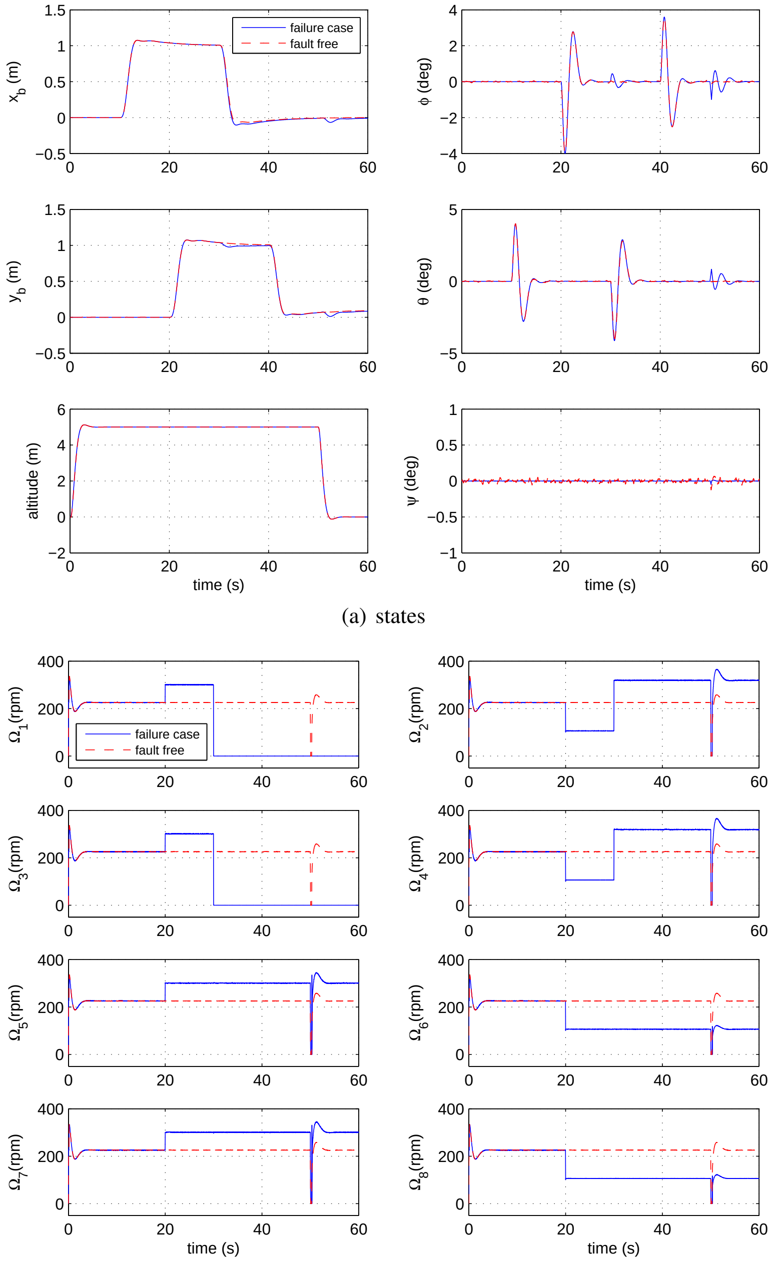
Fig. 5 Rotor 1,4,6,7 failure



(a) states

(b) rotor speed

Fig. 6 Rotor 1,3,5,7 failure at different time



(a) states

(b) rotor speed

Fig. 7 Rotor 2,4,6,8 fault and rotor 1,3 failure

5 Conclusions

This paper has presented a sliding mode control allocation scheme for an octorotor. The scheme takes full advantage of the redundant rotors to handle more than one rotor failure as compared to the existing literature. The core controller is based on a sliding mode design which is robust against faults and uncertainty in the input channels (matched uncertainty). When a fault/failure occurs, no reconfiguration is required and the control effort to the faulty rotors is re-allocated to the healthy ones using control allocation. Simulation results show no visible change in performance (as compared to the fault free case) for various types of fault/failure scenarios, highlighting the efficacy of the proposed scheme.

References

1. V.G. Adîr and A.M. Stoica. Integral LQR control of a star-shaped octorotor. *INCAS Bulletin*, 4(2):3–18, 2012.
2. V.G. Adîr, A.M. Stoica, A. Marks, and J.F. Whidborne. Modelling, stabilization and single motor failure recovery of a 4Y octorotor. In *13th IASTED International Conference on Intelligent Systems and Control (ISC 2011), Cambridge, U.K*, 2011.
3. H. Alwi and C. Edwards. Fault tolerant control using sliding modes with on-line control allocation. *Automatica*, 44(7):1859–1866, 2008.
4. H. Alwi, C. Edwards, and C. P. Tan. *Fault Detection and Fault–Tolerant Control Using Sliding Modes*. Advances in Industrial Control. Springer-Verlag, 2011.
5. C. Balas. Modelling and linear control of a quadrotor. Master of Science Thesis, School of Engineering, Cranfield, 2007.
6. S. Bouabdallah. *Design and control of quadrotors with application to autonomous flying*. PhD thesis, École Polytechnique Fédérale De Lausanne, 2007.
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. G. Ducard and M-D Hua. Discussion and practical aspects on control allocation for a multi-rotor helicopter. In *1st International Conference on UAVs in Geomatics, UAV-g 2011, Zurich, Switzerland*, 2011.
9. C. Edwards, T. Lombaerts, and H. Smaili (Eds.). *Fault Tolerant Flight Control: A Benchmark Challenge*, volume 399. Springer-Verlag: Lecture Notes in Control and Information Sciences, 2010.
10. C. Edwards and S. K. Spurgeon. *Sliding Mode Control: Theory and Applications*. Taylor & Francis, 1998.
11. 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.
12. M. Hamayun, H. Alwi and C. Edwards, An LPV Fault Tolerant Control Scheme Using Integral Sliding Modes 51st IEEE Conference on Decision and Control, Maui, Hawaii, 2012.
13. O. Härkegård and S. T. Glad. Resolving actuator redundancy - optimal control vs. control allocation. *Automatica*, 41(1):137–144, 2005.
14. 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.
15. M. Labadille. Non-linear control of a quadrotor. Master of Science Thesis, School of Engineering, Cranfield, 2007.

16. T. Madani and A. Benallegue. Backstepping sliding mode control applied to a miniature quadrotor flying robot. In *32nd IEEE Conference on Industrial Electronics, IECON*, 2006.
17. 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.
18. T. Madani and A. Benallegue. Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles. In *American Control Conference, New York City, USA,*, 2007.
19. V. M. Martínez. Modelling of the flight dynamics of a quadrotor helicopter. Master of Science Thesis, School of Engineering, Cranfield, 2007.
20. 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.
21. A.S. Sanca, P.J. Alsina, and J.d.J.F. 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.
22. T. Schneider, G. Ducard, K. Rudin, and P. Strupler. Fault-tolerant control allocation for multirotor helicopters using parametric programming. In *International Micro Air Vehicle Conference and Flight Competition, Braunschweig, Germany*, 2012.
23. F. Sharifi, M. Mirzaei, B.W. 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.
24. 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*, 83(6):407–419, 2011.
25. V. I. Utkin. *Sliding Modes in Control Optimization*. Springer-Verlag, Berlin, 1992.
26. 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.
27. 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.
28. Y. Zhang and A. Chamseddine. Fault tolerant flight control techniques with application to a quadrotor UAV testbed. In T. Lombaerts, editor, *Automatic Flight Control Systems - Latest Developments*, pages 119–150. InTech, 2012.