

Robust fault tolerant control allocation for a modern over-actuated commercial aircraft

Liam Vile | Halim Alwi | Christopher Edwards

College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, UK

Correspondence

Christopher Edwards, College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, EX44QF, UK.
Email: c.edwards@exeter.ac.uk

Abstract

This paper presents a novel form of control allocation, designed within a sliding mode framework, for the fault tolerant control of over-actuated systems. The control allocation is designed in such a way as to allow a subset of the actuators to remain inactive under nominal fault-free conditions. In the event that the active set of actuators becomes unable to provide the desired performance, an adaption process takes place which allows the inactive actuators to compensate. A computationally light gradient descent algorithm is proposed to govern the adaption which guarantees that, if possible, actuator saturation is avoided and system performance is maintained, even in the event of severe actuator faults/failures. Rigorous conditions are derived, in terms of the faults/failures, uncertainties in fault reconstruction information and the adaptive process, which ensure sliding occurs in a finite time and that the resulting motion is stable. To demonstrate the effectiveness of the control scheme, a high-fidelity blended wing body aircraft model is also proposed in this paper; this particular configuration of aircraft is nominally unstable, with poor control authority and a large amount of redundancy, making it a suitable candidate for testing reconfigurable fault tolerant control laws in the presence of input constraints.

1 | INTRODUCTION

By abstaining from using a traditional tailplane, and increasing the internal volume, the Blended Wing Body (BWB) aircraft provides an efficient alternative to the conventional “tube with wings” planform that has so far dominated the commercial aviation industry [1]. Although the resulting increase in efficiency is highly desirable, sacrifices are made elsewhere. When compared to conventional aircraft, the BWB has far less control authority and poorer stability characteristics: in both the pitch and yaw axis. Furthermore, a BWB typically features a set of highly coupled and highly redundant actuators, in the form of elevons that span the entire trailing-edge of the wing. The cumulation of these factors means that the aircraft is prone to actuator saturation: this is exacerbated in the event of actuator faults and failures. To prevent actuator saturation-and the associated performance degradation-the aircraft’s redundancy must be intelligently utilised.

A popular choice of actuator redundancy management is Control Allocation (CA). Through its use, it is possible to design

a controller based on a reduced order “virtual” control, this can then be redistributed among the actuator suite independently of the closed-loop system’s performance [2]. In its simplest form, CA can be achieved through a weighted Moore–Penrose inverse of the input distribution matrix, where the inherent weighting facilitates reconfiguration of the control distribution amongst the actuators. Several different choices of weighting matrices have been explored in the literature. The work in [3] chooses the weighting matrix to reflect the health of individual actuators, providing some tolerance to faults and failures. In [4] and [5], the weighting is chosen to help prevent problems associated with saturation. The work in [4] prioritises the use of actuators with the greatest range of motion; whilst [5] utilises the weighting to ensure all actuators saturate at the same point.

Several works have proposed more complex versions of CA for the role of preventing saturation. The work in [6] proposes an alternative to the Moore–Penrose inverse which involves the addition of an adaptive parameter-this can then be used to tune the control distribution. The method of “daisy-chaining” is presented in [7], which splits the actuator suite into different

priority groups; then in the event that one group begins to saturate, the next group in the “chain” is used. This is a popular methodology in the aerospace community. Recently, an adaptive law was proposed in [8] which allows a suitable CA law to be calculated on-line, encompassing actuator non-linearities. Many CA methods utilise on-line optimisation to ensure the control distribution satisfies some criteria. In [9], a parameter within the CA scheme is chosen through an unconstrained optimisation process to minimise any saturation in the system. Contrastingly, quadratic programming is explored in [10] and [11], where the difference in the achieved and commanded “virtual” control is minimised, using the saturation limits as constraints. In the event that redistribution of the “virtual” control alone is not sufficient to prevent saturation, [12] proposes an optimisation scheme which minimises any degradation to the system’s performance through optimising a Lyapunov function.

BWB aircraft are a natural candidate for demonstrating the efficacy of CA schemes because of their unique actuator configuration. The following works have explicitly considered the use of CA for BWBs. In [13], an analysis of the coupling between the BWB’s control surfaces is presented. A discussion on the non-linear interactions between the control surfaces, and how CA can be used to accommodate them, can be found in [11, 14]. To help alleviate the effects of coupling, the work in [15] proposes weighting the lateral and longitudinal CA modules differently, allowing different actuators to be prioritised for specific roles. In the event of actuator faults and failures, these roles are relaxed avoiding any need to reconfigure the controller. The work in [16] proposes a method of designing the CA to minimise the trim-drag of the aircraft.

The main focus of this paper however is to consider the effects of actuator faults and failures. One of the main requirements for fault tolerant control is robustness to both uncertainties and disturbances. Multiple levels of Sliding Mode Controllers (SMCs) are designed for the control of a tailless aircraft in [17] and [18]: one SMC provides a robust “virtual” control law which is then distributed through a second SMC to the actuator suite, ensuring that the “virtual” control is robustly achieved. Works which have considered the use of robust control in conjunction with CA include [19] and [20]. In [19] and [20], H_∞ control and back-stepping are used, respectively, whereas SMC is used in [3] and [21] for FTC of aerospace systems. Higher-order SMCs (HOSMCs) have also been considered for FTC. These feature continuous control laws and therefore avoid the phenomenon of “chattering,” which can be problematic for conventional SMC. In [22], an HOSMC is compared with a conventional SMC for the control of a fighter aircraft subject to different uncertainties and disturbances - including actuator faults - demonstrating the HOSMC’s ability to alleviate the “chattering” experienced by the conventional SMC scheme. An adaptive HOSMC is proposed in [23] which does not require prior knowledge of the bounds on any disturbances/uncertainties to ensure that sliding is induced in finite time. In [24], Non-Linear Dynamic Inversion is used in conjunction with an HOSMC to further reduce the effects of uncertainty in the system.

Some robust control methods have been designed to prevent saturation, without the need for redistribution. An Integral SMC scheme is presented in [25] which blends high- and low-performing controllers; an optimisation scheme chooses the best possible performing controller that keeps actuators within their limits. In a similar vein, [26] proposes a nominal reference model which is adapted on-line to change the system’s performance. The closed-loop performance and the control distribution are designed through a constrained optimisation as part of a Model Predictive Control (MPC) law in [27]. However, due to the complexity of the optimisation, MPC is computationally expensive, making it difficult to accomplish in real time. In [5] and [28], variations of Model Reference Adaptive Control (MRAC) are proposed, which, although do not prevent saturation, provide some robustness against saturation. These schemes are shown to be effective in controlling a flying wing type aircraft (similar to the BWB). In [5] a “minimal control” synthesis is presented which does not rely on the identification of model parameters, whereas in [28], the reference model is of a reduced order, based on the tracking error system.

In this paper, a novel fault tolerant SMC and an adaptive CA scheme are developed. As compared to the work in [3] (where the CA is either fixed or dependent on the actuators’ health), the proposed scheme in this paper is more elaborate and includes an additional weighting in the CA which provides further design freedom to prioritise the use of specified actuators. Furthermore, although the scheme in [3] can handle actuator faults/failures, the scheme does not consider actuator saturation: this paper proposes a scheme which allows the control effort to be redistributed in such a way as to ensure, if possible, that saturation is avoided. Another of the paper’s main contribution is the modification of the CA structure in [9], which allows the coupling in a BWB’s actuator suite to be more effectively handled. This is achieved through allowing a subset of the actuator suite to produce the entire control effort in nominal conditions. If this subset begins to saturate, then an adaptive process allows the entire actuator suite to be used. The SMC is designed to be robust to uncertainties in the fault reconstruction (which is used in the CA).

The structure of the paper is as follows: Section 2 presents an overview of the problem addressed in this paper. In Sections 3 and 4, the novel CA scheme is proposed and then appropriate SMC laws are designed and analysed. Section 5 provides the details of a high-fidelity 6 Degree of Freedom (DoF) non-linear BWB model which will form the basis of the control design, which is discussed in Section 6, and the simulation results, which are shown in Section 7. Section 8 provides some concluding remarks.

The notation used in this paper is quite standard: \mathbb{R} represents the set of all real numbers and I represents an appropriately sized identity matrix. Unless specified otherwise, $\|\cdot\|$ represents the Euclidean 2 norm. The vector $x = \text{col}(x_1, \dots, x_m)$ represents the concatenation of the components into a column vector and $X = \text{diag}(x_1, \dots, x_m)$ represents a matrix whose diagonal elements are the x_i s.

2 | PROBLEM STATEMENT

Consider an n^{th} order over-actuated Linear Time Invariant (LTI) system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + BW_f(t)u(t) \\ y(t) &= Cx(t),\end{aligned}\quad (1)$$

where the system's state, input and the controlled outputs are, respectively, denoted by $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^l$. Here, it is assumed that all the states are available. In this paper, it is also assumed that actuator redundancy is available, that is the system is over-actuated ($m > l$), meaning that there are strictly more actuators than outputs to be controlled. The corresponding system matrices are: $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{l \times n}$. The square matrix $W_f(t) = \text{diag}(w_{f_1}(t), \dots, w_{f_m}(t))$ is a time-varying matrix, which represents any actuator faults and failures present in the system. Each scalar diagonal component $w_{f_i}(t)$ signifies the health of the i^{th} actuator and as such is bounded so that $0 \leq w_{f_i}(t) \leq 1$. A healthy actuator is indicated by $w_{f_i} = 1$, whilst $w_{f_i} = 0$ represents a completely failed actuator-anywhere in between these values indicates a fault. In reality, the health of the actuators are not precisely known. For practical purposes, they can be approximated online by a Fault Detection and Isolation (FDI) system (see, e.g. the sliding mode approach proposed in [29] or the Kalman filter approach in [30]); however, this introduces uncertainty into the feedback system. Therefore, consider the following expansion of $W_f(t)$ in the state equations of (1) so that

$$\dot{x}(t) = Ax(t) + B \underbrace{(I - \Delta(t)) \hat{W}_f(t)}_{W_f(t)} u(t), \quad (2)$$

where $\hat{W}_f(t) \in \mathbb{R}^{m \times m}$ represents the approximation of $W_f(t)$ (as calculated by an FDI scheme) and $\Delta(t) \in \mathbb{R}^{m \times m}$ is a diagonal matrix denoting the uncertainty in the fault reconstruction. To capture the effects of actuator saturation, it is assumed that the individual components of the control signal in (2), $u_i(t)$, are bounded satisfying

$$u_i(t) = \begin{cases} u_{i_{\max}}, & \text{for } u_{c_i}(t) \geq u_{i_{\max}} \\ u_{c_i}(t), & \text{for } u_{i_{\min}} < u_{c_i}(t) < u_{i_{\max}} \\ u_{i_{\min}}, & \text{for } u_{c_i}(t) \leq u_{i_{\min}}, \end{cases} \quad (3)$$

where $u_{c_i}(t)$ is the signal generated by the controller and $u_i(t)$ is the achieved control signal (i.e. that which is seen by the plant). The maximum and minimum limits of the i^{th} actuator are denoted by $u_{i_{\max}}$ and $u_{i_{\min}}$, respectively; typically in aerospace systems, these are known and represent the minimum and maximum deflections of flaps or the minimum and maximum thrust produced by an engine.

