

Scheduled smooth MIMO robust control of aircraft verified through blade element SIL testing

Transactions of the Institute of
Measurement and Control
1–14

© The Author(s) 2016

Reprints and permissions:

sagepub.co.uk/journalsPermissions.nav

DOI: 10.1177/0142331216661760

tim.sagepub.com



Coşku Kasnaoğlu

Abstract

This paper demonstrates a multi-input multi-output (MIMO) robust control approach where multiple scheduled designs are merged to produce a smooth control law. The design is verified using software-in-the-loop (SIL) testing based on blade element theory (BET) for highly realistic flight simulations. An inner-loop attitude controller balances performance and robustness, achieving a fast response time, low overshoot, good noise rejection and minimal lateral-longitudinal coupling. The controllers are formed at several predetermined grid points so the design covers a wide flight envelope. Blade element SIL testing shows that the flight control system preserves stable flight and follows the references well, even under tough weather conditions. The proposed strategy is also compared with a classical autopilot design procedure and is seen to be superior.

Keywords

Blade element theory (BET), flight control, multi-input multi-output (MIMO), non-linear modelling, robust control, smooth control, software-in-the-loop (SIL).

Introduction

Autopilot systems control the attitude and trajectory of an aircraft without input from a human pilot (Blakelock, 1991; Etkin and Reid, 1996; Nelson, 1998). Autopilot technology has progressed considerably in time, from initial designs that simply kept straight and level flight, to current autopilots capable of trajectory following and landing (Lungu et al., 2013; Lungu and Lungu, 2015; Stevens and Lewis, 2003). Lately, autopilot design for airplanes, multicopters and missiles have received substantial research attention. There exists a variety of scientific studies making use of robust and adaptive dynamic inversion (Lungu and Lungu, 2012, 2016), reconfigurable flight control laws (Calise et al., 2001), robust non-linear control (Wang and Stengel, 2000), Lyapunov vector fields (Lawrence et al., 2008), command-filtered backstepping (Du et al., 2015), sliding mode control (Liu et al., 2015), multiple model adaptive control (Ahmadian et al., 2015) and invariant manifolds (Karagiannis and Astolfi, 2010).

Classical autopilot designs typically approximated the flight channels as independent and focused on single-input single-output (SISO) controllers (Blakelock, 1991; Nelson, 1998). Recently, interest in multi-input multi-output (MIMO) controller methods has increased for missiles (Bruyere et al., 2003; Choi et al., 2012; Kang et al., 2009; Tsay, 2007), helicopters (Basri et al., 2014; Sheng et al., 2014; Yang and Liu, 2003) and multicopter vehicles (Ahmad et al., 2001; Efe, 2010; Liu et al., 2014; Özbek et al., 2015), where dynamical couplings are prominent. For fixed-wing aircraft, however, these effects are less pronounced, so approximate SISO techniques still dominate the process.

Nevertheless, there are scenarios such as high crosswinds, aerobatics, structural damage and actuator malfunctions that may result in the aircraft being pushed out of the usual attitudes (de Voogt and van Doorn, 2009; Wong et al., 2006). These scenarios invalidate the standard decoupling assumptions and call for a coupled MIMO design.

It is also not uncommon in classical autopilot design that a controller built for a single operating point is used for the entire flight envelope. This may work for the nominal case but the performance can degrade and destabilize the aircraft under certain conditions (Gill et al., 2015). Designing controllers at multiple trim points and switching among these (gain scheduling) can alleviate this problem and is common practice in missiles. It can be shown, however, that such switching can lead to discontinuities in the control input, which may result in an unstable closed loop, even though all individual controllers are stabilizing (Rugh and Shamma, 2000). Among various methods proposed to eliminate such discontinuities, research work utilizing controller blending for robust flight control is of special interest for this study (Adams et al., 2012). Particular examples include linear parameter varying control with blending for the longitudinal axis of a fighter jet

Department of Electrical and Electronics Engineering, TOBB University of Economics and Technology, Ankara, Turkey

Corresponding author:

Coşku Kasnaoğlu, Department of Electrical and Electronics Engineering, TOBB University of Economics and Technology, Sogutozu Caddesi No. 43, Sogutozu, Ankara 06560, Turkey.

Email: kasnakoglu@etu.edu.tr; kasnakoglu@gmail.com

(Shin et al., 2002), blending based on state–space Youla parameterization for a gain-scheduled missile pitch-axis autopilot (Stilwell, 2001), and scheduling schemes with blending for the flight and propulsion control of a vertical/short take-off and landing aerial vehicle (Aouf et al., 2002).

This paper is an aviation-oriented augmentation of MIMO robust control techniques with smooth scheduling, presented as a systematic methodology on a case study from start to finish. Robust controllers shaping the linearized dynamics are designed at multiple trim points, which are then integrated by a weighing procedure to produce a smooth control law. SIL verification is performed using blade element theory (BET) (Burton et al., 2001; Ingram, 2005), in which all aircraft surfaces are modelled as ‘blade elements’. BET flight simulations provide very realistic results and they are not based on ordinary differential equation models of the aircraft. The latter are typically employed in designing flight controllers, so avoiding the same type of model in the testing phase is advantageous.

As cited above, the literature contains certain studies related to the individual tools utilized in the proposed methodology (e.g. aircraft modelling, MIMO robust control, scheduling, blending, BET) applied to specific modes of the flight dynamics (e.g. roll control, pitch control, navigation and landing). However, it is hard to find a single source merging all of these tools for the goal of designing a complete robust flight control system while viewing the aircraft dynamics as a whole. This serves as the main contribution of this work and contrasts with the classical mindset of forming separate controllers for individual longitudinal/lateral modes within the navigation/landing phases and then stacking these together to form the overall flight controller.

The rest of the paper is organized as follows: the next section derives the mathematical model and then the controller design methodology is presented. The verification method is outlined and the results of a case study are illustrated. Finally, the paper ends with conclusions and ideas for future work.

Mathematical model

The model of the aircraft dynamics is obtained from the force and moment equations:

$$\mathbf{F} = m \left(\frac{\partial \mathbf{V}}{\partial t} + \boldsymbol{\Omega} \times \mathbf{V} \right) \quad (1)$$

$$\mathbf{M} = \frac{\partial (\mathbf{I} \cdot \boldsymbol{\Omega})}{\partial t} + \boldsymbol{\Omega} \times (\mathbf{I} \cdot \boldsymbol{\Omega}) \quad (2)$$

where $\mathbf{V} = [u \ v \ w]^T$ is the velocity vector at the centre of gravity, $\boldsymbol{\Omega} = [p \ q \ r]^T$ is the angular velocity vector about the centre of gravity, $\mathbf{F} = [F_x \ F_y \ F_z]^T$ is the total external force vector, $\mathbf{M} = [L \ M \ N]^T$ is the total external moment vector and \mathbf{I} is the inertia tensor. With proper manipulations, it is possible to obtain a non-linear state space system of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{F}(t), \mathbf{M}(t)) \quad (3)$$

with

$$\mathbf{F} = \mathbf{g}_1(\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t), t) \quad (4)$$

$$\mathbf{M} = \mathbf{g}_2(\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t), t) \quad (5)$$

These equations can be combined into a compact form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{v}(t), t) \quad (6)$$

with state vector \mathbf{x} , input vector \mathbf{u} , disturbance vector \mathbf{v} and time t . The state vector \mathbf{x} ordinarily contains three linear and three angular velocities from \mathbf{V} and $\boldsymbol{\Omega}$, but for practical applications, it is generally less complicated to work with airspeed, angle of attack and sideslip angle as opposed to the linear velocity components. This yields the following state vector:

$$\mathbf{x} = [V \ \alpha \ \beta \ p \ q \ r \ \psi \ \theta \ \phi \ x_e \ y_e \ z_e]^T \quad (7)$$

in terms of which the state space equations can be written (Appendix A). To solve the differential equations in (3), one must obtain the force and moment values $\mathbf{F} = [F_x \ F_y \ F_z]^T$ and $\mathbf{M} = [L \ M \ N]^T$. These are functions of various mass and geometry parameters, the thrust mechanism, as well as the control commands. These forces and moments are conveniently expressed in terms of stability derivatives, which capture the effect of different key parameters of a given force or moments value. For instance, the longitudinal aerodynamical force may be written as

$$F_x = C_{X_0} + C_{X_\alpha} \alpha + C_{X_{\alpha^2}} \alpha^2 + C_{X_{\alpha^3}} \alpha^3 + C_{X_q} \frac{q\bar{c}}{V} + C_{X_{\delta_r}} \delta_r + C_{X_{\delta_f}} \delta_f + C_{X_{\alpha\delta_f}} \alpha \delta_f \quad (8)$$

where C_{X_0} , C_{X_α} , $C_{X_{\alpha^2}}$, $C_{X_{\alpha^3}}$, C_{X_q} , $C_{X_{\delta_r}}$, $C_{X_{\delta_f}}$ and $C_{X_{\alpha\delta_f}}$ are the stability derivatives representing the effect of the term that they multiply on F_x . Expressions for F_y , F_z , L , M and N may be written similarly in terms of their corresponding stability derivatives (Rauw, 2001).

