Flight Evaluations of Sliding Mode Fault Tolerant Controllers

C. Edwards[1], L. Chen[1], A. Khattab[1], H. Alwi[1], M. Sato[2]

Abstract

This paper considers the development of fault tolerant controllers (FTC) and their application to aerospace system. In particular, given the extensive and growing literature in this area, this paper focusses on methods where the schemes have been implemented and flight tested. One thread of the fault tolerant control literature has involved sliding mode controllers. This paper considers a specific class of sliding mode FTC which incorporates control allocation to exploit over-actuation (which is typically present in aerospace systems). The paper describes implementations of these ideas on a small quadrotor UAV and also piloted flight tests on a full-scale twin-engined aircraft.

I. Introduction

Flight has fascinated mankind for thousands of years. When Otto Lilienthal flew for the first time a heavier-than-air vehicle in 1891 he implicitly created a new challenge: controlled flight. Largely out of necessity, he invented the concept of (lateral) stabilization using a rudder [1]. After the first successful powered flight of the Wright Brothers in 1903, the first 'controlled flight' was demonstrated in 1914 by Lawrence Sperry [1]. Sperry and others perfected the concept of automatic stabilized flight which has ultimately led to modern computer controlled fly-by-wire systems based on feedback control [2]. However since traditional feedback control schemes do not inherently have the ability to handle system faults/failures, sensor/actuator/component failures can lead to a dramatic degradation in system performance and loss of stability.

In civil aircraft, traditionally, fault tolerance has been achieved by using multiple redundant hardware: typically, three hydraulic actuators, three hydraulic lines and three hydraulic pumps are associated with each control surface [3]. Sensors in large civil aircraft also rely on triple redundancy: a quantity is usually measured by three independent devices and then a consolidated measurement from all three are then used in the flight control computer [3]. At an academic level at least analytic redundancy has gained popularity in the field of system control [4] due to its ability to deal with faults and/or failures without directly replicating hardware [3]. The main objective of Fault Tolerant Control (FTC) is to mitigate the effect of faults and failures and to maintain a certain (possibly degraded) level of performance [5].

In the last decades, Unmanned Aerial Vehicles (UAV) or 'drones' have gained increasingly important roles in both military and civil applications. UAVs can be deployed to undertake missions regarded as too dull, dirty or dangerous. UAVs can be used in dangerous environments such as nuclear power plants, polluted environments or forest fires for reconnaissance. Recently, large companies such as Amazon, DHL have started to consider the use of UAVs for package delivery (see Figure 1), while Facebook explored the concept of (High Altitude Long Endurance) HALE UAVs to provide internet coverage in rural areas. One of the recent exciting applications of UAVs is for autonomous self-flying passenger transport. A recent unveiling in Dubai of a giant autonomous 'quadrotor' with four arms and eight propellers (see Figure 2), for transporting passengers was proposed for use in service as soon as summer 2017. (In fact, Airbus also envisaged a future passenger transporting UAV to reflect the importance of the idea.)

With the increase in applications of UAVs, especially in civil and commercial applications, there is also pressure to ensure that the UAVs are safe to operate, especially in the absence of pilots. In many UAV applications it is the safety of people on

[1]College of Engineering, Mathematics and Physical Sciences, University of Exeter, EX4 4QF, UK. lc427,ak555,h.alwi,C.Edwards@exeter.ac.uk
[2] Japan Aerospace Exploration Agency, Mitaka, Tokyo 181-0015, Japan. sato.masayuki@jaxa.jp

Fig. 1. UAV civil applications: Cargo transport and delivery

Fig. 2. UAV civil applications: Self-flying taxi

the ground and property which is more important rather than the vehicle itself (since the vehicles are not costly). Up to this point in time most of the incident involving UAVs can arguably be attributed to the operator's shortcomings, however serious implications to the general public are undeniable. In the case of large manned passenger aircraft, in emergency situations, there have been may reported cases in the past where only the skilled action of the pilots has avoided catastrophe [6], [7], [3]. However in un-piloted UAVs, there is an urgent need for automated safety feature that allows for example a UAV to land safely in potentially unknown residential areas (i.e. airworthiness-like approval) [8].

II. Fault tolerant Control Flight Tests

Although the topic of fault tolerant control has been investigated in the last few decades, applications and validations on real aircraft have been scarce. It is envisioned however, that fault tolerant control will be implemented on UAVs before the technology is accepted for manned aircraft counterparts due to the pressing need to ensure safety in the absence of pilots. The following sections document implementations of fault tolerant control schemes on both civil aircraft and UAVs.

A. FTC hardware implementation: Civil and fighter aircraft

The earliest reported implementation of an FTC scheme on a civil aircraft is the Propulsion Controlled Aircraft (PCA) project [9] at NASA's Dryden Flight Research Centre. In this project a control system was designed to use only engine thrust to control the aircraft. The concept was to control the pitch angle using collective thrust (increasing the thrust to climb and decrease thrust to descend), while heading and roll angle control was achieved by using the left and right engines differentially. The PCA was implemented and flight tested on a McDonnell-Douglas MD-11 transport aircraft and F-15 fighter aircraft.