3 | CONTROL ALLOCATION

Consider the system presented in (2). For the purpose of control design, assume the states can be reordered and partitioned to achieve the following form:

$$B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \quad (4)$$

where $B_2 \in \mathbb{R}^{l \times m}$ contains the majority of the control effort and $B_1 \in \mathbb{R}^{(n-l) \times m}$ is considered "small" in comparison [3]. For the sake of simplicity, it is assumed that $\|B_2\| = 1$: such a result is always possible by scaling the last l states of the system. Commonly, in the CA literature, a reduced order "virtual" control signal $v(t) \in \mathbb{R}^l$ is related to the physical control signal $u_c(t)$ through a pseudo-inverse structure. In [9, 31], the following form is proposed:

$$u_c(t) = \underbrace{W_{p1} \hat{W}_f(t) B_2^T (B_2 \hat{W}_f(t) W_{p1} \hat{W}_f(t) B_2^T)^{-1}}_{B_2^\dagger} v(t), \quad (5)$$

where $W_{p1} \in \mathbb{R}^{m \times m}$ is a fixed diagonal matrix which constitutes design freedom, thus allowing the use of certain actuators to be prioritised. Through direct evaluation, it can be verified that this choice of $u_c(t)$ satisfies

$$B_2 (I - \Delta(t)) \hat{W}_f(t) u_c(t) |_{\Delta=0} = v(t), \quad (6)$$

and therefore demonstrates that, despite actuator faults and failures, $v(t)$ is mapped to the bottom partition of (2) and directly represents the effect of the control on the last l channels of $\dot{x}(t)$: as long as there is no uncertainty in the system. Although it is possible to use W_{p1} in (5) to try to accommodate saturation limits, it is not straight-forward as the relationship between W_{p1} and $u_c(t)$ is non-linear. This approach would also remove any facility to prioritise actuators subject to other criteria, such as (in an aerospace context) reducing structural loading or increasing efficiency. As an alternative, consider the addition of an adaptive component to (5): specifically

$$u_c(t) := \underbrace{\left(B_2^\dagger(t) + \left(I - B_2^\dagger(t) B_2 \hat{W}_f(t) \right) \Theta(t) B_2^\dagger(t) \right)}_{\Phi(t)} v(t), \quad (7)$$

where $\Theta(t) = \text{diag}(\theta_1(t), \dots, \theta_m(t))$ is an adaptive parameter, B_2^\dagger is the pseudo inverse of $B_2 \hat{W}_f$ defined in (5) and $B_2^\dagger \in \mathbb{R}^{m \times l}$ is chosen as any appropriate right pseudo inverse of $B_2 \hat{W}_f$ (this will be discussed later). It can be verified that, independently of $\Theta(t)$ and $B_2^\dagger(t)$, (7) satisfies (6) and that $u_c(t)$ varies linearly with respect to $\Theta(t)$. Furthermore, when $\Theta(t) = 0$, the expression (7) reduces to (5), thus retaining the original CA form under nominal conditions. A suitable choice of $\Theta(t)$ is discussed in Section 7.2.

Remark 1. The choice of structure in (7) is inspired by the adaptive formulation in [6]. This has been modified in [9] to accommodate actuator faults and failures. In [9], the pseudo-inverse $B_2^\ddagger(t)$ is chosen such that $B_2^\ddagger(t) = B_2^\dagger(t)$ and a full-rank restriction is placed on the design of W_{p1} . In this paper, $B_2^\ddagger(t)$ is chosen as

$$B_2^\ddagger = W_{p2} \hat{W}_f(t) B_2^T (B_2 \hat{W}_f(t) W_{p2} \hat{W}_f(t) B_2^T)^{-1}, \quad (8)$$

where $W_{p2} \in \mathbb{R}^{m \times m}$ is a second priority weighting matrix. By choosing $B_2^\ddagger \neq B_2^\dagger$, additional design freedom is introduced.

Remark 2. If the restriction on W_{p1} in [9] is relaxed, (5) can be designed to operate within a subset of the actuator suite since the i^{th} actuator can be removed from the distribution of (5) by setting $W_{p1_i} = 0$. This may be desirable in cases where actuators should be reserved for their primary functions which do not fit in the controllers remit; for example, in aerospace systems the main purpose of engines are for longitudinal control but they could also be used for lateral control through differential thrust. In the event of saturation, the adaptive parameter $\Theta(t)$ redistributes the control effort through the adaptive component of (7) and this can be tuned through choice of W_{p2} in (8), to utilise a different set of actuators. In this paper, the entire actuator suite is considered for adaption by making W_{p2} full rank. This allows the CA structure in (7) to use a reduced set of actuators in nominal conditions (when $\Theta(t) = 0$) whilst allowing the entire actuator suite to be used in the event of saturation (by setting $\Theta(t) \neq 0$).

Consider a set of allowable actuator faults and failures \mathcal{W} where \mathcal{W} is such that if $\hat{W}_f(t) \in \mathcal{W}$ then $\det(B_2 \hat{W}_f(t) W_{p1} \hat{W}_f(t) B_2^T) \neq 0$ and $\det(B_2 \hat{W}_f(t) W_{p2} \hat{W}_f(t) B_2^T) \neq 0$.

Remark 3. Within the set \mathcal{W} , the norms of the pseudo-inverse structures B_2^\dagger and B_2^\ddagger , defined in (5) and (8), respectively, are bounded. This is due to the boundedness properties of pseudo-inverses described in [32]. Therefore, the following bounds

$$\gamma_0 > \|B_2^\dagger(t)\| \quad \gamma_1 > \|B_2^\ddagger(t)\|, \quad (9)$$

are always guaranteed to exist. The smallest possible values of γ_0 and γ_1 can be calculated efficiently through the use of Genetic Algorithms to search the entire set \mathcal{W} (which is compact). The gains γ_0 and γ_1 will be exploited in the stability analysis later in the paper.

4 | OPTIMISATION ALGORITHM

This section focusses on the choice of $\Theta(t)$ in (7): specifically an adaptive algorithm will be proposed suitable for running in real time.

4.1 | Cost function

For any given actuator suite, there exists a set of ‘‘virtual’’ control signals, $v(t) \in \mathcal{V}$, which is achievable subject to the actuator suite’s health and saturation limits.

Assumption: The virtual control $v(t)$ always remains in the set \mathcal{V} ; in other words, there must always exist a value of $\Theta(t) = \Theta^*$ which, when substituted into (7), ensures the saturation error

$$e(t) = u_c(t) - u(t) \quad (10)$$

is equal to zero. The problem considered here is one of perfectly allocating $v(t)$, that is, finding the unknown parameter Θ^* . Substituting from (7), into (10), for a generic choice of $\Theta(t)$, and for $\Theta(t) = \Theta^*$ yields

$$\begin{aligned} e(t) &= \left(B_2^\dagger(t) + \left(I - B_2^\dagger(t) B_2 \hat{W}_f(t) \right) \Theta(t) B_2^\ddagger(t) \right) v(t) \\ &\quad - \left(B_2^\dagger(t) + \left(I - B_2^\dagger(t) B_2 \hat{W}_f(t) \right) \Theta^* B_2^\ddagger(t) \right) v(t) \\ &= \left(I - B_2^\dagger(t) B_2 \hat{W}_f(t) \right) (\Theta(t) - \Theta^*) B_2^\ddagger(t) v(t) \\ &= \Lambda(t) (\Theta(t) - \Theta^*), \end{aligned} \quad (11)$$

where

$$\Lambda(t) = \left(I - B_2^\dagger(t) B_2 \hat{W}_f(t) \right) \text{diag} \left(B_2^\ddagger(t) v(t) \right). \quad (12)$$

To develop an algorithm to find Θ^* on-line, define the \mathbf{n}^{th} estimate of Θ^* , at a given time t , as $\Theta_{\mathbf{n}}$, and define the error function (10) when evaluated at $\Theta_{\mathbf{n}}$ as

$$e_{\mathbf{n}} = \Lambda(\Theta_{\mathbf{n}} - \Theta^*). \quad (13)$$

In (13), Λ is considered fixed and represents the value of $\Lambda(t)$ in (12) at the current instant of time. Assuming that the saturation limits in (3) are known, $e_{\mathbf{n}}$ in (13) can be evaluated on-line (without knowing Θ^*) and therefore can be used to iteratively find Θ^* . Furthermore, note that since $\text{rank}(\Lambda) \leq m - l$, $e = 0 \not\Rightarrow \theta = \theta^*$. To develop a iterative scheme to ensure $e_{\mathbf{n}} \rightarrow 0$, consider the quadratic cost function

$$f(\theta_{\mathbf{n}}) = e_{\mathbf{n}}^T e_{\mathbf{n}} = (\Theta_{\mathbf{n}} - \Theta^*)^T \Lambda^T \Lambda (\Theta_{\mathbf{n}} - \Theta^*). \quad (14)$$

It is clear that $f(\theta_{\mathbf{n}}) \geq 0$ and for any $v(t) \in \mathcal{V}$, the minimum of (14) is at $f(\cdot) = 0$. Note that $f(\theta_{\mathbf{n}}) = 0 \not\Rightarrow \theta_{\mathbf{n}} = \theta^*$ but it does ensure that $u_c(t) = u(t)$ and hence $v(t)$ is perfectly allocated.

4.2 | Gradient decent method

To find the minima of the function $f(\cdot)$, the following Gradient Descent algorithm, based on the work in [9], is adopted. Consider the following update law for selecting $\Theta_{\mathbf{n}+1}$ in terms of the

previous estimate Θ_n :

$$\Theta_{n+1} = \Theta_n + \underbrace{\text{diag}(a_n P_n)}_{\xi_n}, \quad (15)$$

where the scalar a_n represents the step length whilst $P_n \in \mathbb{R}^m$ represents the search direction [33]. The step length is chosen to minimise $f(\Theta_{n+1}) = f(\Theta_n + a_n P_n)$ at each iteration. In such algorithms, the search direction is commonly chosen as the steepest negative gradient: here this is given by

$$P_n = -\nabla f(\Theta_n) = -\left[\frac{\partial f}{\partial \Theta_{n_1}}, \frac{\partial f}{\partial \Theta_{n_2}}, \dots, \frac{\partial f}{\partial \Theta_{n_m}} \right]^T, \quad (16)$$

where Θ_{n_i} represents the i^{th} diagonal element of Θ_n . The value of $\nabla f(\Theta_n)$ can be explicitly calculated as

$$\nabla f(\theta_n) = 2\Lambda^T \Lambda(\theta_n - \theta^*) = 2\Lambda^T e_n, \quad (17)$$

from the definition of $f(\theta_n)$ in (14). Evaluating $f(\theta_{n+1})$ in (14), and utilising (15), yields

$$\begin{aligned} f(\theta_{n+1}) &= (\Theta_{n+1} - \Theta^*)^T \Lambda^T \Lambda(\Theta_{n+1} - \Theta^*) \\ &= (\Theta_n + \xi_n - \Theta^*)^T \Lambda^T \Lambda(\Theta_n + \xi_n - \Theta^*) \\ &= f(\Theta_n) + \underbrace{\xi_n^T \Lambda^T \Lambda \xi_n + 2e_n^T \Lambda \xi_n}_{\delta f_n}. \end{aligned} \quad (18)$$

It is clear that minimising $f(\theta_{n+1})$ can be achieved by minimising the term δf_n . Using the definitions of ξ_n from (15) and P_n from (16) and (17), δf_n can be expressed as

$$\begin{aligned} \delta f_n &= a_n^2 \nabla f(\theta_n)^T \Lambda^T \Lambda \nabla f(\theta_n) - 2a_n e_n^T \Lambda \nabla f(\theta_n) \\ &= a_n^2 e_n^T \Lambda \Lambda^T \Lambda \Lambda^T e_n - 4a_n e_n^T \Lambda \Lambda^T e_n \\ &= 4 \underbrace{\|\Lambda \Lambda^T e_n\|^2}_{\mathbf{a}} a_n^2 - 4 \underbrace{\|\Lambda^T e_n\|^2}_{\mathbf{b}} a_n. \end{aligned} \quad (19)$$

Thinking of the right-hand side of (19) as a quadratic scalar equation, in terms of a_n , it is clear that δf_n has a minimum at

$$a_n = \frac{\mathbf{b}}{2\mathbf{a}}. \quad (20)$$

Theorem 1. *The scalar a_n , as defined in (20) always remains bounded and so by updating Θ_{n+1} according to*

$$\Theta_{n+1} = \Theta_n + \text{diag}(a_n P_n), \quad (21)$$

where P_n are chosen as described in (17), will always result in a decrease of the cost-function (14).