Controller design

The aircraft is first trimmed on a two-dimensional grid based on elevation and airspeed. These parameters vary considerably during the flight and have a significant effect on the system dynamics. Denoting the breakpoints for velocity and elevation as

$$V = [v_1 \ v_2 \ \dots \ v_M] \quad (9)$$

$$Z_e = [z_{e,1} \ z_{e,2} \ \dots \ z_{e,N}] \quad (10)$$

the aircraft is to be trimmed at $M \times N$ grid points of the form $(v_0, z_{e0}) \in V \times Z_e$. At each trim point, the goal is to solve the following equations for x_0 and u_0

$$\begin{aligned} v = v_0, z_e = z_{e,0}, \frac{\partial \alpha}{\partial t} = \frac{\partial \beta}{\partial t} = \frac{\partial p}{\partial t} = \frac{\partial q}{\partial t} = \frac{\partial r}{\partial t} = \frac{\partial \psi}{\partial t} \\ = \frac{\partial \theta}{\partial t} = \frac{\partial \phi}{\partial t} = \frac{\partial z_e}{\partial t} = 0 \end{aligned} \quad (11)$$

where v_0 is the desired airspeed, $z_{e,0}$ is the desired elevation and the time derivatives are obtained from the non-linear dynamical equations given in (47)–(62). The trimming process can be posed as the following general optimization problem as

$$\min_{x,u} f(x,u) \quad (12)$$

subject to

$$g_1(x,u) = 0, g_2(x,u) = 0 \quad (13)$$

where

$$f(x,u) = \left(\frac{\partial \alpha}{\partial t}\right)^2 + \left(\frac{\partial \beta}{\partial t}\right)^2 + \left(\frac{\partial p}{\partial t}\right)^2 + \left(\frac{\partial q}{\partial t}\right)^2 + \left(\frac{\partial r}{\partial t}\right)^2 + \left(\frac{\partial \psi}{\partial t}\right)^2 + \left(\frac{\partial \theta}{\partial t}\right)^2 + \left(\frac{\partial \phi}{\partial t}\right)^2 + \left(\frac{\partial z_e}{\partial t}\right)^2 \quad (14)$$

and

$$g_1(x,u) = v - v_0, g_2(x,u) = z_e - z_{e,0} \quad (15)$$

Ideally, $f(x,u) = 0$, but due to numerical implementation and round-off errors, $f(x,u) < 10^{-3}$ is acceptable in practice. It is convenient to augment x and u into a single vector \bar{x} for optimization as follows

$$\bar{x} = [x^T u^T]^T \quad (16)$$

The derivative terms in $f(\bar{x}) = f(x,u)$ are highly non-linear, so a fast and efficient non-linear programming method, namely sequential quadratic programming (SQP), is employed (Appendix B). The operating point solving the optimization problem in (12)–(15) is denoted as (x_0, u_0) , where $x_0 = [v_0, \alpha_0, \beta_0, p_0, q_0, r_0, \psi_0, \theta_0, \phi_0, x_{e0}, y_{e0}, z_{e0}]^T$ is the vector of the aircraft states at the trim condition and $u_0 = [F_{x0}, \delta_{e0}, \delta_{a0}, \delta_{r0}]^T$ is the vector of control inputs to be applied at the trim condition. Apart from the 12 state parameters, lateral acceleration A_y (g) and the azimuth angle χ (rad) are also used in the autopilot design. The task is to design an autopilot to carry out the following references from the operator: airspeed command (v_{cmd}), altitude command (z_e, cmd), flight direction command (χ_{cmd}) and lateral acceleration command ($A_{y,cmd}$). The last command is always set to zero to minimize the lateral force acting on the aircraft. As a change in the flight direction (χ) is achieved by banking (i.e. a change in ϕ), it is reasonable to first design an inner controller to set the roll angle (ϕ) to a desired value. An outer controller is then wrapped around to send ϕ_{cmd} to this inner controller based on χ_{cmd} as in Figure 1.

For inner controller design, the non-linear aircraft model is linearized about the trim conditions (x_0, u_0) , which yields a linear state-space system \mathbf{G} of the form

$$\mathbf{G} : \begin{cases} \dot{\tilde{x}} = A\tilde{x} + B\tilde{u} \\ y = C\tilde{x} + D\tilde{u} \end{cases} \quad (17)$$

where $\tilde{x} = x - x_0$, $\tilde{u} = u - u_0$ and

$$\mathbf{A} = \frac{\partial f}{\partial x}(x_0, u_0), \mathbf{B} = \frac{\partial f}{\partial u}(x_0, u_0), \mathbf{C} = \frac{\partial h}{\partial x}(x_0, u_0), \mathbf{D} = \frac{\partial h}{\partial u}(x_0, u_0) \quad (18)$$

The vectors fields $f(x,u)$ and $h(x,u)$ contain respectively the equations for derivatives of the states and the outputs

$y = [v, z_e, \phi, A_y]^T$ to be controlled. The linearized system can also be expressed in transfer function matrix form

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (19)$$

where \mathbf{I} is the identity matrix.

A MIMO robust controller \mathbf{K} is designed next for \mathbf{G} using loop-shaping approach. In robust multivariable control design, the desired specifications are commonly expressed as

$$\bar{\sigma}(\mathbf{S}(j\omega)) \leq |\mathbf{W}_1^{-1}(j\omega)| \left(\text{or equivalently } \frac{1}{\bar{\sigma}(\mathbf{S}(j\omega))} \geq |\mathbf{W}_1(j\omega)| \right) \quad (20)$$

$$\bar{\sigma}(\mathbf{T}(j\omega)) \leq |\mathbf{W}_3^{-1}(j\omega)| \quad (21)$$

where $\bar{\sigma}$ and $\bar{\sigma}$ denote minimum and maximum singular values, respectively. Here, $\mathbf{S}(s)$ is the sensitivity function defined as

$$\mathbf{S}(s) = (\mathbf{I} + \mathbf{L}(s))^{-1} \quad (22)$$

$\mathbf{T}(s)$ is the complementary sensitivity function defined as

$$\mathbf{T}(s) = \mathbf{L}(s)(\mathbf{I} + \mathbf{L}(s))^{-1} \quad (23)$$

$\mathbf{L}(s)$ is the loop transfer function matrix

$$\mathbf{L}(s) = \mathbf{G}(s)\mathbf{K}(s) \quad (24)$$

$|\mathbf{W}_1^{-1}(j\omega)|$ is the desired disturbance attenuation factor and $|\mathbf{W}_3(j\omega)|$ is the largest anticipated uncertainty of the plant expressed as a multiplicative perturbation. Note that the singular values of $\mathbf{S}(j\omega)$ determine the disturbance attenuation, as $\mathbf{S}(s)$ is actually the closed-loop transfer function from an output disturbance d to plant output y . Note also that $\mathbf{T}(s)$ is indeed the closed-loop transfer function on the whole system. A stabilizing H_∞ controller \mathbf{K} is computed for plant \mathbf{G} to make the sigma plot of the loop transfer function $\mathbf{G}\mathbf{K}$ have the desired loop shape \mathbf{G}_d with accuracy γ . The specifications on disturbance attenuation and multiplicative stability margin in (20) and (21) can be written in terms of singular values of the loop transfer function, as one can make the following approximation for $\bar{\sigma}(\mathbf{L}(s)) \gg 1$

$$\mathbf{S}(s) = (\mathbf{I} + \mathbf{L}(s))^{-1} \approx \mathbf{L}(s)^{-1} \quad (25)$$

and the one below for $\bar{\sigma}(\mathbf{L}(s)) \ll 1$

$$\mathbf{T}(s) = \mathbf{L}(s)(\mathbf{I} + \mathbf{L}(s))^{-1} \approx \mathbf{L}(s) \quad (26)$$

Therefore, if ω_c is the 0 dB crossover frequency of the singular values plot of $\mathbf{G}_d(j\omega)$, the specifications can be stated as

$$\underline{\sigma}(\mathbf{G}(j\omega)\mathbf{K}(j\omega)) \geq \frac{1}{\gamma}\underline{\sigma}(\mathbf{G}_d(j\omega)), \forall \omega < \omega_c \quad (27)$$

$$\bar{\sigma}(\mathbf{G}(j\omega)\mathbf{K}(j\omega)) \leq \gamma\bar{\sigma}(\mathbf{G}_d(j\omega)), \forall \omega > \omega_c \quad (28)$$

Thus, high tracking performance is achieved at low frequencies where the system model is more accurate, and high robustness is achieved at high frequencies where the system