Another project at NASA Dryden called the Intelligent Fight Control System (IFCS) [10] was designed to provide online aerodynamic and stability parameter identification during fault or failure conditions. These identified parameters were then used to reconfigure the controller to help the pilot safely land the aircraft. The parameter identification was done based on an artificial neural network. The flight test on a highly-modified McDonnell Douglas F-15B Eagle aircraft took place on Dec 6, 2002.

In the integrated resilient aircraft control project (IRAC) [11], a modified F-18 fighter aircraft, was used for testing FTC schemes based on adaptive control to enable safe flight in the presence of structural damage, control surface failure or changes in aerodynamic characteristics.

A recent (2016-present) project called VISION is a Europe-Japan collaborative research project. One of the objectives of the project is to implement FDD and FTC schemes on JAXA's MuPAL-α, a manned fly-by-wire experimental aircraft (Figure 3) and the USOL K-50 UAV.

B. FTC hardware implementation: Multirotor UAV

Perhaps not surprisingly literature describing the implementation of FTC schemes on UAVs is more plentiful and on-going – especially on multirotor systems – since their small size allows them to be tested in relative safety in a controlled laboratory environment.

1) FTC on Quadrotors: In [12], a gain scheduled PID was designed and tested to match each fault situation and the appropriate gains were selected depending on the tracking error or actuator status. For implementation, a UAV called Qball-X4 developed by Quanser Inc. was used after being modified from a quadrotor (4 motors) to a hexacopter Qball-X6 (6 motors). The work in [12] compared LQR designs and MRAC schemes for scenarios involving propeller damage of up to 16% loss of rotor effectiveness.

Fig. 3. JAXA MuPAL-alpha experimental airplane

In [13], passive and active FTC were achieved experimentally based on sliding mode ideas with a comparison between the two methods. In [14], an AFTC scheme for a nonlinear model of a quadrotor was considered where model predictive control was combined with horizon estimation and an unscented Kalman filter for parameter estimation. The work also considers actuator saturation although it requires an accurate model for the free fault system to calculate the trim control signals. In [12], model reference adaptive control was tested without explicit FDI i.e. no information of the fault was required. The scheme was tested in the presence of partial damage to one propeller and loss of effectiveness in the total thrust.

In [15], SMC was used for active FTC while in [16] SMC was used for passive FTC. A comparison between the two methods in [16], showed that similar level of performance can be achieved by carefully tuning the passive control or using cascade SMC.

It has to be noted that most of the FTC implementation work on quadrotors described above only considers faults and not total actuator failures. The work in [17] and [18], however, represents some of the limited number of publications that consider total rotor failures for a quadrotor. Conventional control strategies cannot deal with total failure of one of the rotor whilst retaining nominal performance, but here the UAV was allowed to manoeuvre with free rotation or constant rotational speed in the yaw (vertical) axis while the remaining healthy rotors are used to maintain control of roll and pitch angles. The controller considered was based on an LQR scheme and the implementation results on a quadrotor showed a successful safe landing was still possible.

2) Octocopters or Hexarotors for FTC: From a practical point of view, the availability of redundant actuators allows the possibility of handling total failures to certain actuators. The work in [19] considers an octocopter UAV in a co-axial configuration. The FDI is based on a nonlinear sliding mode observer to provide information to a PID controller. A bank of PID controllers was designed offline for each fault case, and the gains were set up as a look-up table. In [20], adaptive estimators were used to detect and isolate faults in the accelerometer and gyroscope. The schemes also have the capability to estimate (bias) faults. A passive FTC was designed based on a nonlinear adaptive strategy and acceptable performance was achieved in the presence of faults. In [21], a hexacopter was considered in indoor experiments using a Vicon visual motion tracking system. The FTC scheme depended on a parametric programming control allocator to automatically redistribute control signals in the event of faults/failures.

III. LPV SLIDING MODE CONTROLLER DESIGN

This section considers a class of FTC controller created from combining sliding mode ideas with a simple form of control allocation [3].