Proof. The proof is similar to Proposition 1 in [9]. The algorithm is guaranteed to converge to a local minimum where $f(\Theta_n) = 0$, but does not guarantee $\Theta_n \rightarrow \Theta^*$. \square

Remark 4. In Section 7.2, the computational load of the algorithm is discussed; it is demonstrated that the algorithm is sufficiently “light” to run in real time on common, existing employed micro-processors.

5 | SLIDING MODE CONTROL DESIGN

In this section, a SMC procedure will be used to design a robust control law for $v(t)$ to control (2) through the CA structure (7).

5.1 | Sliding surface stability analysis

Assuming there is no saturation in the system ($u(t) = u_c(t)$), substituting (7) into (2) yields the following faulty “virtual” system:

$$\dot{\hat{x}}(t) = A_x \hat{x}(t) + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} (I - \Delta(t)) \hat{W}_f(t) \Phi(t) v(t). \quad (22)$$

Define a coordinate transformation $x \mapsto T_r x = \hat{x}$, where

$$T_r = \begin{bmatrix} I & -B_1 W_{\rho 1} B_2^T (B_2 W_{\rho 1} B_2^T)^{-1} \\ 0 & I \end{bmatrix}; \quad (23)$$

then in the new coordinates the system in (22) becomes

$$\dot{\hat{x}}(t) = \hat{A} \hat{x}(t) + \begin{bmatrix} B_1 B_2^N (I - \Delta(t)) \hat{W}_f(t) \Phi(t) \\ I - B_2 \Delta(t) \hat{W}_f(t) \Phi(t) \end{bmatrix} v(t), \quad (24)$$

where $\hat{A} = T_r A T_r^{-1}$ and

$$B_2^N = I - W_{\rho 1} B_2^T (B_2 W_{\rho 1} B_2^T)^{-1} B_2. \quad (25)$$

Under “ideal” operating conditions, that is, where the actuators are all healthy and perfectly monitored ($W_f(t) = \hat{W}_f(t) = I$), there is no saturation ($u(t) = u_c(t)$), and there is no adaption ($\Theta(t) = 0$); it can be verified through direct evaluation that

$$B_1 B_2^N (I - \Delta(t)) \hat{W}_f(t) \Phi(t) = 0, \quad (26)$$

which means (24) reduces to the regular form [34]:

$$\dot{\hat{x}}(t) = \underbrace{\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix}}_{\hat{A}} \hat{x}(t) + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{\hat{B}_v} v(t). \quad (27)$$

This system will be used for the basis of the control design. Assuming that the system pair (\hat{A}, \hat{B}_v) is controllable, $(\hat{A}_{11}, \hat{A}_{12})$

is also controllable and a reduced order switching function can be designed [34]. Consider the hyperplane defined by

$$\mathcal{S} = \{\hat{x}(t) \in \mathbb{R}^n : s(t) = 0\}, \quad (28)$$

where the switching function $s(t) : \mathbb{R}^n \rightarrow \mathbb{R}^l$ is given by

$$s(t) = \underbrace{\begin{bmatrix} M & I \end{bmatrix}}_s \underbrace{\begin{bmatrix} \tilde{x}_1(t) \\ \tilde{x}_2(t) \end{bmatrix}}_{\tilde{x}(t)}. \quad (29)$$

Here, the matrix $M \in \mathbb{R}^{l \times (n-l)}$ represents the design freedom: a minimal requirement being that M must be chosen so that $\tilde{A}_{11} - \tilde{A}_{12}M$ is stable. To analyse the dynamics of (24), when confined to the hyperplane \mathcal{S} , define a second coordinate transformation according to $\hat{x} \mapsto T_s \hat{x} = \tilde{x} = \text{col}(\hat{x}_1, s)$ where

$$T_s = \begin{bmatrix} I & 0 \\ M & I \end{bmatrix}. \quad (30)$$

In the new coordinate system, (24) can be rewritten as

$$\begin{bmatrix} \dot{\hat{x}}_1(t) \\ \dot{s}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{bmatrix}}_{\tilde{A}} \begin{bmatrix} \hat{x}_1(t) \\ s(t) \end{bmatrix} + \tilde{B}_v(t)v(t), \quad (31)$$

where $\tilde{A} = T_s A T_s^{-1}$, the matrix $\tilde{B}_v(t)$ is defined as

$$\tilde{B}_v(t) = \begin{bmatrix} B_1 B_2^N (I - \Delta(t)) \hat{W}_f(t) \Phi(t) \\ \Psi(t) \end{bmatrix}, \quad (32)$$

and where

$$\Psi(t) = I_l + (M B_1 B_2^N (I - \Delta(t)) - B_2 \Delta(t)) \hat{W}_f(t) \Phi(t). \quad (33)$$

In particular, from (31), $\tilde{A}_{11} = \hat{A}_{11} - \hat{A}_{12}M$ and is therefore Hurwitz stable. When confined to the hyperplane \mathcal{S} , the system is said to be in an ideal sliding motion [34]. This is characterised by $\dot{s}(t) = s(t) = 0$. Substituting these values into the bottom partition of (31) and solving for $v(t)$ yields the control effort necessary to maintain sliding. This gives the so-called equivalent control, $v_{eq}(t)$, which in this case can be calculated as

$$v_{eq}(t) = -\Psi^{-1}(t) \tilde{A}_{21} \hat{x}_1(t). \quad (34)$$

Substituting (34) for $v(t)$ in the top partition of (31) yields

$$\dot{\hat{x}}_1(t) = (\tilde{A}_{11} - B_1 B_2^N (I - \Delta(t)) \hat{W}_f(t) \Phi(t) \Psi^{-1}(t) \tilde{A}_{21}) \hat{x}_1(t), \quad (35)$$

which describes the equations of motion governing sliding. The reduced order dynamic (35) can be thought of as the negative

feedback interconnection of the non-linear feedback gain

$$\tilde{u}(t) := \underbrace{(I - \Delta(t)) \hat{W}_f(t) \Phi(t) \Psi^{-1}(t)}_{\Omega(t)} \tilde{y}(t), \quad (36)$$

and the LTI plant

$$\begin{aligned} \dot{\hat{x}}_1(t) &= \tilde{A}_{11} \hat{x}_1(t) + B_1 B_2^N \tilde{u}(t) \\ \tilde{y}(t) &= \tilde{A}_{21} \hat{x}_1(t), \end{aligned} \quad (37)$$

which has the associated transfer function

$$\tilde{G}(s) = \tilde{A}_{21} (sI - \tilde{A}_{11})^{-1} B_1 B_2^N. \quad (38)$$

For the subsequent analysis define

$$\gamma_2 = \|M B_1 B_2^N\| \quad (39)$$

$$\gamma_3 = \|\tilde{G}(s)\|_{\infty}. \quad (40)$$

Since M in (29) is chosen to stabilise \tilde{A}_{11} , it follows that $\tilde{G}(s)$ is stable and therefore γ_3 is finite.

Theorem 2. *The sliding motion presented in (35) is asymptotically stable for any fault/failure combination $\hat{W}_f(t) \in \mathcal{W}$ if the uncertainty $\Delta(t)$ satisfies*

$$\|\Delta(t)\| < \Delta_{max} \leq \frac{1 - (\gamma_2 + \gamma_3)(\gamma_0 + (1 + \gamma_1)\gamma_1 \Theta_{max})}{(1 + \gamma_2 + \gamma_3)(\gamma_0 + (1 + \gamma_1)\gamma_1 \Theta_{max})}, \quad (41)$$

where γ_0 and γ_1 are defined in (9), γ_2 is defined in (39), γ_3 is defined in (40) and Θ_{max} is a bound on the adaptive matrix such that $\Theta_{max} > \|\Theta(t)\|$, where $\Theta(t)$ is defined in (7).

Proof. For the equivalent control in (34) to be unique, the gain $\Psi(t)$ -as defined in (33) must be invertible. This is guaranteed if the following inequality holds true:

$$\left\| (M B_1 B_2^N (I - \Delta(t)) - B_2 \Delta(t)) \hat{W}_f(t) \Phi(t) \right\| < 1. \quad (42)$$

It can be shown that the CA structure $\Phi(t)$, from (7), satisfies

$$\|\Phi(t)\| < \Phi_{max} < \gamma_0 + (1 + \gamma_1)\gamma_1 \Theta_{max}. \quad (43)$$

Similarly, using (43), (39) and the fact that $\|B_2\| = 1$, it can be verified that

$$\begin{aligned} & \left\| (M B_1 B_2^N (I - \Delta(t)) - B_2 \Delta(t)) \hat{W}_f(t) \Phi(t) \right\| \\ & \leq \|M B_1 B_2^N\| (1 + \|\Delta(t)\|) \Phi_{max} + \|\Delta(t)\| \Phi_{max} \\ & \leq \gamma_2 (1 + \|\Delta(t)\|) \Phi_{max} + \|\Delta(t)\| \Phi_{max}. \end{aligned} \quad (44)$$

Rearranging (44), it can be shown that

$$\gamma_2 \Phi_{max} + (1 + \gamma_2) \Phi_{max} \hat{\Delta}_{max} < 1 \quad (45)$$

is a sufficient condition for (42) to be satisfied. Because in general $\|(1 - X)^{-1}\| \leq (1 - \|X\|)^{-1}$ [35], the following bound on $\|\Psi(t)^{-1}\|$ can be calculated using (45) and the definition of $\Psi(t)$ from (33):

$$\|\Psi(t)^{-1}\| < \frac{1}{1 - \gamma_2 \Phi_{max} - (1 + \gamma_2) \Phi_{max} \Delta_{max}}. \quad (46)$$