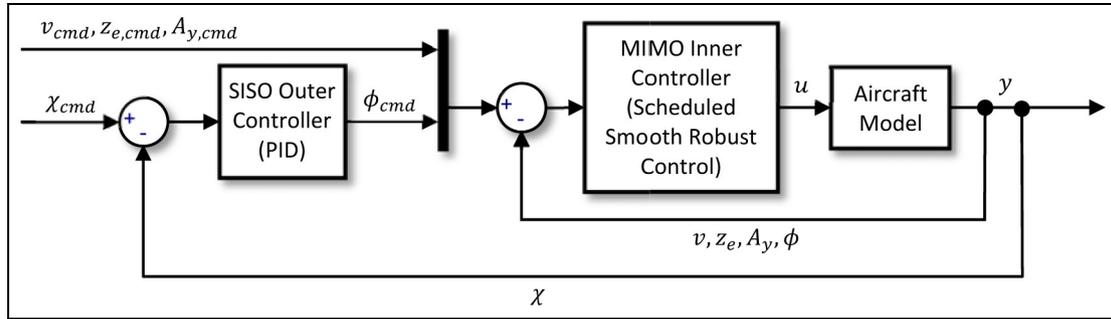


Figure 1. Closed-loop system structure for autopilot design.

model is less accurate and noise effects are stronger. A stable minimum-phase loop-shaping, squaring-down prefilter \mathbf{W} is computed using greatest common divisor (GCD) formulas (Le and Safonov, 1992) such that the shaped plant $\mathbf{G}_s = \mathbf{G}\mathbf{W}$ is square and that the desired shape \mathbf{G}_d is achieved with good accuracy in a desired frequency range $(\omega_{min}, \omega_{max})$ by the shaped plant, i.e.

$$\sigma(\mathbf{G}_d) \approx \sigma(\mathbf{G}_s), \forall \omega \in (\omega_{min}, \omega_{max}) \quad (29)$$

Normalized coprime factor synthesis theory is then used to compute an optimal loop-shaping controller for the shaped plant. (Appendix C). Such a controller is computed at every operating point and these are merged into a single controller, which produces a smooth weighted output of the individual controllers. If at time t it holds that

$$v_i < v(t) \leq v_{i+1}, z_{e,i} < z_e(t) \leq z_{e,i+1} \quad (30)$$

then the input to the aircraft is computed as a convex combination (Boyd and Vandenberghe, 2004) of the form

$$u(t) = c_1(t)u_1(t) + c_2(t)u_2(t) + c_3(t)u_3(t) + c_4(t)u_4(t) \quad (31)$$

where $u_1(t), u_2(t), u_3(t)$ and $u_4(t)$ are the outputs of the controllers designed at grid points $(v_i, z_{e,i}), (v_{i+1}, z_{e,i}), (v_i, z_{e,i+1})$ and $(v_{i+1}, z_{e,i+1})$, respectively, and $c_1(t), c_2(t), c_3(t)$ and $c_4(t) \in \mathbb{R}^+$ satisfy

$$c_1(t) + c_2(t) + c_3(t) + c_4(t) = 1 \quad (32)$$

The weights $c_1(t), c_2(t), c_3(t)$ and $c_4(t)$ indicate how close the current velocity $v(t)$ and elevation $z(t)$ are to the grid points and are computed as

$$\begin{aligned} c_1(t) &= (1 - \lambda_1(t))(1 - \lambda_2(t)) \\ c_2(t) &= \lambda_1(t)(1 - \lambda_2(t)) \\ c_3(t) &= (1 - \lambda_1(t))\lambda_2(t) \\ c_4(t) &= \lambda_2(t) \end{aligned} \quad (33)$$

where

$$\lambda_1(t) = \frac{v(t) - v_i}{v_{i+1} - v_i} \quad (34)$$

$$\lambda_2(t) = \frac{z_e(t) - z_{e,i}}{z_{e,i+1} - z_{e,i}} \quad (35)$$

so that

$$\begin{aligned} \begin{bmatrix} v(t) \\ z_e(t) \end{bmatrix} &= c_1(t) \begin{bmatrix} v_i \\ z_{e,i} \end{bmatrix} + c_2(t) \begin{bmatrix} v_{i+1} \\ z_{e,i} \end{bmatrix} \\ &+ c_3(t) \begin{bmatrix} v_i \\ z_{e,i+1} \end{bmatrix} + c_4(t) \begin{bmatrix} v_{i+1} \\ z_{e,i+1} \end{bmatrix} \end{aligned} \quad (36)$$

It is straightforward to verify that the convex combination condition (32) is satisfied. This concludes the design of the inner controller (the middle block in Figure 1).

An outer controller is then wrapped around the inner controller to generate an appropriate roll angle command ϕ_{cmd} from a desired flight direction χ_{cmd} . As the outer controller is a SISO compensator, a simple proportional integral derivative (PID) controller is utilized

$$\frac{\Phi_{cmd}(s)}{X_{cmd}(s)} = K_p + \frac{K_i}{s} + \frac{K_d s}{s/N + 1} \quad (37)$$

The filtered derivative suppresses the noise effects associated with differentiation and ensures a practically realizable system. The coefficients can be selected empirically or via automated tuning methods such as Ziegler–Nichols or internal model control (IMC) (Ho et al., 1995). This finalizes the construction of the flight control system.

Verification

The control system is verified with software-in-the-loop (SIL) simulations incorporating BET. The surfaces of the aircraft (e.g. propellers, wings and stabilizers) are divided into several sections, the lift/drag forces acting on each section are computed separately and the composite effect is applied to the entire aerial vehicle (Figure 2). This approach contrasts with traditional flight simulations relying on empirical data (e.g. stability derivatives) in predefined lookup tables and is widely accepted to be more realistic albeit computationally expensive. It is also a good choice for testing the control design presented here, as the mathematical model utilized is based on stability derivatives. Hence, a different (and more accurate) flight simulation technique that does not rely on stability derivatives serves as a better test. The main idea of BET can be

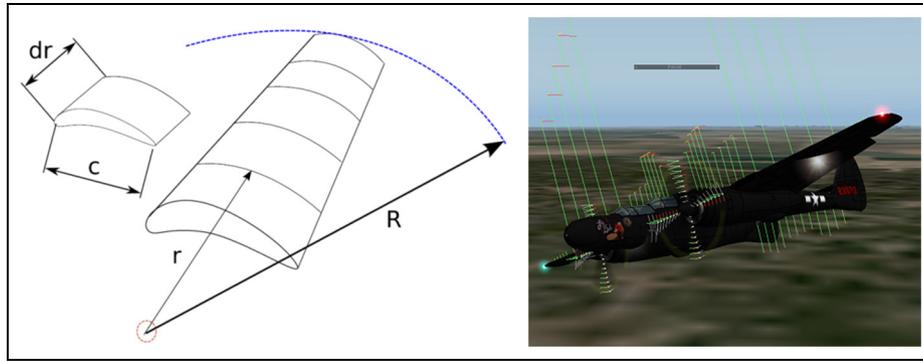


Figure 2. Illustration of the main idea of blade element theory (BET) on a propeller blade (left). During flight simulation, the aircraft is split into a number of surfaces and the forces on each are computed by BET (right).

summarized on a propeller blade shown in Figure 2 (left). The blade is divided into N elements, each of which experiences a slightly different flow. Lift and drag coefficients (C_L and C_D) are readily available for numerous airfoil shapes from wind tunnel tests. Using relative velocities, the flow over each element can be related to these tests. The flow is slightly turned passing over the airfoil so inlet and exit flow conditions are averaged to improve accuracy. Carrying out the necessary computations yields

$$dF_x = dL \sin \beta + dD \cos \beta \quad (38)$$

$$dF_\theta = dL \cos \beta - dD \sin \beta \quad (39)$$

$$dL = \sigma' \pi \rho \frac{V^2 (1-a)^2}{\cos^2 \beta} C_L r dr \quad (40)$$

$$dD = \sigma' \pi \rho \frac{V^2 (1-a)^2}{\cos^2 \beta} C_D r dr \quad (41)$$

$$\sigma' = \frac{Bc}{2\pi r} \quad (42)$$

where dL , dD , dF_x and dF_θ are respectively the lift, drag, axial and tangential forces, β is the relative flow angle, ρ is the air density, V is the flow velocity, a is the axial induction factor, r is the radius, σ' is the local solidity, B is the number of blades and c is chord length (Ingram, 2005). The procedure is carried out on the entire aircraft to compute all the forces, using which the flight dynamics can be simulated as in Figure 2 (right).

This step concludes the design and verification of the controller design. A flowchart of the entire process is given in Figure 3. A case study is presented in the next section.