Consider an LPV system with subject to actuator faults/failures in the form

$$\dot{x}_p(t) = A_p(\rho)x_p(t) + B_p(\rho)(I_m - K(t))u_p(t) \tag{1}$$

where $A(\rho) \in \mathbb{R}^{n\times n}$ and $B(\rho) \in \mathbb{R}^{n\times m}$. The state vector in (1) is denoted by $x_p \in \mathbb{R}^n$ and $u_p \in \mathbb{R}^m$ denotes the control input. In this paper it is assumed the state is available for control design as is the time varying parameter $\rho \in \mathbb{R}^{n_r}$. The parameter ρ is assumed to belong to a polytope $\Omega \subset \mathbb{R}^{n_r}$. Here for simplicity it is assumed $A(\rho)$ is affinely dependent on $\rho(t)$ so that

$$A(\rho) = A_0 + A_1\rho_1(t) \ldots + A_{n_r}\rho_{n_r}(t)$$

The diagonal weighting matrix $K(t)$ in (1), is defined as

$$K(t) := \mathrm{diag}(k_1(t), \ldots, k_m(t)) \tag{2}$$

where the $k_i(t) \in [\ 0 \quad 1\ ]$, for $i = 1 \ldots m$, are time varying scalars which model the loss of effectiveness of the actuators [3]. For a fault-free actuator $k_j(t) = 0$ and for a failed actuator $k_j(t) = 1$. When $0 < k_j(t) < 1$, the actuator is affected by the fault and behaves with reduced effectiveness.

Finally

$$y_c(t) = C_c x_p(t) \tag{3}$$

where $C_c \in \mathbb{R}^{l\times n}$ is used to denote the controlled outputs of interest. It is assumed that the system is over-actuated and $l \leq m$ and this redundancy will be handled by a control allocation mechanism [3].
To incorporate tracking, define integrator states according to

$$\dot{x}_r(t) = r(t) - C_c x_p(t) \tag{4}$$

where $r(t) \in \mathbb{R}^l$ represents the differentiable command signal
Define an augment state according to $x = col(x_r, x_p)$ and a corresponding augmented system

$$\dot{x}(t) = \underbrace{\begin{bmatrix} 0 & -C_c \\ 0 & A_p(\rho) \end{bmatrix}}_{A(\rho)} x(t) + \underbrace{\begin{bmatrix} 0 \\ B_p(\rho) \end{bmatrix}}_{B(\rho)} (I - K(t))u(t) + \underbrace{\begin{bmatrix} I_l \\ 0 \end{bmatrix}}_{B_c} r(t) \tag{5}$$

In this paper, it is assumed the input distribution matrix $B(\rho)$ in (5) can be factorized into

$$B(\rho) = B_v B_2(\rho) \tag{6}$$

where $B_v \in \mathbb{R}^{(n+l)\times l}$ is rank l, and only $B_2(\rho) \in \mathbb{R}^{l\times m}$ depends on the scheduling parameters.

Assumption 3.1: Assume $\text{rank}(B_2(\rho)) = l$ for all $\rho \in \Omega$

Remark 3.1: This factorization can always be achieved (at least approximately in aerospace systems [22], [3].

Assumption 3.2: The pair $(A(\rho), B_v)$ is controllable.

Since by assumption $\text{rank}(B_v) = l$, there exists a coordinate transformation $x \mapsto \bar{x}$ which leads to a new state-space representation in which the input distribution has the form

$$\bar{B_v} = \begin{bmatrix} 0 \\ I_l \end{bmatrix} \tag{7}$$

and (5) can be written as

$$\dot{\bar{x}}(t) = A(\rho)\bar{x}(t) + \begin{bmatrix} 0 \\ B_2(\rho)(I - K(t)) \end{bmatrix} u(t) + B_c r(t) \tag{8}$$

Define a virtual control $v(t) \in \mathbb{R}^l$ as

$$v(t) := B_2(\rho)u(t) \tag{9}$$

This virtual control will subsequently be designed based on sliding mode control principles. Suppose the actual physical control signals sent to the actuators is given by

$$u(t) := B_2(\rho)^\dagger v(t) \tag{10}$$

where $B_2(\rho)^\dagger = B_2(\rho)^T (B_2(\rho)B_2(\rho)^T)^{-1}$ then as a consequence, substituting from (10), equation (8) can be written as

$$\dot{\bar{x}}(t) = A(\rho)\bar{x}(t) + \begin{bmatrix} 0 \\ I \end{bmatrix} v - \begin{bmatrix} 0 \\ B_2(\rho)K(t)B_2(\rho)^\dagger \end{bmatrix} v(t) + B_c r(t) \tag{11}$$

The virtual control $v(t)$ will be designed from (11) in the nominal case when $K(t) = 0$: specifically for the nominal system

$$\dot{\bar{x}}(t) = \bar{A}(\rho)\bar{x}(t) + B_v v + B_c r(t) \tag{12}$$

where B_v is defined in (7).

A. Definition of the switching function and control law

Closed-loop stability of the system in (11) for all admissible values of $K(t)$. Partition the states as $\bar{x} = col(\bar{x}_1, \bar{x}_2)$ and partition $\bar{A}(\rho)$ and B_c in (11) as

$$\bar{A}(\rho) = \begin{bmatrix} A_{11}(\rho) & A_{12}(\rho) \\ A_{21}(\rho) & A_{22}(\rho) \end{bmatrix}, B_c = \begin{bmatrix} B_{c1}(\rho) \\ B_{c2}(\rho) \end{bmatrix} \tag{13}$$

where $A_{11}(\rho) \in \mathbb{R}^{n\times n}$.