From the definition of the non-linear feedback gain $\Omega(t)$ in (36) and the bound on $\|\Psi(t)^{-1}\|$ from (46)

$$\begin{aligned} \|\Omega(t)\| &= \|(I - \Delta(t)) \hat{W}_f(t) \Phi(t) \Psi^{-1}(t)\| \\ &\leq \|(I - \Delta(t)) \hat{W}_f(t) \Phi(t)\| \|\Psi^{-1}(t)\| \\ &\leq \frac{(1 + \Delta_{max}) \Phi_{max}}{1 - \gamma_2 \Phi_{max} - (1 + \gamma_2) \Phi_{max} \Delta_{max}}. \end{aligned} \quad (47)$$

Finally, from the small gain theorem [36], if the inequality

$$\|\Omega(t)\| \|\tilde{G}(s)\|_\infty < 1 \quad (48)$$

is satisfied, then the closed loop system described by (37) and (36) is stable. Using the bound on $\Phi(t)$ from (47) and (40), it follows that

$$\|\Omega(t)\| \|\tilde{G}(s)\|_\infty \leq \frac{\gamma_3 (1 + \Delta_{max}) \Phi_{max}}{1 - \gamma_1 \Phi_{max} - (1 + \gamma_1) \Phi_{max} \Delta_{max}}, \quad (49)$$

and therefore, the condition that

$$\frac{\gamma_3 (1 + \Delta_{max}) \Phi_{max}}{1 - \gamma_2 \Phi_{max} - (1 + \gamma_2) \Phi_{max} \Delta_{max}} < 1 \quad (50)$$

is sufficient to guarantee stability. Rearranging (50) and substituting into (43) yields the inequality in (41). \square

5.2 | Sliding mode control laws

In this paper, the SMC law for the ‘‘virtual’’ control has been selected to have the general form

$$v(t) = v_l(t) + v_n(t), \quad (51)$$

where the linear component is given by

$$v_l(t) = -\tilde{A}_{21} \hat{x}_1(t) - \tilde{A}_{22} s(t) \quad (52)$$

and the non-linear component is given by

$$v_n(t) = -\rho(x, t) \frac{s(t)}{\|s(t)\|}. \quad (53)$$

Theorem 3. *The reachability condition is satisfied if the function $\rho(x, t)$ in (53) is chosen to satisfy*

$$\rho(t, x) \geq \frac{\Phi_{max} (\gamma_2 (1 + \Delta_{max}) + \Delta_{max}) \|v_l(t)\| + \eta}{1 - \Phi_{max} (\gamma_2 (1 + \Delta_{max}) + \Delta_{max})}, \quad (54)$$

where η is a positive design scalar and Φ_{max} is defined in (43). This ensures that a sliding motion is guaranteed to occur in a finite time and to continue to be sustained for all subsequent time.

Proof. The equation that governs the dynamics of the switching function can be obtained by substituting the ‘virtual’ control law (51) for $v(t)$ in the bottom partition of (31). This yields

$$\dot{s}(t) = \tilde{A}_{21} \tilde{x}_1(t) + \tilde{A}_{21} s(t) + \Psi(t) (v_l(t) + v_n(t)). \quad (55)$$

Using the component-wise definitions of $v(t)$ in (52) and (53), Equation (55) can be simplified to

$$\dot{s}(t) = (\Psi(t) - I) v_l(t) + \Psi(t) v_n(t). \quad (56)$$

From the definition of $\Psi(t)$ in (33) and using the fact that $\|B_2\| = 1$, it can be seen that

$$\begin{aligned} s^T \dot{s} &= -\rho \|s\| + s^T (MB_1 B_2^N (I - \Delta) - B_2 \hat{\Delta}) \hat{W}_f \Phi v_n \\ &\quad + s^T (MB_1 B_2^N (I - \Delta) - B_2 \Delta) \hat{W}_f \Phi v_l \\ &\leq \|s\| (\rho \|MB_1 B_2^N (I - \Delta) - B_2 \Delta\| \Phi_{max} - \rho \\ &\quad + \|MB_1 B_2^N (I - \Delta) - B_2 \Delta\| \|v_l(t)\| \Phi_{max}) \\ &\leq \|s\| (\rho (\|MB_1 B_2^N\| (1 + \|\Delta\|) + \|\Delta\|) \Phi_{max} - \rho \\ &\quad + (\|MB_1 B_2^N\| (1 + \|\Delta\|) + \|\Delta\|) \|v_l(t)\| \Phi_{max}). \end{aligned} \quad (57)$$

Using the definitions of γ_2 from (39) and $\hat{\Delta}_{max}$ from (41), Equation (57) can be further simplified to

$$\begin{aligned} s^T \dot{s} &\leq \|s\| (-\rho (1 - \gamma_2 \Phi_{max} (1 + \Delta_{max}) - \Phi_{max} \Delta_{max}) \\ &\quad + (\Phi_{max} \gamma_2 (1 + \Delta_{max}) + \Phi_{max} \Delta_{max}) \|v_l(t)\|), \end{aligned} \quad (58)$$

finally through substituting (54) into (58), it can be verified that this choice of $\rho(t, x)$ satisfies the reachability condition [34]

$$s^T \dot{s} \leq -\eta \|s\|. \quad (59)$$

This guarantees s is forced to zero in a finite time and remains zero for all subsequent time i.e there exists a time $t = t_s$ at which $s(t_s) = 0$, and for all time $t > t_s$, $s(t) = 0$. \square

TABLE 1 Summary of aerofoil sectional properties

Section	Span (m)	$\frac{1}{4}$ Chord sweep ($^\circ$)	Dihedral ($^\circ$)	Taper	Inner aerofoil	Outer aerofoil	Inner twist ($^\circ$)	Outer twist ($^\circ$)
1	10	57	0.0	0.58	NACA 25116	NACA 0016	-2.00	-1.06
2	3.0	57	2.0	0.78	NACA 0016	NACA 0016	-1.06	-0.52
3	4.5	20	1.5	0.65	NACA 0016	SC2 0612	-0.52	-0.01
4	6.0	31	1.5	0.66	SC2 0612	SC2 0610	-0.01	0.82
5	10	35	3.0	0.63	SC2 0610	SC2 0610	0.82	-0.40
6	5.3	35	3.0	0.69	SC2 0610	SC2 0610	-0.40	-2.15
R	5.0	28	79	0.60	NACA 0012	NACA 0012	0.00	0.00

6 | NON-LINEAR BLENDED WING BODY AIRCRAFT MODEL

This section outlines the development of a non-linear 6 Degree of Freedom (6DoF) BWB model. In [37], several BWB configurations were presented and analysed, which formed the basis of the work in [38]; however, both of these endeavours consider the case of static aerodynamic coefficients. In reality, these quantities are dependent on the aircraft's state. To ensure that the BWB's dynamics are properly captured, the aircraft geometry presented in [37, 38] has been reproduced and analysed in *TORNADO*-a Vortex Lattice Method (VLM) [39]. The results from the analysis have been used to generate lookup tables for the aerodynamic coefficients.

6.1 | Aircraft geometry

In Table 1, details are provided of seven distinct aerofoil sections. The root aerofoil is represented by section 1 and the winglets are represented by section R. Interpolating between these sections, along the span, and mirroring at the root provides a complete description of the aircraft's geometry. For the centre-wing (wing section 1), a reflex camber aerofoil (NACA-25116) is chosen to provide a positive (nose-up) pitching moment, aiding trimming of the aircraft. The outer-wing (wing sections 4–6) features supercritical aerofoils (SC2-0610 and SC2-0612) to provide better high-speed performance. The inner and outer wings are blended together through a symmetrical aerofoil (NACA-0016) in the centre (wing sections 2 and 3). The twist distribution is designed to reduce the angle of attack at the wing-tips, reducing the chances of “tip-stall,” and at the centre-wing, to better shape the span-wise lift distribution. The aircraft's positive dihedral helps to provide some stability in the roll axis. Additional geometric and inertial properties

TABLE 2 BWB geometric properties

Parameter	Notation	Value	Units
Aircraft mass	\bar{m}	371,280	Kg
Inertia about X_b axis	J_{xx}	47.03×10^6	Kgm^2
Inertia about Y_b axis	J_{yy}	25.06×10^6	Kgm^2
Inertia about Z_b axis	J_{zz}	99.73×10^6	Kgm^2
Wing area	S_{ref}	1497.03	m^2
Mean chord	\bar{c}	28.26	m
Root chord	c_{root}	50.00	m
Wing span	\bar{b}	87.50	m
Chordwise position of:			
Centre of gravity	CoG	29.4	m
Centre of pressure	CoP	31.6	m

of the aircraft are shown in Table 2. Note that the Centre of Gravity (CoG), Centre of Pressure (CoP) and the aircraft's mass are considered constant in this paper (although in reality these will be dependent on the aircraft's payload and the flying conditions).

6.2 | Equations of motion

Two different reference frames are commonly used to describe the motion of an aircraft. The earth reference frame (X_e, Y_e, Z_e) is used to describe the position of an aircraft's CoG with respect to a fixed point on Earth, whilst the body reference frame

(X_b, Y_b, Z_b) describes the orientation and motion of the aircraft with respect to its CoG. Within the body reference frame, the following standard equations of motion [40, 41] can be derived:

$$\begin{aligned}
\dot{u}_b &= u_b r - w_b q - g \sin \phi + F_x / \bar{m} \\
\dot{v}_b &= w_b p - u_b r + g \cos \theta \sin \phi + F_y / \bar{m} \\
\dot{w}_b &= u_b q - v_b p + g \cos \theta \cos \phi + F_z / \bar{m} \\
\dot{p} &= r q (J_{yy} - J_{zz}) / J_{xx} + M_l / J_{xx} \\
\dot{q} &= p r (J_{zz} - J_{xx}) / J_{yy} + M_m / J_{yy} \\
\dot{r} &= q p (J_{xx} - J_{yy}) / J_{zz} + M_n / J_{zz} \\
\dot{\phi} &= p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\
\dot{\theta} &= q \cos \phi - r \sin \phi \\
\dot{\psi} &= q \sin \phi \sec \theta + r \cos \phi \sec \theta,
\end{aligned} \tag{60}$$

where u_b, v_b and w_b represent the aircraft's velocities (ms^{-1}) along the X_b, Y_b and Z_b axes, respectively. The Euler angles ϕ, θ and ψ represent the aircraft's rotational position (rad) around the X_b, Y_b and Z_b axes, respectively¹—these are widely known as the roll, pitch and yaw angles. The rate of change of these angles ($rad s^{-1}$) are denoted by p, q and r . The aircraft's mass is denoted by \bar{m} , the moments of inertia are denoted by J_{ii} , where ii is a pairing of axis (in the body reference frame), and g represents the acceleration due to gravity (ms^{-2}). The body forces (F_X, F_Y, F_Z) (N) and moments (M_l, M_m, M_n) (Nm^{-1}) are related to the aerodynamic coefficients $C_i(x, u)$ through

$$\begin{bmatrix} F_X \\ F_Y \\ F_Z \end{bmatrix} = \begin{bmatrix} C_X(x, u) \\ C_Y(x, u) \\ C_Z(x, u) \end{bmatrix} \bar{q} S_{ref} + DCM_{eb}^{Z_e} \bar{m} g + F_{eng}, \tag{61}$$

$$\begin{bmatrix} M_l \\ M_m \\ M_n \end{bmatrix} = \begin{bmatrix} C_l(x, u) \bar{b} \\ C_m(x, u) \bar{c} \\ C_n(x, u) \bar{b} \end{bmatrix} \bar{q} S_{ref} + M_{eng}, \tag{62}$$

where F_{eng} and M_{eng} represent the forces and moments produced by the aircraft's engines. The matrix $DCM_{eb}^{Z_e}$ is a partial directional cosine matrix which transforms a force in the Z_e direction (in this case the aircraft's weight) into the body reference frame: this is given by

$$DCM_{eb}^{Z_e} = \begin{bmatrix} \cos(\phi) \sin(\theta) \cos(\psi) + \sin(\phi) \sin(\psi) \\ \cos(\phi) \sin(\theta) \sin(\psi) - \sin(\phi) \cos(\psi) \\ \cos(\phi) \cos(\theta) \end{bmatrix}. \tag{63}$$

¹ In this paper, rotational angles will follow the "right hand rule" and therefore positive will be defined as a clockwise rotation around an axis when viewed from the origin.