Application example

In this section, the autopilot design methodology outlined in the previous section is demonstrated on a popular general aviation aircraft, namely the Cessna 172. The geometry and mass parameters of this aircraft, together with its stability derivatives and performance specifications are given in Table 1. Based on the table, a 6×6 grid of operating points are constructed with all possible combinations of airspeeds

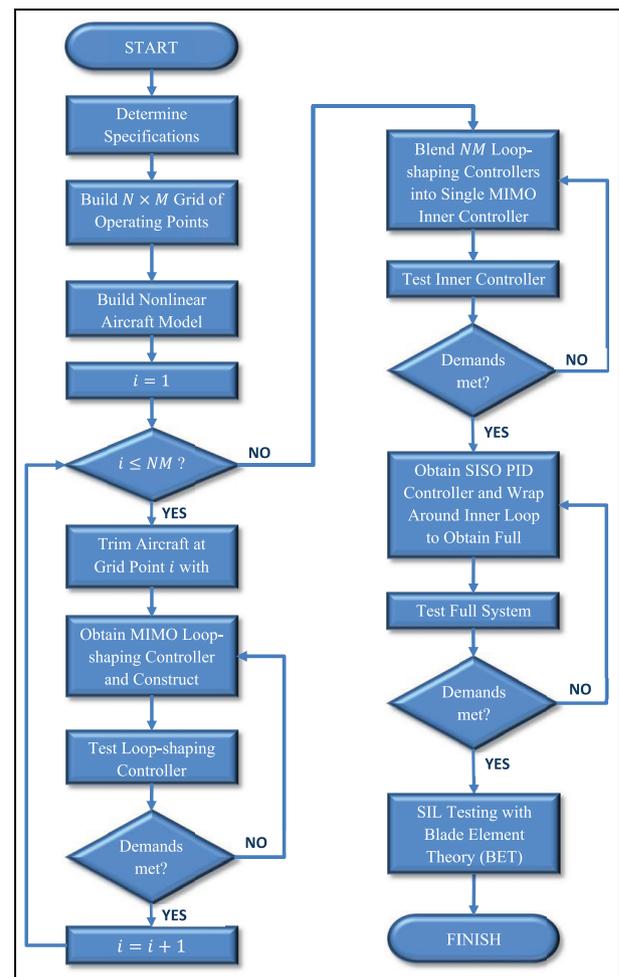


Figure 3. Flowchart outlining the flight control system design methodology presented.

$v = [25 \ 35 \ 45 \ 55 \ 65 \ 75]$ and elevation values $z_e = [0 \ 500 \ 1000 \ 1500 \ 2000 \ 2500]$. At each value, the optimization problem given in (12)–(15) is solved by SQP, as described previously, to obtain the operating points given in Table 2. For each operating point, an inner is controller designed with desired loop-shape

Table 1. Mass, geometry parameters, performance specifications and stability derivatives for Cessna 172.

Geometry and mass parameters		Aerodynamic D-force derivatives		Aerodynamic L-force derivatives		Aerodynamic Y-force derivatives	
\bar{c}	1.4935	C_{D_0}	0.031	C_{L_0}	0.31	C_{Y_0}	0
b	10.9118	C_{D_α}	0.13	C_{L_α}	5.143	C_{Y_β}	-0.31
S	16.1651	C_{D_q}	0	C_{L_q}	3.9	C_{Y_p}	-0.037
I_{xx}	1285.3	$C_{D_{\delta_e}}$	0.06	$C_{L_{\delta_e}}$	0.43	C_{Y_r}	0.21
I_{yy}	1824.9	C_{D_h}	0	C_{L_h}	0	$C_{Y_{\delta_a}}$	0.0
I_{zz}	2666.9					$C_{Y_{\delta_r}}$	0.187
J_{xy}	0						
J_{xz}	0						
J_{yz}	0						
m	1043.3						

Performance specifications		Aerodynamic L-moment derivatives		Aerodynamic M-moment derivatives		Aerodynamic N-moment derivatives	
Cruise speed	65	C_{l_0}	0	C_{m_0}	-0.015	C_{n_0}	0
Stall speed	24	C_{l_β}	-0.089	C_{m_α}	-0.89	C_{n_β}	0.065
Never exceed speed	84	C_{l_p}	-0.47	C_{m_q}	-12.4	C_{n_p}	-0.03
Maximum crosswind	7.7	C_{l_r}	0.096	$C_{m_{\delta_e}}$	-1.28	C_{n_r}	-0.099
Service ceiling	4100	$C_{l_{\delta_a}}$	-0.178	C_{m_h}	0	$C_{n_{\delta_a}}$	-0.053
		$C_{l_{\delta_r}}$	0.0147			$C_{n_{\delta_r}}$	-0.0657

All units are SI.

$$\mathbf{G}_d(s) = \text{diag}\left(\frac{2}{s}, \frac{2}{s}, \frac{2}{s}, \frac{2}{s}\right) \quad (43)$$

where ‘diag’ stands for the diagonal matrix. Figure 4 shows the singular value plot at one particular grid point $v = 65$ m/s and $z_e = 1000$ m. It is seen that the loop transfer function $\mathbf{L}(s)$ approximates $\mathbf{G}_d(s)$ within certain tolerance bounds. The bounds on the singular values of the sensitivity function $\mathbf{S}(s)$ and complementary sensitivity function $\mathbf{T}(s)$ favour performance at low frequencies and robustness at high frequencies, as described previously. As $\mathbf{L}(s) \approx \mathbf{G}_d(s)$, this suggests that

$$\mathbf{T}(s) \approx \mathbf{L}(s)(\mathbf{I} + \mathbf{L}(s))^{-1} \approx \mathbf{G}_d(s)(\mathbf{I} + \mathbf{G}_d(s))^{-1} \quad (44)$$

from which one obtains

$$\mathbf{T}(s) = \text{diag}\left(\frac{2}{s+2}, \frac{2}{s+2}, \frac{2}{s+2}, \frac{2}{s+2}\right) \quad (45)$$

$$= \begin{bmatrix} \frac{2}{s+2} & 0 & 0 & 0 \\ 0 & \frac{2}{s+2} & 0 & 0 \\ 0 & 0 & \frac{2}{s+2} & 0 \\ 0 & 0 & 0 & \frac{2}{s+2} \end{bmatrix}$$

From here, one observes the following: the individual transfer functions for the diagonal channels are approximated by

$$\frac{V_{cmd}(s)}{V(s)} = \frac{z_{e,cmd}(s)}{z_e(s)} = \frac{A_{y,cmd}(s)}{A_y(s)} = \frac{\phi_{cmd}(s)}{\phi(s)} \approx \frac{2}{s+2} \quad (46)$$

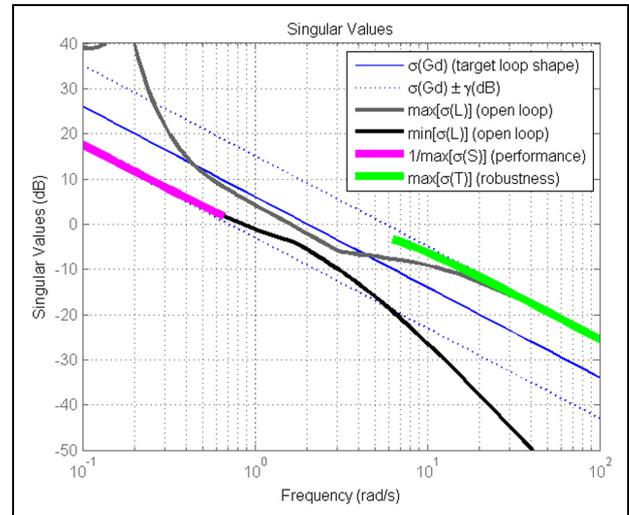


Figure 4. Singular value plot for the loop-shaping controller at grid point $v = 65$ m/s and $z_e = 1000$ m. Frequency characteristics are similar for other operating points. The design balances performance and robustness by boosting the gain high at low frequencies where the model is accurate and suppressing it at high frequencies where the model is poor and noise effects are dominant.

so the closed-loop system will be able to track all references successfully with minimal overshoot and a settling time of approximately $t_s = 5\tau = 5 \cdot \frac{1}{2} = 2.5$ s.

The off-diagonal entries of $\mathbf{T}(s)$ are roughly zero, which indicates that the coupling between different command-response pairs are eliminated.

Table 2. Trim points for Cessna 172 at all possible combinations of airspeeds $v = [25\ 35\ 45\ 55\ 65\ 75]$ and heights $z_e = [0\ 500\ 1000\ 1500\ 2000\ 2500]$.