Define a classical linear switching function according to

$$s(t) := S\bar{x}(t) \tag{14}$$

where

$$S = \begin{bmatrix} M & I_l \end{bmatrix} \tag{15}$$

and $M \in \mathbb{R}^{l\times n}$ represents the design degree of freedom.

During sliding $\dot{s}(t) = s(t) = 0$, and the reduced order sliding motion is governed by

$$\dot{\bar{x}_1}(t) = \hat{A}_{11}(\rho)\bar{x_1}(t) + B_{c1} r(t) \tag{16}$$

Fig. 4. 3DR IRIS+ quadrotor

where $\hat{A}_{11}(\rho) = (A_{11}(\rho) - A_{12}(\rho)M)$. Since $\hat{A}_{11}(\rho)$ is dependent on M, the choice of M can be viewed as a state feedback problem for quadratically stabilizing the pair $(A_{11}(\rho), A_{12}(\rho))$.

The structure of the proposed virtual control law is

$$v = -SA(\rho)x(t) - \Phi s(t) - \hat{B}_{c2}r(t) + v_n \tag{17}$$

where $\hat{B}_{c2} = SB_c$ and $\Phi \in \mathbb{R}^{l\times l}$ is a stable design matrix and the nonlinear term

$$v_n = -\mathscr{K}(t,x)\frac{P_2 s(t)}{\|P_2 s(t)\|} \quad \text{if} \quad s(t) \neq 0 \tag{18}$$

The actual input u_p using fixed control allocation is

$$u_p = B_2(\rho)^T(B_2(\rho)B_2(\rho)^T)^{-1}\Big(-\hat{A}_{21}(\rho)x_1(t) - (\hat{A}_{22}(\rho) - \Phi)s(t) - \hat{B}_{c2}r(t) - \mathscr{K}(t,x)\frac{P_2 s(t)}{\|P_2 s(t)\|}\Big) \tag{19}$$

Notice that equation (19) does not contain knowledge of the faults $K(t)$ and so (19) can be implemented without the necessity of deploying an FDI unit. The control structure just described has been implemented on both an unmanned quadrotor and a piloted (full-scale) aircraft.

IV. IRIS+ Multirotor UAV

The UAV used in this work is the IRIS+ multirotor UAV developed by 3DR which comes ready to fly with a transmitter and battery included in the package

A. IRIS 3DR quadrotor Equation of Motion

For modelling purposes the states are roll angle, pitch angle, yaw angle, velocities in the x,y,z axes, roll rate, pitch rate, yaw rate while (x,y,z) represent position in the x,y,z axes. The system parameters are mass m and I_{xx}, I_{yy}, I_{zz} which represent inertia about x,y,z body axes respectively.

Here, only the roll, pitch and yaw rates will be controlled using sliding mode control scheme developed earlier (altitude will be controlled using a separate outer loop control). By linearizing the nonlinear equation of motion (see for example [23]) about hover and considering the attitude states only

$$x = \begin{bmatrix} \phi & \theta & \psi & p & q & r \end{bmatrix}^T$$

The corresponding system and input distribution matrices

$$A = \begin{bmatrix} O_{3\times3} & I_3 \\ O_{3\times3} & O_{3\times3} \end{bmatrix} \quad \text{and} \quad B = \underbrace{\begin{bmatrix} 0 \\ I_3 \end{bmatrix}}_{B_v} B_2 \tag{20}$$

where

$$B_2 = \begin{bmatrix} -c_2 bl_1 & c_2 bl_2 & c_2 bl_1 & -c_2 bl_2 \\ c_3 bl_3 & -c_3 bl_4 & c_3 bl_3 & -c_3 bl_4 \\ c_4 d & c_4 d & -c_4 d & -c_4 d \end{bmatrix} \tag{21}$$