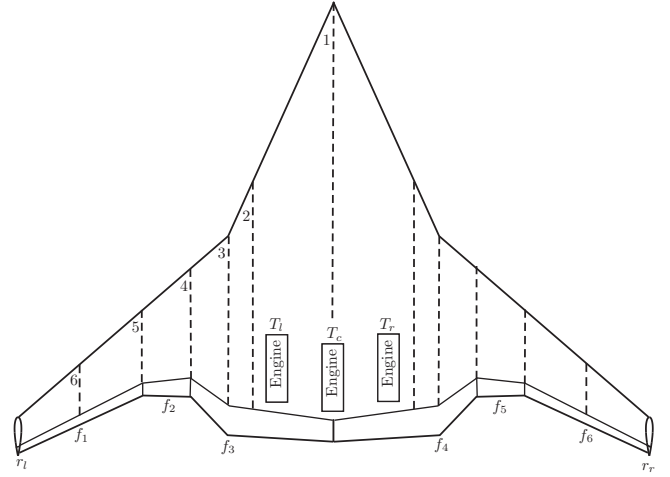


FIGURE 1 BWB layout (with aerofoil sections from Table 1 labeled)

The geometric parameters \bar{b} and \bar{c} represent the aircraft's wingspan and the mean aerodynamic chord, respectively, while the dynamic pressure (ρa) is denoted by \bar{q} and can be calculated through

$$\bar{q} = \frac{1}{2} \bar{\rho} V_i^2, \tag{64}$$

where $\bar{\rho}$ represents air density (Kgm^{-3}) and

$$V_i = \sqrt{u_b^2 + v_b^2 + w_b^2} \tag{65}$$

represents the true airspeed of the aircraft (ms^{-1}).

6.3 | Actuator suite

The BWB is equipped with 10 elevons spanning the trailing edge of the wing. These are aggregated into six control surfaces (f_1, \dots, f_6) (rad). The aircraft also features two winglet rudders (r_l, r_r) (rad) and three engines—one centered and the other two symmetrically offset (T_l, T_c, T_r) (N). The actuator suite is shown in Figure 1. The sign convention used here describes a positive deflection of the elevons as upward (negative in the Z_b axis), whilst for the rudders a positive deflection is defined as leftward (negative in the Y_b axis).

The BWB's three engines are positioned relative to the aircraft's CoG. Each engine is 0.5 m higher than the CoG and inclined upward by 2.5° . The outboard engines are displaced in the Y_b axis by 5 m and canted inward by 2° . The forces and moments produced by the engines on the aircraft, as used in Equations (61) and (62), can be approximated as:

$$F_{eng} = \begin{bmatrix} T_l + T_c + T_r \\ \sin(2^\circ) \cdot (T_l - T_r) \\ -\sin(2.5^\circ) \cdot (T_l + T_c + T_r) \end{bmatrix}, \tag{66}$$

and

$$M_{eng} = \begin{bmatrix} 5 \sin(2.5^\circ) \cdot (T_l - T_r) \\ (T_l + T_c + T_r)/2 \\ 5 \cdot (T_l - T_r) \end{bmatrix}, \quad (67)$$

where T_l , T_c and T_r represent the thrust (N) produced by the engines as labeled in Figure 1.

6.4 | Aerodynamic coefficients

The aerodynamic coefficients C_i , used in Equations (61) and (62), can be described through the equations

$$\begin{aligned} C_X &= C_{X_0} + C_{X_\alpha} \alpha + \frac{\bar{c}}{V_t} C_{X_q} q + C_{X_\delta} \\ C_Y &= C_{Y_\beta} \beta + \frac{\bar{b}}{V_t} (C_{Y_r} r + C_{Y_p} p) + C_{Y_\delta} \\ C_Z &= C_{Z_0} + C_{Z_\alpha} \alpha + \frac{\bar{c}}{V_t} C_{Z_q} q + C_{Z_\delta} \\ C_l &= C_{l_\beta} \beta + \frac{\bar{b}}{V_t} (C_{l_r} r + C_{l_p} p) + C_{l_\delta} \\ C_m &= C_{m_0} + C_{m_\alpha} \alpha + \frac{\bar{c}}{V_t} C_{m_q} q + C_{m_\delta} \\ C_n &= C_{n_\beta} \beta + \frac{\bar{b}}{V_t} (C_{n_r} r + C_{n_p} p) + C_{n_\delta} \end{aligned} \quad (68)$$

where α is the Angle of Attack (AoA) (*rad*), β is the side-slip angle (*rad*) and V_t denotes the true airspeed of the aircraft (ms^{-1}). The coefficients in (68) are actually functions of the system state such that $C_{ij} = C_{ij}(x)$; this notation has been simplified in (68) to improve readability. The coefficients C_{i_δ} denote the changes of C_i caused by the aerodynamics control surfaces. The total effect of the control surfaces on the aircraft's aerodynamic derivatives (68) can be expressed as

$$C_{i_\delta}(x) = \sum_{m=1}^6 C_{i_{\delta_m}}(x) \cdot \delta_m, \quad (69)$$

where $C_{i_{\delta_m}}$ represents the derivatives of the coefficient C_i with respect to the m^{th} control surfaces' perturbation from the neutral point, denoted by δ_m .

The BWB aircraft, as described in §6.1, was rendered in the VLM program *TORNADO* [39] to study its aerodynamic properties. From experimentation, it was found that the most significant changes in the aerodynamic coefficients were due to changes in V_t and α (since m and CoG are assumed constant); therefore, it was decided to base the lookup tables on these parameters. Using a grid of 24 different values of V_t (ranging from $50ms^{-1}$ to $280ms^{-1}$) and 61 different values of α

(ranging from -5° to 10°), a total of 1464 different operating conditions were analysed using the VLM. Results from the analysis were then transformed into lookup tables to approximate the functions $C_{ij}(x)$ as used in (68). The data from some of these lookup tables are shown in Figure 2, in which it can be seen the aerodynamic coefficients vary with the aircraft's state (as opposed to remaining constant). Out of the 48 total lookup tables, those shown in Figure 2 were considered the best cross-section of the entire set, both in terms of their individual shapes and the type of coefficients they represent.

6.5 | Linear model

$$\begin{aligned} A &= \begin{bmatrix} 0 & 0 & 1 & 0.074 \\ 0.048 & -0.023 & 0.087 & -1.004 \\ 0 & -1.806 & -2.803 & 0.504 \\ 0 & 0.384 & -0.171 & -0.046 \end{bmatrix} \\ B &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.005 & -0.008 & -0.001 & -0.001 & 0.001 & 0.001 & 0.008 & 0.005 & 0 & 0 \\ 0.266 & -2.914 & -1.513 & -1.761 & 1.761 & 1.513 & 2.914 & 0.266 & 0.005 & -0.005 \\ -0.153 & 0.222 & 0.050 & 0.069 & -0.069 & -0.050 & -0.222 & -0.153 & 0.050 & -0.050 \end{bmatrix} \end{aligned} \quad (70)$$

For the purposes of control design, the non-linear model described in the previous sections was linearised using the *MATLAB* functions: *linmod* and *operspec*. The linearisation point was chosen at a typical cruise condition with an airspeed of $200ms^{-1}$ at an altitude of 3000m. The linearised model showed very little cross-coupling between the lateral and longitudinal dynamics: this allows the problems of lateral and longitudinal control to be addressed independently. In this paper, only lateral control will be considered. The lateral dynamics can be described in the form of (1) where the A and B matrices are shown in (70). The (controlled) output matrix is given by

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \quad (71)$$

The lateral system's states $x(t)$, controlled outputs $y(t)$, and inputs $u(t)$ are given by

$$x(t) = [\phi \ \beta \ p \ r]^T \quad y(t) = [\phi \ \beta]^T, \quad (72)$$

$$u(t) = [\delta r_l \ \delta f_1 \ \delta f_2 \ \delta f_3 \ \delta f_4 \ \delta f_5 \ \delta f_6 \ \delta r_r \ \delta T_l \ \delta T_r]^T, \quad (73)$$

where the elements of $u(t)$ represent perturbations from the trim conditions (not the neutral point) of the corresponding controls in Figure 1. The trim value of $u(t)$, generated through

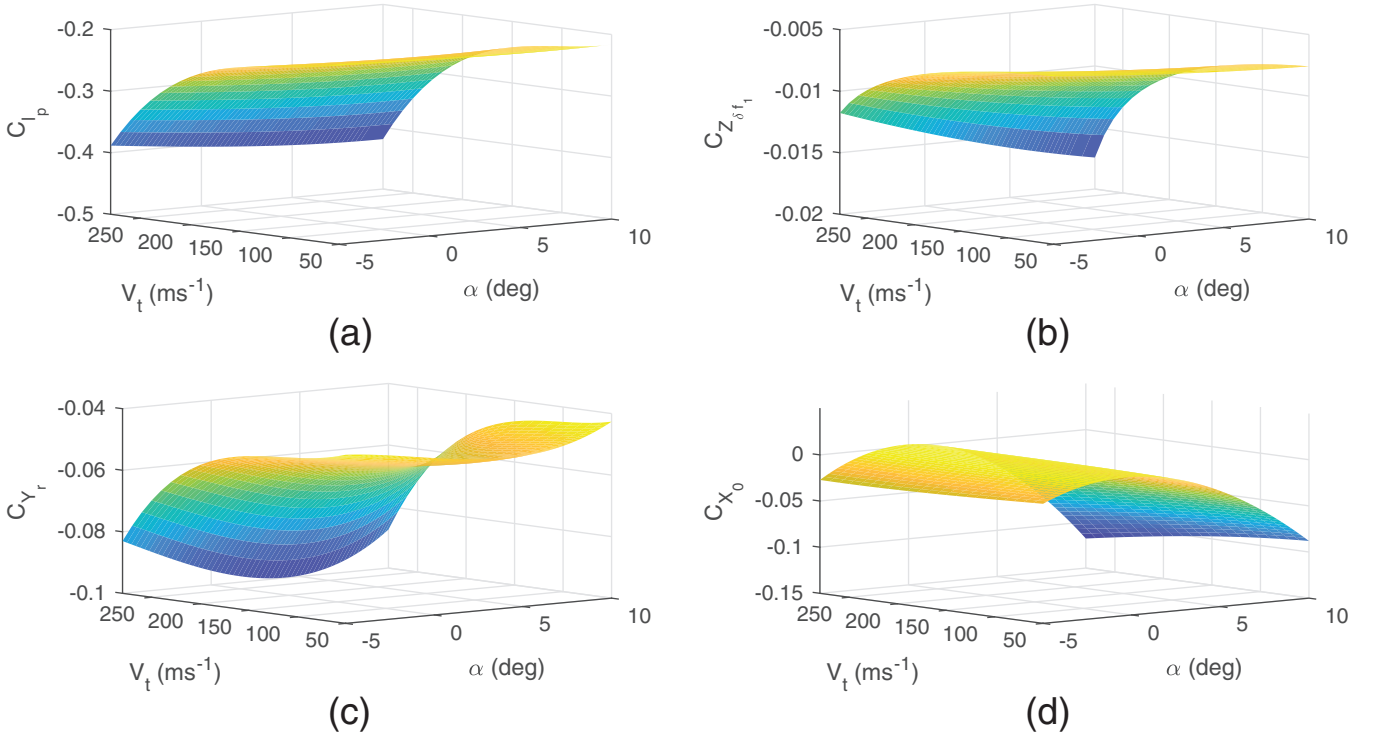


FIGURE 2 Aerodynamic coefficient plots

the linearisation process, is given by

$$u_{lim} = \begin{bmatrix} 0 & -0.11 & -0.11 & -0.11 & -0.11 & -0.11 & -0.11 & -0.11 & 0 & 0.16 & 0.16 \end{bmatrix}^T.$$