Grid point		State at trim point (x_0)									Input at trim point (u_0)						
v_0	z_{e0}	v	α	β	p	q	r	ψ	θ	ϕ	x_e	y_e	z_e	F_x	δ_e	δ_a	δ_r
25	0	25	0.275181	0	0	0	0	0	0.275181	0	0	0	0	351.0216	-0.20306	0	0
35	0	35	0.110271	0	0	0	0	0	0.110271	0	0	0	0	488.5068	-0.08839	0	0
45	0	45	0.042093	0	0	0	0	0	0.042093	0	0	0	0	682.5558	-0.04099	0	0
55	0	55	0.007516	0	0	0	0	0	0.007516	0	0	0	0	927.3178	-0.01694	0	0
65	0	65	-0.01239	0	0	0	0	0	-0.01239	0	0	0	0	1221.721	-0.0031	0	0
75	0	75	-0.02489	0	0	0	0	0	-0.02489	0	0	0	0	1565.468	0.005586	0	0
25	500	25	0.29171	0	0	0	0	0	0.29171	0	0	0	500	345.0801	-0.21455	0	0
35	500	35	0.118764	0	0	0	0	0	0.118764	0	0	0	500	474.6706	-0.0943	0	0
45	500	45	0.047244	0	0	0	0	0	0.047244	0	0	0	500	659.2422	-0.04457	0	0
55	500	55	0.010969	0	0	0	0	0	0.010969	0	0	0	500	892.3518	-0.01935	0	0
65	500	65	-0.00992	0	0	0	0	0	-0.00992	0	0	0	500	1172.829	-0.00482	0	0
75	500	75	-0.02303	0	0	0	0	0	-0.02303	0	0	0	500	1500.351	0.004293	0	0
25	1000	25	0.309237	0	0	0	0	0	0.309237	0	0	0	1000	339.5565	-0.22673	0	0
35	1000	35	0.127774	0	0	0	0	0	0.127774	0	0	0	1000	461.3946	-0.10056	0	0
45	1000	45	0.05271	0	0	0	0	0	0.05271	0	0	0	1000	636.8067	-0.04837	0	0
55	1000	55	0.014633	0	0	0	0	0	0.014633	0	0	0	1000	858.6826	-0.02189	0	0
65	1000	65	-0.00729	0	0	0	0	0	-0.00729	0	0	0	1000	1125.742	-0.00665	0	0
75	1000	75	-0.02106	0	0	0	0	0	-0.02106	0	0	0	1000	1437.634	0.002922	0	0
25	1500	25	0.327834	0	0	0	0	0	0.327834	0	0	0	1500	334.4649	-0.23967	0	0
35	1500	35	0.137338	0	0	0	0	0	0.137338	0	0	0	1500	448.6697	-0.10721	0	0
45	1500	45	0.058513	0	0	0	0	0	0.058513	0	0	0	1500	615.2281	-0.0524	0	0
55	1500	55	0.018523	0	0	0	0	0	0.018523	0	0	0	1500	826.2768	-0.0246	0	0
65	1500	65	-0.0045	0	0	0	0	0	-0.0045	0	0	0	1500	1080.413	-0.00859	0	0
75	1500	75	-0.01896	0	0	0	0	0	-0.01896	0	0	0	1500	1377.253	0.001465	0	0
25	2000	25	0.347578	0	0	0	0	0	0.347578	0	0	0	2000	329.8235	-0.25339	0	0
35	2000	35	0.147499	0	0	0	0	0	0.147499	0	0	0	2000	436.4876	-0.11428	0	0
45	2000	45	0.064678	0	0	0	0	0	0.064678	0	0	0	2000	594.486	-0.05669	0	0
55	2000	55	0.022656	0	0	0	0	0	0.022656	0	0	0	2000	795.1016	-0.02747	0	0
65	2000	65	-0.00154	0	0	0	0	0	-0.00154	0	0	0	2000	1036.795	-0.01065	0	0
75	2000	75	-0.01673	0	0	0	0	0	-0.01673	0	0	0	2000	1319.147	-8.27E-05	0	0
25	2500	25	0.368554	0	0	0	0	0	0.368554	0	0	0	2500	325.6552	-0.26798	0	0
35	2500	35	0.158299	0	0	0	0	0	0.158299	0	0	0	2500	424.8409	-0.12179	0	0
45	2500	45	0.071233	0	0	0	0	0	0.071233	0	0	0	2500	574.5603	-0.06125	0	0
55	2500	55	0.027051	0	0	0	0	0	0.027051	0	0	0	2500	765.125	-0.03053	0	0
65	2500	65	0.001608	0	0	0	0	0	0.001608	0	0	0	2500	994.8428	-0.01284	0	0
75	2500	75	-0.01437	0	0	0	0	0	-0.01437	0	0	0	2500	1263.255	-0.00173	0	0

All units are SI.

After individual controllers are built for every grid point, these are merged into a single controller, as described above. Figure 5 illustrates the response of the composite controller for $v = 60$ m/s and $Z_e = 1200$ m. This is not a grid point, so a controller was not designed for this operating condition *per se*, but a smooth and satisfactory performance is obtained by weighing the outputs of four controllers at (55 m/s, 1000 m), (55 m/s, 1500 m), (65 m/s, 1000 m) and (65 m/s, 1500 m). The diagonals converge to unit value in about 2.5 s and the off-diagonals are almost identically zero. Similar results were obtained for all operating conditions within the flight envelope $25 < v < 75$ and $0 < z_e < 2500$.

Finally, an outer PID controller is wrapped around to generate an appropriate roll angle command ϕ_{cmd} from a desired azimuth angle χ_{cmd} . This SISO controller is easy to design empirically, as the inner-loop is well-behaved thanks to the scheduled smooth loop-shaping controller. The PID

coefficients were chosen as $K_p = 4$, $K_i = 1$, $K_d = 1$ and $N = 100$ after a few iterations on the numerical model.

For verification, SIL simulations are performed with the flight controller implemented in MATLAB/Simulink and the blade element simulations carried out by the flight simulator X-Plane. The former sends thrust, elevator, aileron, rudder commands to the latter and receives flight simulation results in real-time every 25 ms through user datagram protocol (UDP) packets (Figure 6). Numerous scenarios were studied in this configuration and one is presented here as an example. The aircraft cruising at 65 m/s and 1000 m is given appropriate commands v_{cmd} , z_e , A_y , χ_{cmd} and χ_{cmd} (red dotted lines in Figure 7) for manoeuvres required to navigate to a target airfield, line-up with the runway, descend, flare and touchdown. A_y is always zero for turn coordination and flight comfort. The bank angle reference ϕ_{cmd} generated by the outer controller for the inner controller is also shown. For the first 500 s,

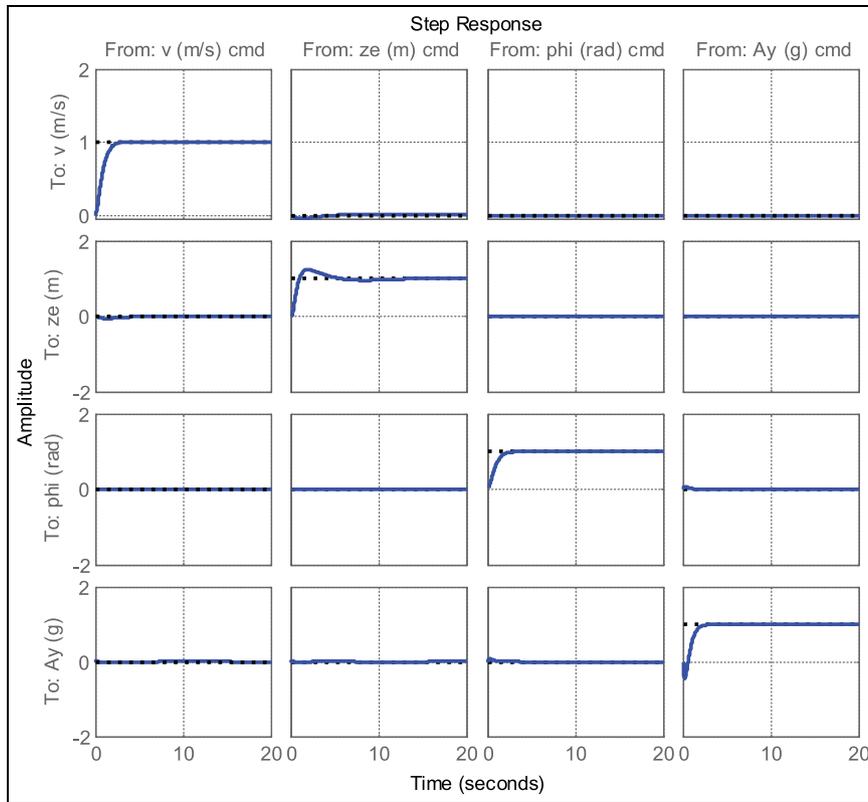


Figure 5. Aircraft states resulting from simulations of the aircraft model in closed-loop for flight around $v=60$ m/s and $Z_e=1200$ m. The response is well damped and settles in about 2.5 s. Note also that the non-diagonal responses are effectively zero, suggesting successful decoupling among diverse channels. A similar performance is obtained for all velocities and elevations within the flight envelope $25 < v < 75$ and $0 < z_e < 2500$.

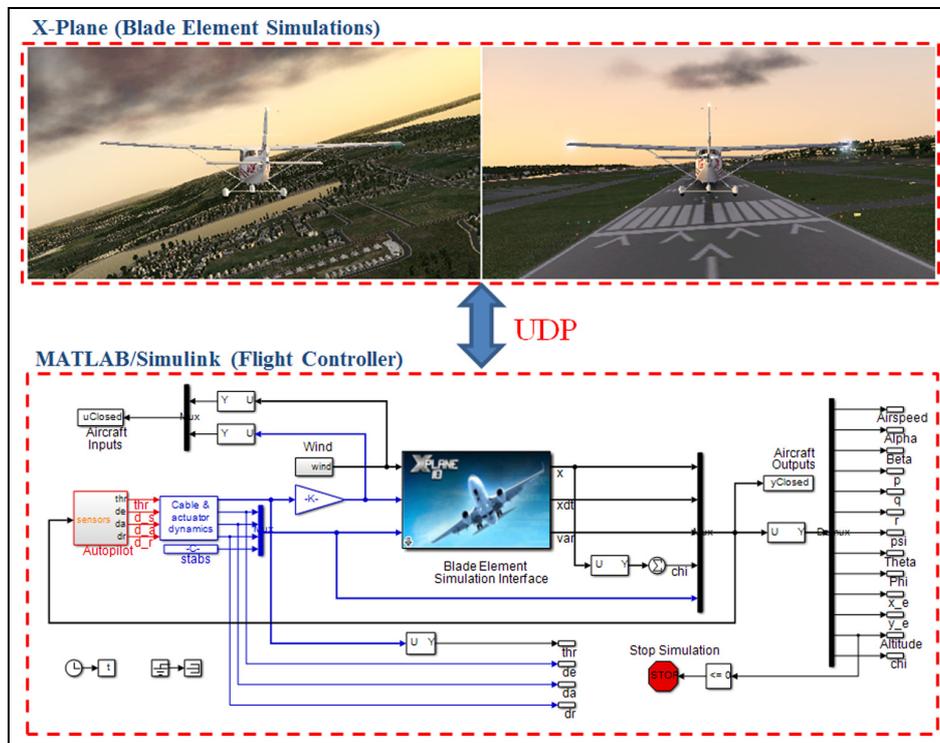


Figure 6. Software-in-the-loop (SIL) simulations with the flight control system in MATLAB/Simulink interfaced with blade element simulations in X-Plane. Real-time data exchange is performed through UDP packets.