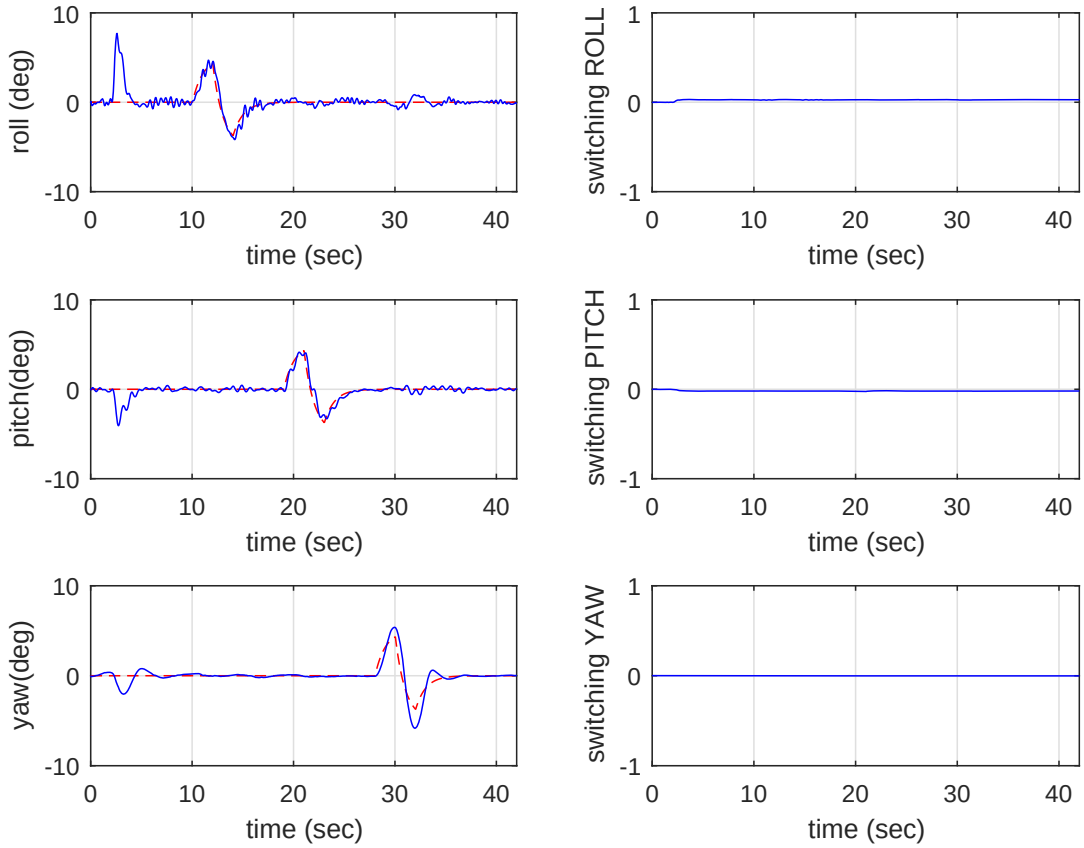


Fig. 5. Controlled states and switching functions (Faulty case)

and $c_2 = \frac{1}{I_{xx}}, c_3 = \frac{1}{I_{yy}}, c_4 = \frac{1}{I_{zz}}$. The parameters b, d, l_1, l_2, l_3, l_4 are thrust and drag factors, and moment arm lengths respectively. The controlled output distribution matrix is given by $C_c = \begin{bmatrix} I_3 & O_{3\times 3} \end{bmatrix}$ This LTI model has been used for controller synthesis and is considered as a spacial case of the LPV representation in (1). Here the inputs are

$$u(t) = \begin{bmatrix} \omega_1^2 & \omega_2^2 & \omega_3^2 & \omega_4^2 \end{bmatrix}^T \tag{22}$$

where ω_i^2 represents the square of the individual motor rotational velocities.

B. Implementation

The IRIS+ multirotor UAV is controlled using the Pixhawk flight controller as shown in Figure 4. This flight controller is open source hardware and the firmware is available as open source software [24]. The open nature of the firmware and hardware of the Pixhawk makes it popular not only among hobbyist, but also researchers.

1) Implementation and test results: Figure 5 shows good Euler angle tracking performance despite the presence of faults. The switching functions again show very little deviation from zero despite the presence of faults. Finally Figure 6 shows the effect of the faults on the PWM signals. This implementation shows the robustness of sliding mode control in the presence of faults in the quadrotor motors.

V. MuPAL-α Platform

MuPAL-α, owned and operated by the Japan Aerospace Exploration Agency (JAXA), is a multi-purpose research aircraft used for testing advanced guidance and control technologies and evaluating research on human factors. It is a Dornier Do228-202 aircraft equipped with a research FBW system [25]. Whenever necessary, the safety pilot can override the FBW system and take direct control of the flight surfaces via the Do228's mechanical control system [26] (see Fig. 7).