The units of $u(t)$ are *rad* for the aerodynamic controls and *MN* for the engines. Note that since the central engine was found to have a negligible effect on the lateral dynamics, it is omitted from $u(t)$. The saturation limits u_{max} and u_{min} , in (3), are set as

$$u_{max} = \begin{bmatrix} 0.65 & 0.52 & 0.44 & 0.26 & 0.26 & 0.44 & 0.52 & 0.65 & 0.17 & 0.17 \end{bmatrix}^T$$

$$u_{min} = -\begin{bmatrix} 0.65 & 0.52 & 0.35 & 0.26 & 0.26 & 0.35 & 0.52 & 0.65 & 0.16 & 0.16 \end{bmatrix}^T.$$

7 | CONTROL DESIGN

In this section, the specifics of a controller for the lateral control of a non-linear BWB are discussed. The controller will be based upon applying the theory discussed in Sections 3 and 5 to the linear BWB model from Section 6.5.

7.1 | Control allocation design

The CA will be based on the distribution structure in (7) where the individual components are defined in (5) and (8). For the nominal component of the CA the associated priority matrix is selected as

$$W_{p1} = \text{diag}(1, 0, 0, 1, 1, 0, 0, 1, 0, 0), \quad (74)$$

meaning that only the rudders and the two central elevons (r_l , f_3 , f_4 and r_r) are used under nominal conditions, leaving the rest of the controls to focus on longitudinal control. The priority weighting associated with the adaptive component is selected as

$$W_{p2} = \text{diag}\left(\frac{(u_{max} - u_{min}) \odot (u_{max} - u_{min})}{\min(u_{max} - u_{min})}\right), \quad (75)$$

which prioritises the use of actuators with the greatest range of motion and allows the adaptive component to access the entire actuator suite in the event of the nominal set saturating.

7.2 | On-line optimisation

The optimisation scheme proposed in Section 4.2 is sufficiently computationally light to be executed in real time. In the following simulations, the optimisation scheme is run every 0.01 s for

a total of 100 iterations. The continuous time controller is thus essentially implemented in discrete time at a sample rate of 0.01 s. A “forgetting factor” is implemented on $\Theta(t)$ which allows the adaptive parameter to return to zero after a period of saturation has ended: thus returning to the nominal CA law.

To demonstrate the efficiency of this algorithm, the number of floating point operations required to run it was counted using [42]. Consider the case where $m = 10$ and $l = 2$ (i.e. the BWB example studied in this paper). Since at each individual controller update time-step Λ is fixed, it therefore only needs to be calculated once per time-step. This requires a total of approximately 13,000 floating point operations. Calculating the value of Θ_{n+1} , through (15), requires 1400 operations per iteration. Assuming that the optimisation process is run every 0.01 s for 100 iterations, a total of 15×10^6 operations are required each second—this is representative of the simulations presented in this paper. An Intel(R) Core(TM) i7 - 9700K processor running at 3.6 GHz is capable of 44GFLOPs (10^9 floating point operations per second), and therefore in this particular example, the optimisation uses less than 0.04% of the processor’s capacity. The algorithm is also suitable to run on smaller processors. The Raspberry Pi is capable of 0.132GFLOPs [43] and the ARM-core Cortex-A9 (a common mobile phone processor) is capable of 0.372GFLOPs [44]; therefore, in this example, the optimisation algorithm uses 11.9% and 4.2% of the respective processors’ capacity. Note that although this optimisation algorithm is implementable in real time, it still requires a large amount of computation compared to other calculations undertaken to update the control signal: for example, calculating $u(t)$ for a given value of $\Theta(t)$ and $x(t)$, through (51) and (7), requires only 334 floating point operations per time-step (3.34×10^{-6} GFLOPs).

7.3 | A practical control law with tracking

Here, reference tracking capabilities are introduced by augmenting system (2) with integral action states [34]. This guarantees perfect tracking of fixed reference signals at steady state. The augmented system state is given by $\bar{x}(t) = \text{col}(x_r(t), x(t))$ where

$$\dot{x}_r(t) = r(t) - y(t). \quad (76)$$

In (76), $r(t)$ is a (continuous) external reference signal [34] and $y(t) = Cx(t)$ represents the controlled outputs, as defined in (1). The overall augmented system is

$$\dot{\bar{x}}(t) = \bar{A}\bar{x}(t) + \bar{B}W_f(t)u(t) + B_r r(t), \quad (77)$$

where the system matrices are

$$\bar{A} = \begin{bmatrix} 0 & -C \\ 0 & A \end{bmatrix} \quad \bar{B} = \begin{bmatrix} 0 \\ B \end{bmatrix} \quad B_r = \begin{bmatrix} I \\ 0 \end{bmatrix}. \quad (78)$$

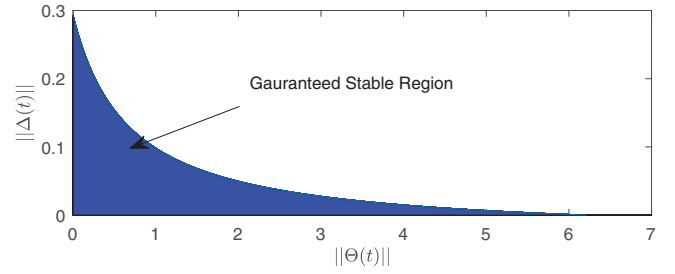


FIGURE 3 Stability region plot

If the control signal $u(t)$ is chosen to stabilise (77), then perfect tracking of a fixed reference signal is achieved [34]. It can be shown by direct evaluation that if (A, B, C) has no transmission zeros at the origin and (A, B) is controllable, the augmented system (\bar{A}, \bar{B}) is controllable. A small “feed-forward” modification to $v_l(t)$ in (52) is required to include the reference signal: specifically

$$v_l(t) = -\tilde{A}_{21}\hat{x}_1(t) - \tilde{A}_{22}s(t) - MB_r r(t). \quad (79)$$

To avoid the phenomenon of “chattering” [34], the following continuous approximation of (53) is used in the simulations:

$$v_n = -\rho(t, x) \frac{s(t)}{\|s(t)\| + \delta}, \quad (80)$$

where δ is a small positive design scalar. In this paper, $\delta = 0.0025$ and a fixed value of ρ is chosen where $\rho = 0.2$. These values were selected through intensive simulation testing. Firstly, ρ is selected large enough to satisfy (54), and therefore ensure sliding occurs (but not so unnecessarily large as to cause excessive “chattering”). Finally, δ is chosen to be as small as possible to prevent “chattering” whilst maintaining a reasonable approximation of an “ideal” sliding motion.

7.4 | Sliding surface stability analysis

The sliding surface was designed using the LMI approach outlined in [45], which aims to solve an LQR-like problem whilst minimising the norm γ_3 . The weighting matrix used for this design is given as

$$Q = \text{diag}(10, 10, 1, 2, 5, 5), \quad (81)$$

where the first two entries (those associated with the integral action states) are weighted heavily to provide good tracking performance.

Using genetic algorithms, suitable values of γ_0 and γ_1 can be found as 3.015 and 1.705, respectively. By direct evaluation, the values of γ_2 and γ_3 can be calculated as 0.0149 and 0.0176, respectively. Substituting these values into the equality (41) defines the region of guaranteed stability, as shown in Figure 3. The figure shows that the system is most robust to uncertainty

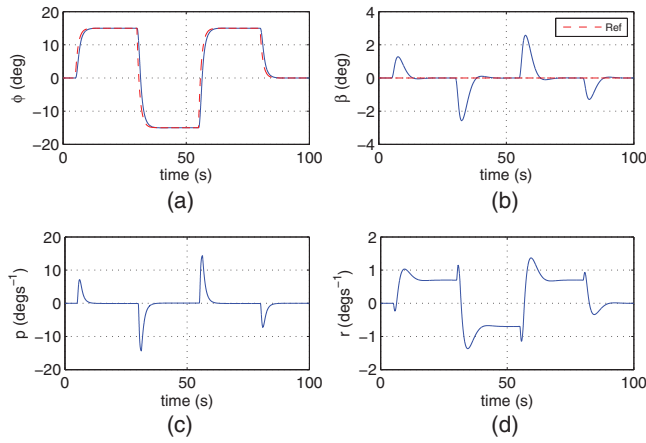


FIGURE 4 States: nominal case

when $\Theta(t) = 0$, at this point stability is guaranteed as long as $\|\Delta(t)\| < 0.29$. As $\|\Theta(t)\|$ increases, the allowable value of $\|\Delta(t)\|$ decreases exponentially and reaches the value of zero when $\|\Theta(t)\| = 6.01$.

8 | SIMULATION RESULTS

The following simulation results demonstrate the effectiveness of the proposed control scheme. The simulations consist of the non-linear BWB (from §6) undertaking a “wing rocking” manoeuvre in three different scenarios: nominal (with perfect actuator health and $\Theta(t) = 0$), faulty without the adaptive CA element ($\Theta(t) = 0$) and in a faulty scenario with the adaptive CA. Three different cases will be considered in this section to demonstrate the efficacy of the proposed scheme in maintaining system performance when subjected to actuator faults/failures (especially when dealing with saturation): specifically this paper considers

- A nominal fault free ($W_f(t) = I, \Delta(t) = 0$) situation.
- A fault/failure scenario ($W_f(t) \neq I, \Delta(t) \neq 0$) with the fixed, non-optimised CA from [3]-this equivalent to (7) when $\Theta(t) = 0$ throughout the simulation.
- A fault/failure scenario ($W_f(t) \neq I, \Delta(t) \neq 0$) with the proposed optimised CA. Here, $\Theta(t)$ is not fixed and is chosen on-line using the gradient descent algorithm.