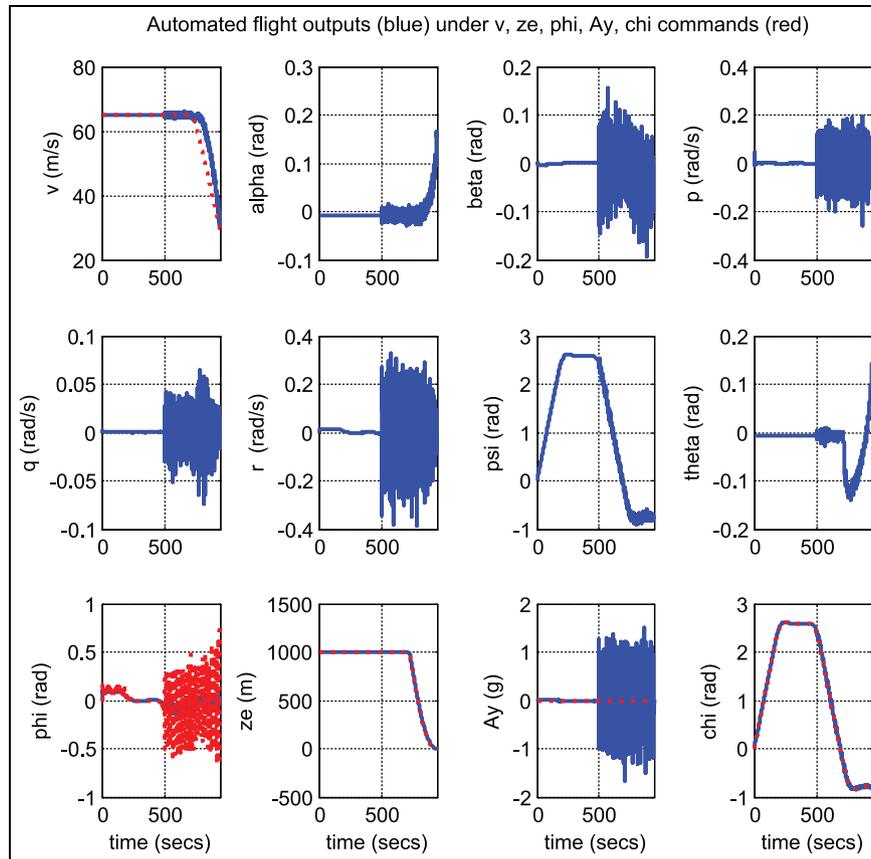


Figure 7. Aircraft states resulting from software-in-the-loop (SIL) simulations in closed-loop utilizing the proposed control strategy. The references are also shown in the top plot as red dotted lines. Strong winds up to 15 m/s are present for $t > 500$ s, including the landing phase.

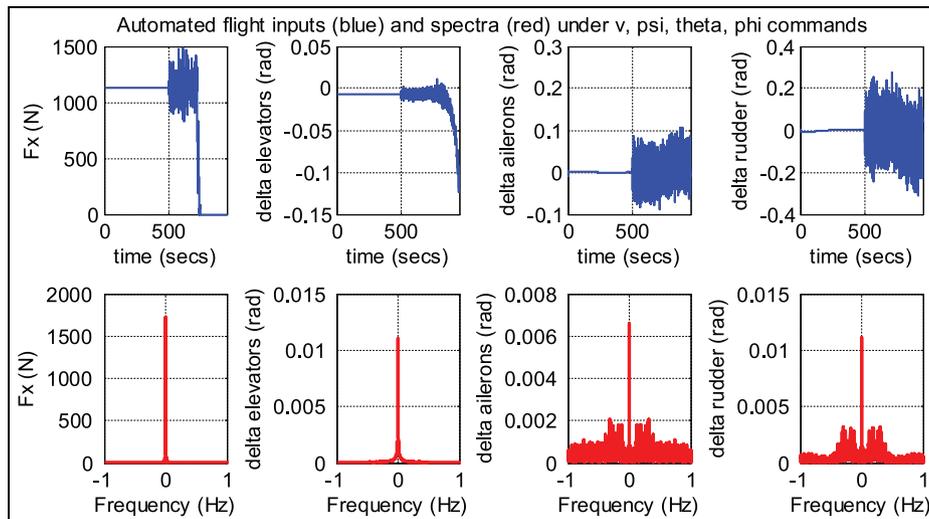


Figure 8. Aircraft inputs (top row) and amplitude spectra (bottom row) from software-in-the-loop (SIL) simulations in closed-loop utilizing the proposed control strategy. Strong winds up to 15 m/s are present for $t > 500$ s, including the landing phase.

the weather remains calm but after $t = 500$ s, the aircraft is subjected to high winds of magnitude up to 15 m/s with the crosswind component at times reaching 7 m/s. Oscillations in the states are unavoidable for such high winds; still the flight

control system maintains a stable attitude and executes the landing. The control inputs applied to the aircraft and their frequency spectra are presented in Figure 8. The thrust and surface deflections remain within reasonable limits at all times

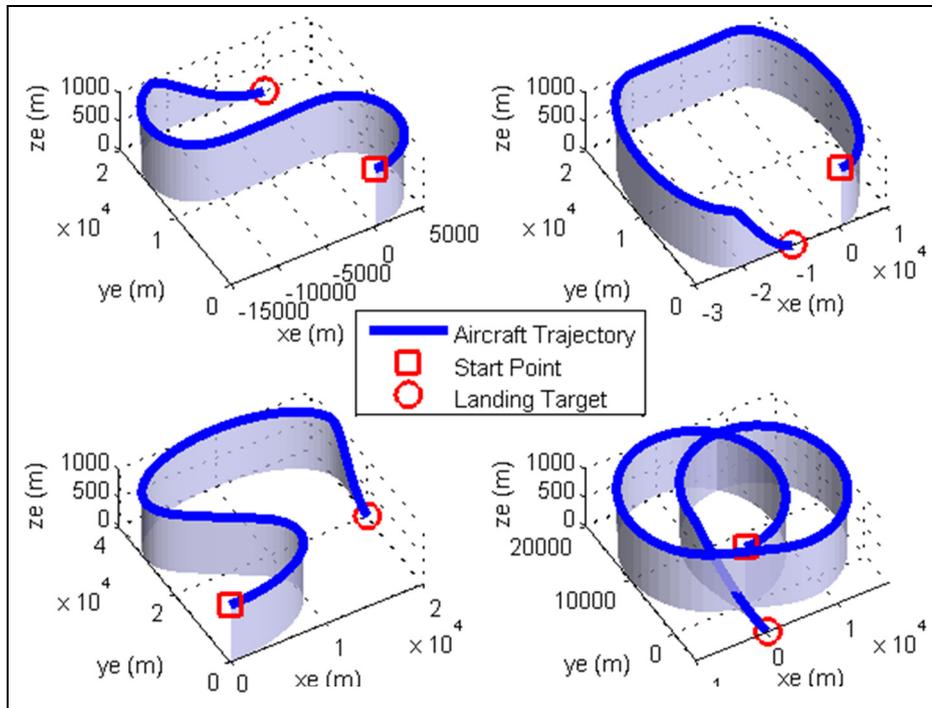


Figure 9. Aircraft trajectories resulting from software-in-the-loop (SIL) simulations in closed-loop utilizing the proposed control strategy. Top left subfigure corresponds to the states in Figure 7. Trajectories for three different scenarios are presented in the other subfigures. For all cases, strong winds up to 15 m/s are present for $t > 500$ s, including the landing phase.

and the control inputs do not contain significant power at frequencies higher than about 0.5 Hz. This ensures that the control does not cause any sharp thrust changes or wild oscillations in control surfaces. The 3D trajectory for this scenario is also visualized in the top left subfigure of Figure 9. Additional scenarios were also tested with successful outcomes and trajectories for three of these are shown in the other subfigures of Figure 9. Overall, the flight control system maintains stable flight and responds well to commands received, even under unfavourable atmospheric conditions.