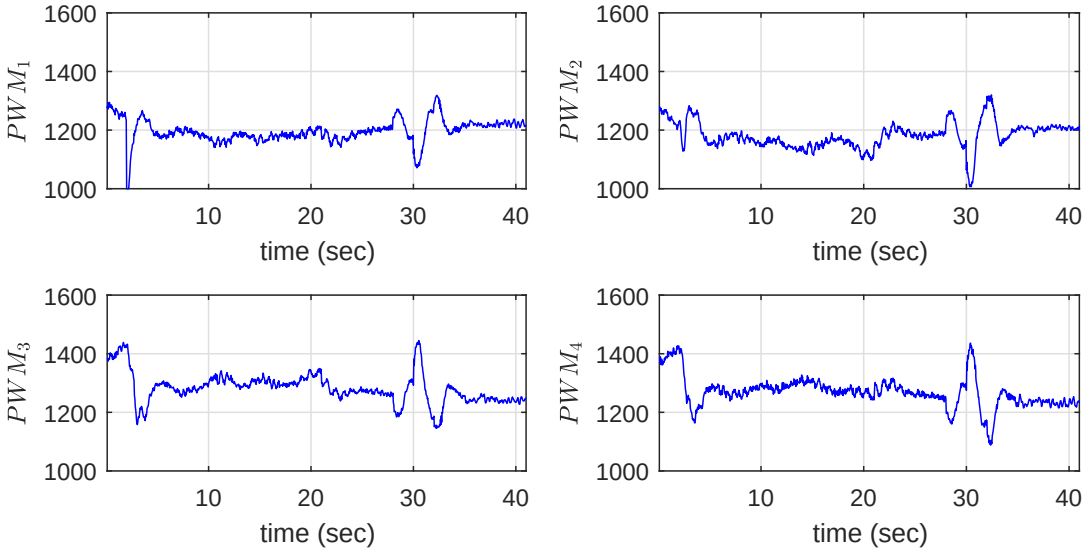


Fig. 6. PWM signals (Faulty case)

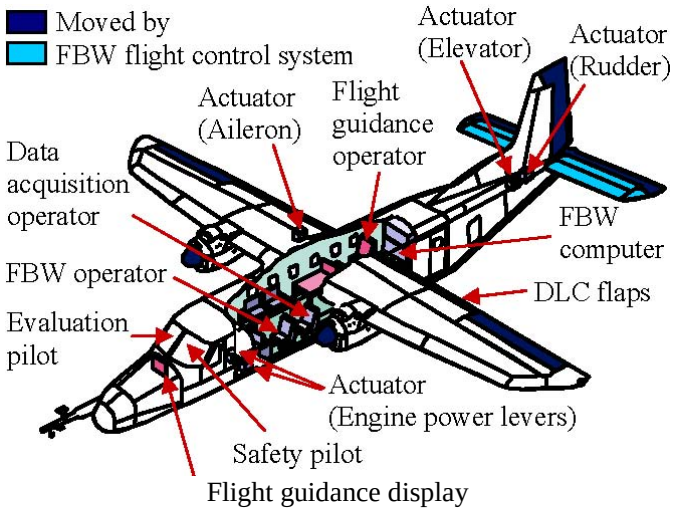


Fig. 7. Fly-by-wire configuration

A. LPV modelling

An LPV model of the lateral dynamics of MuPAL-α has been created in [27] to use as the basis for designing the control law. The scheduling parameters

$$\rho = \begin{bmatrix} v_{ias} & v_{ias}^2 \end{bmatrix} \tag{23}$$

where v_{ias} represents the indicated airspeed. This paper will focuss on lateral-directional control (because compared to the longitudinal axis it has actuator redundancy which can be exploited in terms of fault tolerant control). For modelling purposes the system states are considered to be

$$x_p = \begin{bmatrix} \phi & \beta & r & p \end{bmatrix}^T \tag{24}$$

which denote roll angle, sideslip angle, yaw rate and roll rate, respectively. The system inputs u_p are given by

$$u_p = \begin{bmatrix} \delta_{td} & \delta_a & \delta_r \end{bmatrix}^T \tag{25}$$

where δ_{td} represents differential trust, and δ_a and δ_r represent the aileron and rudder surface deflections.
Here a regional pole placement design [28] has been used to place the closed-loop poles to the left of the vertical line through -1 in the complex plane thus defining the matrix M in (15). Here $\Phi = 0.5I_2$ and $P_2 = I_2$.
Prior to any flight test, according to JAXA's protocols, any controller must satisfy stringent performance requirement during HIL testing. During the hardware in the loop (HIL) testing, the actual aircraft is employed on the ground with the FBW attached via an umbilical chord to an external PC running a model to simulate the in-fight dynamics and provide (virtual) sensor measurements. Details of this process are described in [27].

B. Flight tests

During the piloted flight tests, the faults have been emulated (at a software level), in the sense that the signal output from the control law is modified (to represent the effect of the fault) before the modified signal is sent to the actuators. During the flight tests, a steady turn with a roll angle of $\pm 20deg$ is treated as the reference command. This creates a coordinated 'S' turn manoeuvre. Figure 8 illustrates the flight test results when faults occur simultaneously on the aileron and the rudder. Here the aileron and the rudder work at 70% efficiency and 90% efficiency respectively. This is associated with $K(t) = \text{diag}(1, 0.7, 0.9)$. It must be stressed however that this information/knowledge is unknown to the controller. The same manoeuvres are now repeated in this faulty scenario. Clearly, although there exist simultaneous rudder and aileron faults, the proposed scheme can still achieve good roll and sideslip tracking performance. Figure 8(b) shows that sliding occurs and the sliding motions are maintained during flight tests. The aileron and rudder commands and their surface deflections are also shown in Fig. 8(c). The presence of faults is clearly apparent and the aileron, and the rudder cannot follow the respective demands due to the existence of faults.