8.1 | Nominal case

The results for the nominal case are achieved by setting $W_f(t) = \hat{W}_f(t) = I$ and the adaptive parameter $\Theta(t)$ in (7) is fixed at zero. In Figures 4 and 5, nominal performance is shown. Note that with perfect actuator health, the nominal CA in Figure 5 does not violate any of the actuator’s saturation limits (dashed black lines).

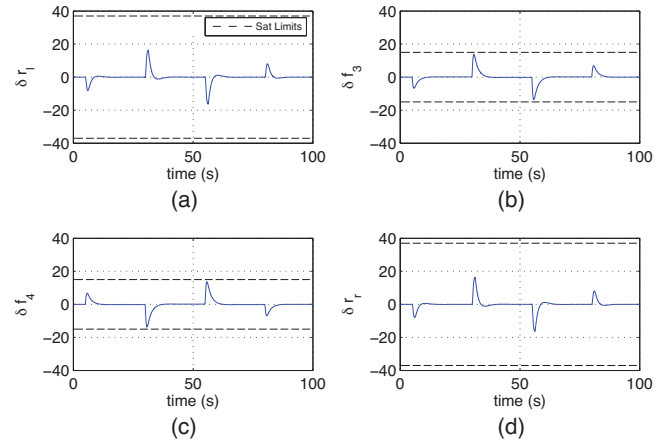


FIGURE 5 Control signals (deg): nominal case

8.2 | Faulty case—fixed CA

In this scenario, a fixed actuator fault and failure is considered where

$$W_f = \text{diag}(0, 1, 0.1, 1, 0, 0.1, 0, 1, 1, 1). \quad (82)$$

This includes complete failures of r_7 and f_4 which are part of the subset used by the nominal CA in (5). Setting $w_{f_i} = 0$ (from the start of the simulation) essentially simulates a total loss effectiveness whereby the i^{th} actuator is non-responsive to any command signal and remains locked at the trim point, whilst setting $0 < w_{f_i} < 1$ simulates a “loss of effectiveness” fault. To simulate the uncertainty introduced by an FDI scheme, the diagonal components of $\Delta(t)$ are set as sine waves with: random magnitudes (between 0 and 0.1), random frequencies (between 3rads^{-1} and 0rads^{-1}) and random phase angles. The diagonal components of $\hat{W}_f(t)$ are bounded such that $0 \leq \hat{w}_{f_i}(t) \leq 1$.

As in the nominal case $\Theta(t)$ is fixed at zero, effectively reducing the controller to the previous SMC scheme with fixed CA [3, 15]. Once faults and failures occur in the nominal actuator set, the performance deteriorates and the manoeuvre induces saturation. This can be seen in Figure 6 and is most clear in the roll angle (ϕ) response shown in Figure 6a: for example, in response to the first negative reference signal (at 30 s), the rise time is increased from 4.28 to 4.71 s and there is a 66% overshoot (compared to 0% in the nominal case). There is also some clear degradation in the yaw (β) response shown in Figure 6b. There is a 45% increase in the overshoot between 32 – 34 s and 57 – 59 s; there is also a 2° increase in overshoot between 38 – 44 s and 62 – 70 s (compared to 0.1° in the nominal case, giving a 2000% increase). The reason for this can be seen in Figure 7 where the remaining healthy elevon (f_3 shown in Figure 7b) saturates in an attempt to compensate for the failed actuator (f_4 shown in Figure 7b). As f_3 enters saturation, the controller attempts to compensate again, by increasing the signal to f_3 , pushing the system deeper into saturation.

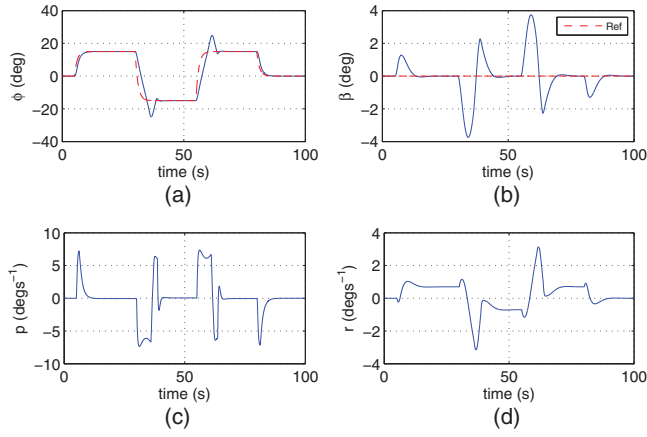


FIGURE 6 States: faulty case fixed CA

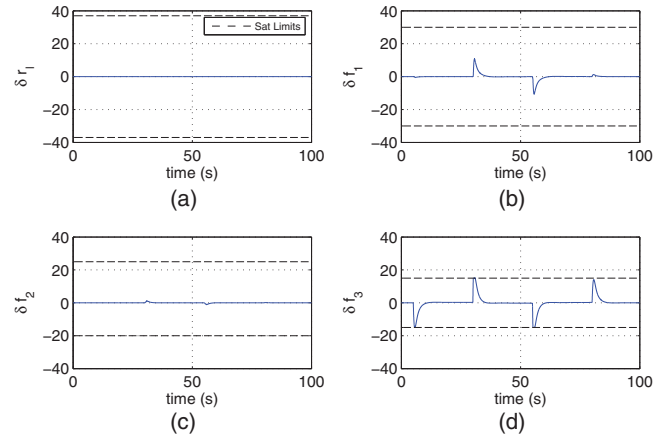


FIGURE 9 Left wing control signals (deg): failure case with optimised CA

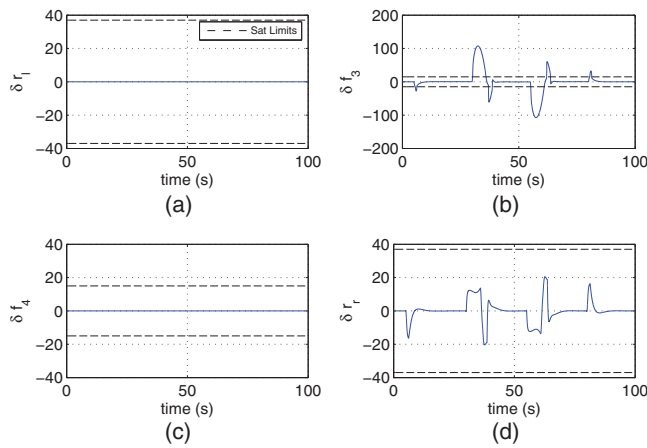


FIGURE 7 Control Signals (deg): faulty case fixed CA

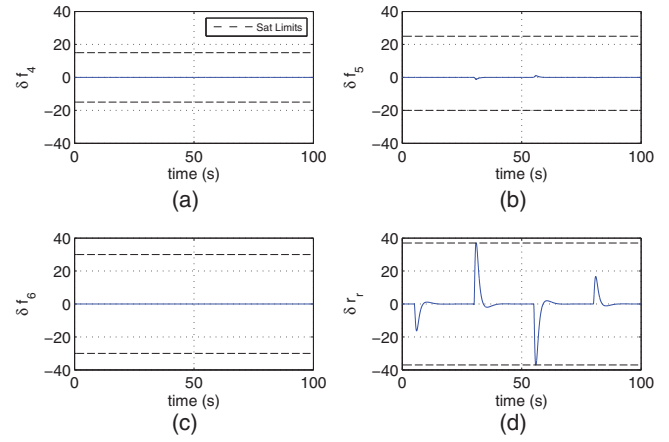


FIGURE 10 Right wing control signals (deg): failure case with optimised CA

8.3 | Faulty case—optimised CA

In this scenario, the same fault/failure case is considered as with the faulty fixed CA scenario; however, now $\Theta(t)$ is optimised on-line. The system response in Figure 8 demonstrates

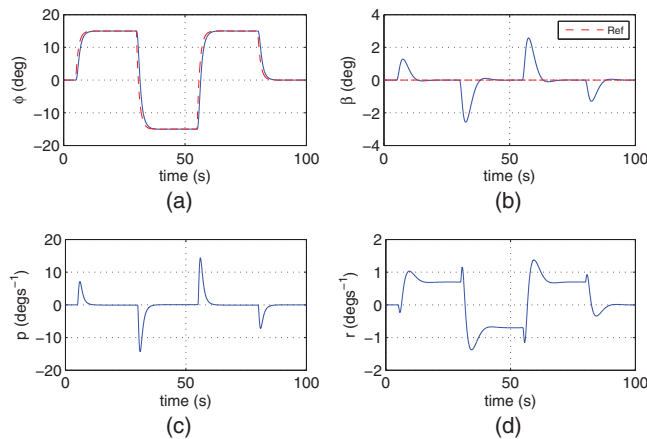


FIGURE 8 States: failure case with optimised CA

the effectiveness of the optimised CA scheme, providing a system response that is identical to the nominal case despite the actuator faults and failures. Figures 9–11 show the control signals generated by the optimised scheme. It can be seen that, whilst the majority of the control effort is still provided by the remaining actuators within the nominal set (f_3 and r_r), actuators from outside of the nominal set are used to help prevent saturation. In Figures 9d and 10d, it can be seen that the respective actuators both reach their saturation limits at 30 and 55 s, at which point other actuators, which do not belong to the nominal set, begin to provide some of the control effort, namely: δf_1 in Figure 9b and the two engines T_l and T_r in Figure 11.

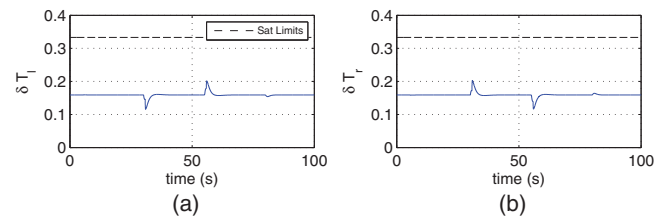


FIGURE 11 Engine control signals (MN): failure case with optimised CA

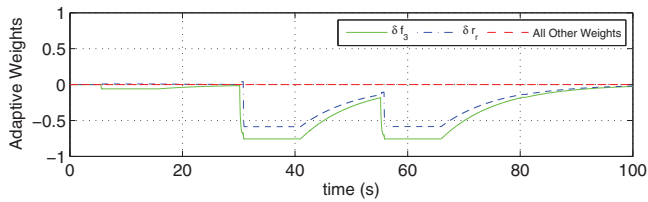


FIGURE 12 Adaptive components of $\Theta(t)$: failure case with optimised CA

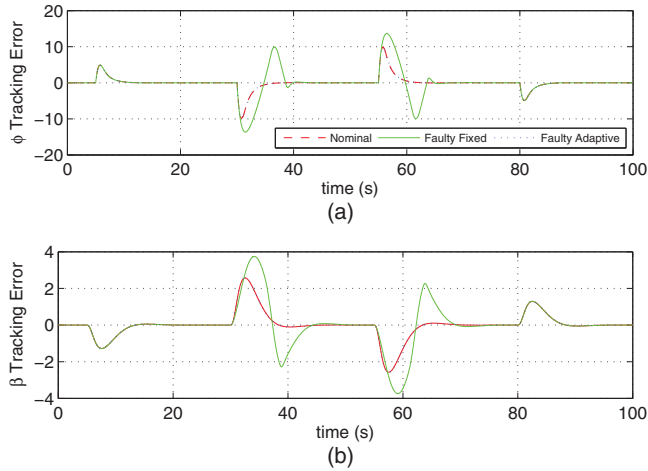


FIGURE 13 Comparison of tracking error for the three cases (deg)

In Figure 12, the product of the on-line optimisation is presented and shows that when the nominal CA begins to saturate, the adaptive component weights the saturating actuators negatively, effectively passing on the control effort to the other remaining actuators. Once the period of saturation ends, the effects of the “forgetting factor” can be seen as the weights begin to return to zero, retaining the nominal CA.