To illustrate the improvement provided by the proposed approach, we also present a comparison with a standard autopilot design approach. Independent flight channels are assumed and a SISO controller is built for each one around the operating point for $v = 65$ m/s and $z_e = 1000$ m. These are then used in tandem during the flight (Blakelock, 1991; Etkin and Reid, 1996; Stevens and Lewis, 2003). The natural choice is to regulate airspeed v with thrust F_x , altitude z_e with elevator deflection δ_e , bank angle ϕ with aileron deflection δ_a and the lateral acceleration A_y with rudder deflection δ_r . A PID controller is designed for each of these tasks using either the Ziegler–Nichols or the Skogestad IMC automatic tuning method, which is followed by empirical coefficient adjustment for fine tuning (Astrom and Haggund, 2006). After subsequent trials, the PID controllers that result in the best response characteristics are merged and integrated with the outer controller. This overall controller is then given necessary reference commands to trace the same trajectory as in the example above, i.e. the top left subfigure of Figure 9. The

aircraft states resulting from SIL simulations are shown in Figure 10. Comparing these with the results obtained from the proposed scheduled smooth MIMO robust control approach (Figure 7), one notes that there are more oscillations in many of the states, including the outputs being controlled. These effects are especially prominent in the landing phase, as the aircraft is no longer in the vicinity of the operating point; note especially the trouble in following the velocity reference. Also visible in Figure 10 are the inter-channel coupling effects, which have not been accounted for explicitly. Despite the velocity and altitude references remaining constant for the navigation phase of the flight, there are still some oscillations (e.g. at $t \sim 50$ s) on the velocity v and altitude z_e , caused by the varying roll command ϕ_{cmd} . There is also some error in following the height reference during the landing phase, which is illustrated clearer by the 3D trajectory plot in Figure 11. Also superimposed on this figure for comparison is the trajectory for the flight control system designed using the proposed approach. The classical design performs acceptably during the midportion where the aircraft flies straight at $v = 65$ m/s and $z_e = 1000$ m, which is precisely the operating point for the control design. Note, however, the deviation from the desired path during the initial turn, which can be linked to coupling effect resulting from the turn command. Observe also the poor performance during landing. This is due to leaving the vicinity of the operating point and the presence of strong wind disturbances, possibly exceeding the maximum crosswind component (Table 1). The landing target is missed considerably as a consequence. In contrast, the trajectory obtained for the proposed approach is smooth,

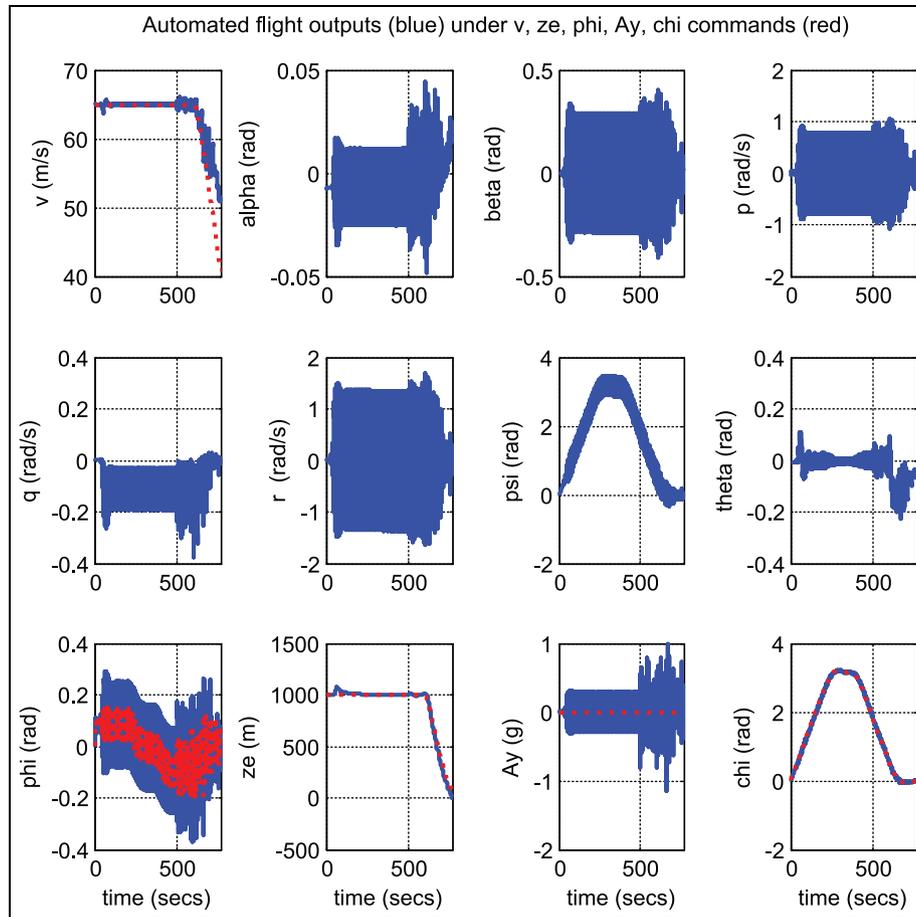


Figure 10. Aircraft states resulting from software-in-the-loop (SIL) simulations in closed-loop utilizing a classical autopilot design strategy. Separate PID controllers were built for each flight channel around a single operating point ($v=65$ m/s and $z_e=1000$ m). The references are also shown in the top plot as red dotted lines. Strong winds up to 15 m/s are present for $t>500$ s, including the landing phase.

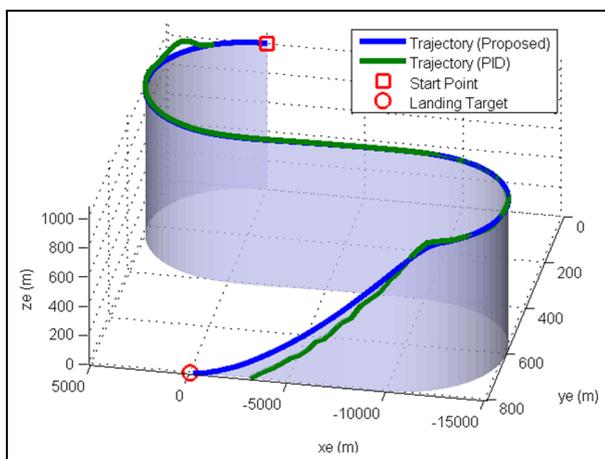


Figure 11. Aircraft trajectories resulting from software-in-the-loop (SIL) simulations in closed-loop under the proposed strategy (blue) and under a classical strategy of separate PID controllers around a single operating point (green). Strong winds up to 15 m/s are present for $t>500$ s, including the landing phase.

the coupling effects are eliminated and the landing target is hit successfully, even under the presence of high winds.

Conclusions and future works

This paper illustrates a MIMO robust control methodology where several scheduled designs are smoothly joined to produce a wide-envelope autopilot. The structure is validated using SIL tests with accurate flight simulations based upon BET. An inner-loop attitude controller balances performance and robustness, accomplishing a rapid response, low overshoot, solid noise rejection and small inter-channel coupling. Several inner controllers are blended smoothly to expand the flight envelope while evading destabilizing discontinuities. Blade element SIL testing reveals that the flight control system maintains stable flight and tracks references properly, even under challenging atmospheric conditions. The proposed approach is also compared with the classical individual channel SISO autopilot design approach and is seen to be superior, especially in eliminating inter-channel couplings and executing the landing under strong winds.

Currently work is in progress for establishing hardware-in-the-loop (HIL) and flight test platforms to validate the outlined design strategy (Atlas et al., 2015; Ertin et al., 2013; Kasnakoglu and Kaynak, 2010; Kaynak et al., 2010; Korkmaz et al., 2013). Future research directions also include studying alternative methods for the controllers and investigating the possibility of reducing controller size.

Acknowledgements

The author thanks the Scientific and Technological Research Council of Turkey (TUBITAK) for funding this research under project 113E581, as well as the members of the TOBB ETU UAV Laboratory for their valuable contributions.

Conflict of interest

The author declares that there is no conflict of interest.

Funding

The author disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by The Scientific and Technological Research Council of Turkey (TUBITAK) [113E581].