VI. Conclusion

This paper considered the development of fault tolerant controllers and their application to aerospace systems: focussing on techniques and methods which have been implemented and flight tested. In particular one thread of the FTC literature has involved sliding mode controllers and in this paper a specific class of sliding mode FTC scheme has been described which incorporates control allocation to exploit over-actuation and allowing total actuator failures to be mitigated. The paper has described the implementations of these ideas on a small quadrotor UAV and also piloted flight tests on a full-scale twin-engined aircraft.

References

[1] C. Edwards, T. Lombaerts, and H. Smaili, *Fault tolerant flight control: A benchmark challenge*. Springer, 2010.

[2] P. Goupil and G. Puyou, "Airbus state-of-practice for flight control system design and certification," AIRBUS, Tech. Rep. D.1.1.1, 2013.

[3] H. Alwi, C. Edwards, and C. P. Tan, *Fault Detection and Fault-Tolerant Control Using Sliding Modes*. Springer, 2011.

[4] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annual reviews in control*, vol. 32, no. 2, pp. 229–252, 2008.

[5] R. Khan, P. Williams, P. Riseborough, A. Rao, and R. Hill, "Designing a nonlinear model predictive controller for fault tolerant flight control," *arXiv preprint arXiv:1609.01529*, 2016.

[6] "The 10 greatest emergency landings," Nov 2016. [Online]. Available: http://www.historynet.com/the-10-greatest-emergency-landings.htm

[7] H. Alwi, C. Edwards, O. Stroosma, and J. Mulder, "Evaluation of a sliding mode fault-tolerant controller for the el al incident," *Journal of guidance, control, and dynamics*, vol. 33, no. 3, pp. 677–694, 2010.

[8] F. Bateman, H. Noura, and M. Ouladsine, "Actuators fault diagnosis and tolerant control for an unmanned aerial vehicle," pp. 1061–1066, 2007.

[9] B. Dunbar, "Propulsion controlled aircraft, pca," Jan 2016. [Online]. Available: https://www.nasa.gov/centers/dryden/history/pastprojects/PCA/index.html

[10] ——, "Intelligent flight control system, ifcs," Jan 2016. [Online]. Available: https://www.nasa.gov/centers/dryden/research/IFCS/index.html

[11] ——, "Integrated resilient aircraft control," Jan 2016. [Online]. Available: https://www.nasa.gov/centers/dryden/research/FA-18_IRAC/index.html

[12] A. Chamseddine, Y. Zhang, C.-A. Rabbath, C. Fulford, and J. Apkarian, "Model reference adaptive fault tolerant control of a quadrotor uav," *AIAA Infotech@ Aerospace, St. Louis, Missouri, USA*, vol. 2931, 2011.

[13] T. Li, Y. Zhang, and B. W. Gordon, "Passive and active nonlinear fault-tolerant control of a quadrotor unmanned aerial vehicle based on the sliding mode control technique," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 227, no. 1, pp. 12–23, 2013.

[14] H. A. Izadi, Y. Zhang, and B. W. Gordon, "Fault tolerant model predictive control of quad-rotor helicopters with actuator fault estimation," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6343–6348, 2011.

[15] A.-R. Merheb, H. Noura, and F. Bateman, "A novel emergency controller for quadrotor uavs," in *2014 IEEE Conference on Control Applications (CCA)*. IEEE, 2014, pp. 747–752.

[16] ——, "Design of passive fault–tolerant controllers of a quadrotor based on sliding mode theory," *International Journal of Applied Mathematics and Computer Science*, vol. 25, no. 3, pp. 561–576, 2015.

[17] M. W. Mueller and R. D'Andrea, "Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers," pp. 45–52, 2014.

[18] M. W. Mueller and R. D Andrea, "Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles," *The International Journal of Robotics Research*, p. 0278364915596233, 2015.

[19] M. Saied, B. Lussier, I. Fantoni, C. Francis, H. Shraim, and G. Sanahuja, "Fault diagnosis and fault-tolerant control strategy for rotor failure in an octorotor," pp. 5266–5271, 2015.

[20] R. C. Avram, "Fault diagnosis and fault-tolerant control of quadrotor uavs," Ph.D. dissertation, Wright State University, 2016.

[21] T. Schneider, G. Ducard, K. Rudin, and P. Strupler, "Fault-tolerant control allocation for multirotor helicopters using parametric programming," *rN*, vol. 1, p. r2, 2012.

[22] T. A. Johansen and T. I. Fossen, "Control allocation – a survey," *Automatica*, vol. 49, pp. 1087–1103, 2013.