8.4 | Discussion

In Figure 13a, a comparison of the tracking errors for the three different scenarios are shown. The RMS (root mean squared) value of the tracking error signals are shown in Table 3. These values show that, when subjected to faults/failures, the fixed CA has nearly 99% worse tracking performance in ϕ and 61% worse tracking performance in β . In comparison, the tracking performance of the optimised CA, when subjected to the same

TABLE 3 RMS values of controlled output tracking error

Tracking Error		Faulty	Faulty
Output Channel	Nominal	Faulty CA	Optimised CA
ϕ	1.804	3.589	1.804
β	0.6993	1.138	0.6993

set of faults/failures, is basically identical to the fault-free case. Note that the fixed CA controller (developed previously in [3]) has the same “virtual” control law as the proposed scheme, and therefore provides a fair comparison between the new scheme and one that already exists in the literature. The results from the three different scenarios (Figures 4–13) show the shortcomings of the fixed CA controller and the advantages of the proposed scheme. This can most clearly be seen in Table 3. Although the fixed CA has the ability to deal with faults/failures, the controller has no method of accommodating saturation limits. The results clearly demonstrate that the more severe the fault/failure situation, the more likely actuator saturation is to occur due to the extra control effort assigned to the remaining healthy actuators. In the fixed CA case, this results in performance loss. The proposed scheme not only has the ability to handle faults/failures (through the same mechanism as the fixed CA) but also features an efficient adaptive component which allows it to adapt to saturation. In the case of severe faults/failures, this scheme was able to maintain the nominal fault-free performance by preventing saturation.

9 | CONCLUSION

This paper has presented a novel control allocation method which exclusively uses a subset of the actuator suite, until saturation occurs. A sliding mode controller is designed for use with control allocation, and rigorous conditions are derived to guarantee the stability of the closed-loop system. The scheme is shown to be capable of dividing the actuator suite (of an over-actuated system) such that a specified set of actuators is exclusively used; in the event that this set begins to saturate, the entire actuator suite is made available. An on-line optimisation is proposed to adapt the control allocation to ensure the total required control effort is met, whilst keeping actuators within their specified limits. The scheme is proven to be computationally “light” enough to run in real time. The effectiveness of the scheme is demonstrated using a high-fidelity blended wing body aircraft model. Simulation results show the scheme’s ability to maintain the nominal fault-free performance despite being presented with severe, uncertain faults/failures and saturation limits. In comparison, when an existing fixed control allocation law is used, some actuators are prone to saturate which results in a substantial amount of performance loss.

REFERENCES

- Okonkwo, P., Smith, H.: Review of evolving trends in blended wing body aircraft design. *Prog. Aerosp. Sci.* 82, 1–56 (2016)
- Johansen, T., Fossen, T.: Control allocation - A survey. *Automatica* 49(5), 1087–1103 (2013)
- Alwi, H., Edwards, C.: Fault tolerant control using sliding modes with on-line control allocation. *Automatica* 44(7), 1859–1866 (2008)
- Davidson, J., et al.: Real-time adaptive control allocation applied to a high performance aircraft. In: 5-th SIAM Conference on Control & Its Applications, San Diego, CA, 2001, pp. 1–11
- Campbell, B., Lieven, N.: Failure Tolerant Control using Minimal controller Synthesis for the X-38 Crew Return Vehicle. In: 5th Intern. ESA Conference on Spacecraft Guidance, Navigation and Control Systems, Frascati, Italy (2003)

6. Boskovic, J., Mehra, R. K.: Control allocation in overactuated aircraft under position and rate limiting. In: Proceedings of the 2002 American Control Conference, Anchorage, AK, 2002, pp. 791–796
7. Buffington, J., Enns, D.: Lyapunov stability analysis of daisy chain control allocation. *Journal of Guidance, Control, and Dynamics* 19(6), 1226–1230 (1996)
8. Tohidi, S., et al.: Adaptive control allocation for over-actuated systems with actuator saturation. *IFAC-PapersOnLine* 50(1), 5492–5497 (2017)
9. Vile, L., et al.: Sliding mode fault tolerant control allocation with saturation avoidance for a blended wing body aircraft. In: Conference on Decision and Control, Nice, France (2019)
10. Harkegard, O., Glad, S.: Resolving actuator redundancy - Optimal control vs. control allocation. *Automatica* 41(1), 137–144 (2005)
11. Matamoros, I., Visser, C. C.: Incremental nonlinear control allocation for a tailless aircraft with innovative control effectors. In: AIAA Guidance, Navigation, and Control Conference, Kissimmee, Florida, 8–12 January 2018
12. Castro, R., Brembeck, J. Lyapunov-based control allocation for over-actuated nonlinear systems. In: American Control Conference, Philadelphia, PA, USA (2019)
13. Waters, S., et al.: Control allocation performance for blended wing body aircraft and its impact on control surface design. *Aerosp. Sci. Technol.* 29(1), 18–27 (2013)
14. Yi, J., et al.: Nonlinear control allocation for a blended wing body aircraft. In: 2016 International Conference on Electrical Engineering and Automation, Xiamen, China, 2016, pp. 594–603
15. Vile, L., et al.: Fault tolerant control of a blended wing body aircraft using priority weighted control allocation and sliding modes. In: American Control Conference, Philadelphia, PA (2019).
16. Huijts, C., Voskuijl, M.: The impact of control allocation on trim drag of blended wing body aircraft. *Aerosp. Sci. Technol.* 46, 72–81 (2015)
17. Wells, S., Hess, R.: Multi-input/multi-output sliding mode control for a tailless fighter aircraft. *Journal of Guidance, Control, and Dynamics* 26(3), 463–473 (2003)
18. Shtessel, Y., et al.: Tailless aircraft flight control using multiple time scale reconfigurable sliding modes. *IEEE Trans. Control Syst. Technol.* 10, 288–296 (2002)
19. Ma, J., et al.: Robust reconfigurable flight control based on control reallocation for a tailless aircraft. In: International Symposium on Systems and Control in Aerospace and Astronautics, Shenzhen, China (2008)
20. Shuang, W., et al.: An anti-windup fault tolerant control scheme with guaranteed transient performance for tailless flying wing aircrafts. In: AIAA Guidance, Navigation, and Control Conference. Grapevine, TX, USA (2017)
21. Hamayun, M., et al.: A fault tolerant direct control allocation scheme with integral sliding modes. *Int. J. Appl. Math. Comput. Sci.* 25(1), 93–102 (2015)
22. Raj, K., et al.: Finite-time sliding mode and super-twisting control of fighter aircraft. *Aerosp. Sci. Technol.* 82–83, 487–498 (2018)
23. Lu, B., et al.: Adaptive super-twisting sliding mode fault tolerant control for MIMO uncertain aircraft system. In: 2017 36th Chinese Control Conference (CCC), Dalian, China, 2017, pp. 7087–7092
24. Wang, X., et al.: Incremental sliding-mode fault-tolerant flight control. *Journal of Guidance, Control, and Dynamics* 42(2) (2019)
25. Vile, L., Alwi, H., Edwards, C. Scheduled fault tolerant integral sliding mode control for BWB aircraft. In: 2020 European Control Conference (ECC), St Petersburg, Russia, 2020, pp. 1903–1908
26. Lombaerts, T. J. J., et al. Pseudo control hedging and its application for safe flight envelope protection. In: AIAA Guidance, Navigation, and Control Conference, Toronto, Canada (2010)
27. Ling, K., et al., A FPGA implementation of model predictive control. In: Proceedings of the American Control Conference, Minneapolis, MN (2006)
28. Fradkov, A., et al. Adaptive passification-based fault-tolerant flight control. In: 17th IFAC Symposium on Automatic Control in Aerospace, Toulouse, France (2007)
29. Edwards, C., et al.: Sliding mode observers for fault detection and isolation. *Automatica* 36(4), 541–553 (2000)
30. Zhang, Y., Jiang, J.: Design of integrated fault detection, diagnosis and reconfigurable control systems. In: Proceedings of the IEEE Conference on Decision and Control, Phoenix, AZ, USA, 1999, vol. 4, pp. 3587–3592
31. Alwi, H., Edwards, C.: Fault tolerant sliding modes control allocation with control surface priority weighting. In: Proceedings of the IEEE Int Conf on Control Applications, Yokohama, Japan, 2010, pp. 1057–1062
32. Stewart, G.: On scaled projections and pseudoinverses. *Linear Algebra Appl.* 112(C), 189–193 (1989)
33. Nocedal, J., Wright, S.: Numerical Optimization, Series in Operations Research and Financial Engineering. Springer, Berlin (2006)
34. Edwards, C., Spurgeon, S.: Sliding Mode Control: Theory and Applications. Taylor & Francis, London (1998)
35. Horn, R., Johnson, C.: Topics in Matrix Analysis. Cambridge University Press, Cambridge (2011)
36. Zames, G.: Input-Output Feedback Stability and Robustness, 1959–85. *IEEE Control Syst.* 16(3), 61–66 (1996)
37. Castro, H. Flying and handling qualities of a fly-by-wire blended-wing-body civil transport aircraft. Ph.D. thesis, Cranfield University (2003)
38. Rahman, N. U. Propulsion and flight controls integration for a blended wing body transport aircraft. Ph.D. thesis, Cranfield University (2010)
39. Melin, T. User's guide and reference manual for Tornado. <http://tornado.redhammer.se/images/manual.pdf>
40. Nelson, R.: Flight Stability and Automatic Control. McGraw-Hill Education, New York (1989)
41. Etkin, B., Teichmann, T.: Dynamics of Flight: Stability and Control. Wiley, New York (1959)
42. Qian, H. Counting the floating point operations (FLOPS). MATLAB Central File Exchange, 2020. uk.mathworks.com/matlabcentral/fileexchange/50608-counting-the-floating-point-operations-flops
43. Basford, P., et al.: Performance analysis of single board computer clusters. *Future Gener. Comput. Syst.* 102, 278–291 (2020)
44. Nikolskiy, V., Stegailov, V.: Floating-point performance of ARM cores and their efficiency in classical molecular dynamics. *J. Phys. Conf. Ser.* 681, 012049 (2016)
45. Alwi, H., Edwards, C.: Fault tolerant control of a civil aircraft using a sliding mode based scheme. In: Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, Seville, Spain (2005)

How to cite this article: Vile L, Alwi H, Edwards C. Robust fault tolerant control allocation for a modern over-actuated commercial aircraft. *IET Control Theory Appl.* 2021;15:307–322.
<https://doi.org/10.1049/cth2.12042>