[23] H. Alwi and C. Edwards, "Sliding mode fault-tolerant control of an octorotor using linear parameter varying-based schemes," *IET Control Theory & Applications*, vol. 9, no. 4, pp. 618–636, 2015.

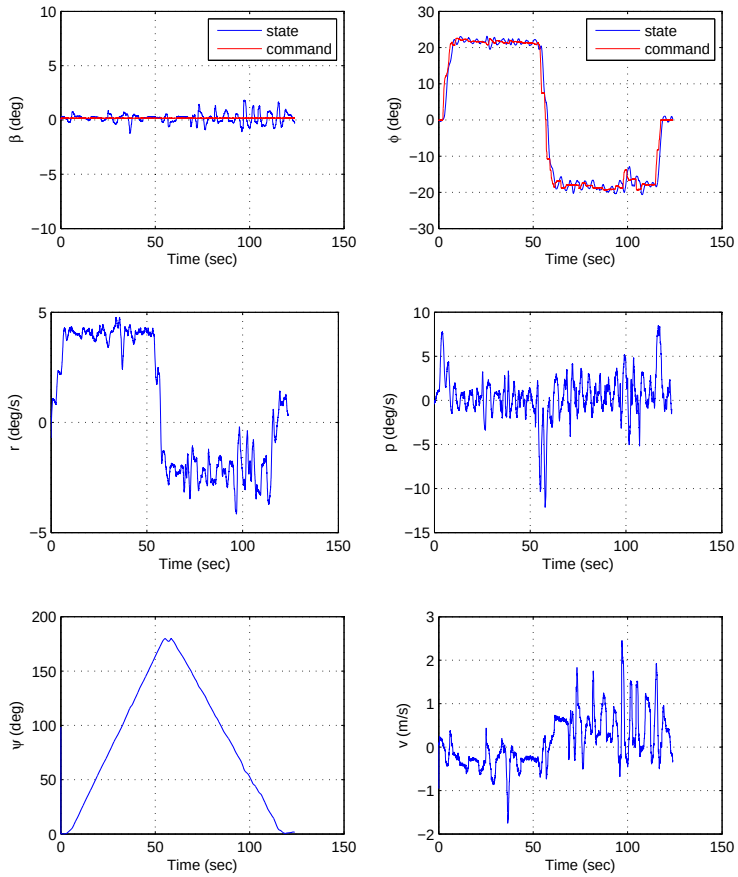
[24] M. Oborne, "Mission planner— ardupilot dev team," http://ardupilot.org/planner/, 2017, [Online; accessed 28-Feb-2017].

[25] M. Sato and A. Satoh, "Flight control experiment of multipurpose-aviation-laboratory-α in-flight simulator," *Joural OF Guidance, Control and Dynamics*, vol. 34, 2011.

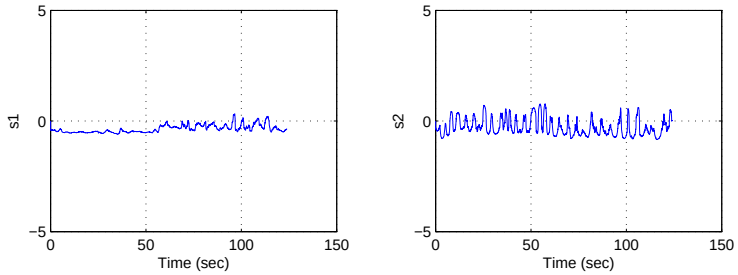
[26] K. Masui and Y. Tsukano, "Developement of a new in-flight simulator MuPAL-α," *AIAA paper 2000-4574*, Aug. 2000.

[27] L. Chen, H. Alwi, C. Edwards, and M. Sato, "Hardware-in-the-loop evaluation of an LPV sliding mode fixed control allocation scheme on the MuPAL-α research aircraft," in *IEEE Conference on Control Technology and Applications*, 2017, pp. 590–595.

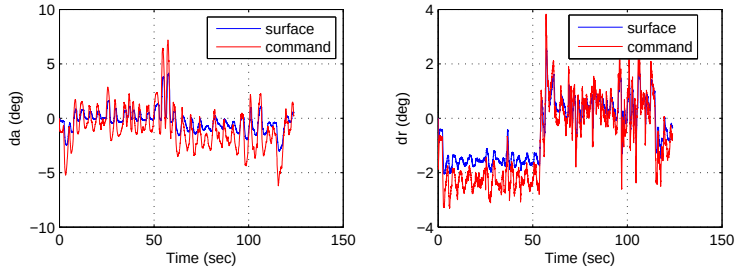
[28] M. Chilali and P. Gahinet, "H_∞ design with pole placement constraints: An LMI approach," vol. 41, no. 3, 1996.



(a) The trajectories of the system states



(b) Switching functions



(c) Commands and surface deflections

Fig. 8. Aileron and rudder faults – $K = \text{diag}(1, 0.7, 0.9)$: states, switching functions and control surface deflections (Flight test)