References

- Adams RJ, Buffington JM, Sparks AG, et al. (2012) *Robust Multivariable Flight Control*. London: Springer Science & Business Media.
- Ahmad SM, Chipperfield AJ and Tokhi MO (2001) Parametric modelling and dynamic characterization of a two-degree-of-freedom twin-rotor multi-input multi-output system. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 215(2): 63–78.
- Ahmadian N, Khosravi A and Sarhadi P (2015) Multiple model adaptive control using second level adaptation for a delta-wing aircraft. *International Journal of Intelligent Unmanned Systems* 3(1): 2–17.
- Aouf N, Bates DG, Postlethwaite I, et al. (2002) Scheduling schemes for an integrated flight and propulsion control system. *Control Engineering Practice* 10(7): 685–696.
- Astrom KJ and Hagglund T (2006) *Advanced PID control*. Research Triangle Park, NC: ISA – The Instrumentation, Systems, and Automation Society.
- Atlas E, Erdoğan Mİ, Ertin OB, et al. (2015) Hardware-in-the-loop test platform design for UAV applications. *Applied Mechanics and Materials, Manufacturing Science and Technology* (VI): 681.
- Basri MA, Husain AR and Danapalasingam KA (2014) Intelligent adaptive backstepping control for MIMO uncertain non-linear quadrotor helicopter systems. *Transactions of the Institute of Measurement and Control* 1(1): 0142331214538900.
- Blakelock JH (1991) *Automatic Control of Aircraft and Missiles*. New York: John Wiley & Sons.
- Boyd S and Vandenberghe L (2004) *Convex Optimization*. Cambridge: Cambridge University Press.
- Bruyere L, White BA and Tsourdos A (2003) Dynamic inversion for missile lateral velocity control via polynomial eigenstructure assignment. In: *AIAA Guidance, Navigation and Control Conference and Exhibit*, 11–14 August, Austin, Texas.
- Burton T, Sharpe D, Jenkins N, et al. (2001) *Wind Energy Handbook*. Chichester: John Wiley & Sons.
- Calise AJ, Lee S and Sharma M (2001) Development of a reconfigurable flight control law for tailless aircraft. *Journal of Guidance, Control, and Dynamics* 24(5): 896–902.
- Chen M and Desoer C (1982) Necessary and sufficient condition for robust stability of linear distributed feedback systems. *International Journal of Control* 35(2): 255–267.
- Choi B, Kang S, Kim HJ, et al. (2012) Roll–pitch–yaw integrated μ -synthesis for high angle-of-attack missiles. *Aerospace Science and Technology* 23(1): 270–279.
- de Voogt JA and van Doorn RR (2009) Accidents associated with aerobatic maneuvers in US aviation. *Aviation, Space, and Environmental Medicine* 80(8): 732–733.
- Du R, Meng K, Zhou D, et al. (2015) Design of three-dimensional nonlinear guidance law with bounded acceleration command. *Aerospace Science and Technology* 46: 168–175.
- Efe MÖ (2010) Integral sliding mode control of a quadrotor with fractional order reaching dynamics. *Transactions of the Institute of Measurement and Control* 33(8): 985–1003.
- Ertin OB, Korkmaz H, Kaynak U, et al. (2013) Hardware-in-the-loop test platform for a small fixed wing unmanned aerial vehicle embedded controller. In: *The 32nd IASTED International Conference on Modelling, Identification and Control (MIC 2013)*, Innsbruck, Austria.
- Etkin B and Reid LD (1996) *Dynamics of Flight: Stability and Control*. 3rd ed. New York: John Wiley & Sons.
- Fletcher R (1987) *Practical Methods of Optimization*. New York: John Wiley & Sons.
- Gill PE, Murray W, Saunders MA, et al. (1984) Procedures for optimization problems with a mixture of bounds and general linear constraints. *ACM Transactions on Mathematical Software (TOMS)* 10(3): 282–298.
- Gill SJ, Lowenberg MH, Neild SA, et al. (2015) Nonlinear dynamics of aircraft controller characteristics outside the standard flight envelope. *Journal of Guidance, Control, and Dynamics* 38(12): 2301–2308.
- Glover K and McFarlane D (1989) Robust stabilization of normalized coprime factor plant descriptions with H_∞ -bounded uncertainty. *IEEE Transactions on Automatic Control* 34(8): 821–830.
- Ho WK, Hang CC and Zhou JH (1995) Performance and gain and phase margins of well-known PI tuning formulas. *Control Systems Technology* 3(2): 245–248.
- Ingram G (2005) *Wind Turbine Blade Analysis using the Blade Element Momentum Method*. Durham: School of Engineering, Durham University.
- Kang S, Kim HJ, Lee JI, et al. (2009) Roll–pitch–yaw integrated robust autopilot design for a high angle-of-attack missile. *Journal of Guidance, Control, and Dynamics* 32(5): 1622–1628.
- Karagiannis D and Astolfi A (2010) Non-linear and adaptive flight control of autonomous aircraft using invariant manifolds. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 224(4): 403–415.
- Kasnakoglu C and Kaynak U (2010) Automatic recovery and autonomous navigation of disabled aircraft after control surface actuator jam. In: *AIAA Guidance, Navigation and Control Conference*, Toronto, Canada.
- Kaynak U, Akbaba R, Kibar A, et al. (2010) Design and manufacture of a fuel cell powered unmanned air vehicle. In: *International Unmanned Vehicles Workshop UVW2010*, Istanbul, Turkey.
- Korkmaz H, Ertin OB, Kasnakoglu C, et al. (2013) Design of a flight stabilizer system for a small fixed wing unmanned aerial vehicle using system identification. In: *IFAC Workshop on Advances in Control and Automation Theory for Transportation Applications (ACATTA 2013)*, Istanbul, Turkey, pp. 145–149.
- Lawrence DA, Frew EW and Pisano WJ (2008) Lyapunov vector fields for autonomous unmanned aircraft flight control. *Journal of Guidance, Control, and Dynamics* 31(5): 1220–1229.

- Le V and Safonov MG (1992) Rational matrix GCDs and the design of squaring-down compensators – a state-space theory. *IEEE Transactions on Automatic Control* 37(3): 384–392.
- Liu H, Li D, Kim J, et al. (2014) Real-time implementation of decoupled controllers for multirotor aircrafts. *Journal of Intelligent & Robotic Systems* 73(1–4): 197–207.
- Liu X, Liu Z, Shan J, et al. (2015) Anti-disturbance autopilot design for missile system via finite time integral sliding mode control method and nonlinear disturbance observer technique. *Transactions of the Institute of Measurement and Control* 1(1): 0142331215603793.
- Lungu R and Lungu M (2012) Adaptive control of helicopter pitch angle and velocity. *Journal of Aerospace Engineering* 27(5): 04014018.
- Lungu R and Lungu M (2015) Application of H2/H ∞ and dynamic inversion techniques to aircraft landing control. *Aerospace Science and Technology* 46: 146–158.
- Lungu R and Lungu M (2016) Adaptive flight control law based on neural networks and dynamic inversion for micro-aerial vehicles. *Neurocomputing* 199: 40–49.
- Lungu R, Lungu M and Grigorie LT (2013) Automatic control of aircraft in longitudinal plane during landing. *IEEE Transactions on Aerospace and Electronic Systems* 49(2): 1338–1350.
- Nelson RC (1998) *Flight Stability and Automatic Control*. New York: McGraw Hill.
- Özbek NS, Önkol M and Efe MÖ (2015) Feedback control strategies for quadrotor-type aerial robots: a survey. *Transactions of the Institute of Measurement and Control* 1(1): 0142331215608427.
- Rauw M (2001) *FDC 1.2 – A Simulink Toolbox for Flight Dynamics and Control Analysis*. Haarlem: Zeist.
- Rugh WJ and Shamma JS (2000) Research on gain scheduling. *Automatica* 36(10): 1401–1425.
- Schittkowski K (1986) NLPQL: a FORTRAN subroutine solving constrained nonlinear programming problems. *Annals of Operations Research* 5(2): 485–500.
- Shin JY, Balas GJ and Kaya MA (2002) Blending methodology of linear parameter varying control synthesis of F-16 aircraft system. *Journal of Guidance, Control, and Dynamics* 25(6): 1040–1048.
- Sheng S, Sun C, Duan H, et al. (2014) Longitudinal and lateral adaptive flight control design for an unmanned helicopter with coaxial rotor and ducted fan. In: *33rd Chinese Control Conference (CCC)*, pp. 130–135.
- Stevens B and Lewis F (2003) *Aircraft Control and Simulation*. 2nd ed. New York: Wiley-Interscience.
- Stilwell DJ (2001) State-space interpolation for a gain-scheduled autopilot. *Journal of Guidance, Control, and Dynamics* 24(3): 460–465.
- Tsay TS (2007) Decoupling the flight control system of a supersonic vehicle. *Aerospace Science and Technology* 11(7): 553–562.
- Wang Q and Stengel RF (2000) Robust nonlinear control of a hypersonic aircraft. *Journal of Guidance, Control, and Dynamics* 23(4): 577–585.
- Wong DK, Pitfield DE, Caves RE, et al. (2006) Quantifying and characterising aviation accident risk factors. *Journal of Air Transport Management* 12(6): 352–357.
- Yang CD and Liu WH (2003) Nonlinear H-infinity decoupling hover control of helicopter with parameter uncertainties. *American Control Conference*, pp. 3454–3459.

Appendix A: Aircraft model equations

The differential equations for the aircraft model are as follows

$$\dot{V} = \frac{1}{m} (F_x \cos \alpha \cos \beta + F_y \sin \beta + F_z \sin \alpha \sin \beta) \quad (47)$$

$$\dot{\alpha} = \frac{1}{V \cos \beta} \left[\frac{1}{m} (-F_x \sin \alpha + F_z \cos \alpha) \right] + q - (p \cos \alpha + r \sin \alpha) \tan \beta \quad (48)$$

$$\dot{\beta} = \frac{1}{V} \left[\frac{1}{m} (-F_x \cos \alpha \sin \beta + F_y \cos \beta - F_z \sin \alpha \sin \beta) \right] + p \sin \alpha - r \cos \alpha \quad (49)$$

$$\dot{p} = P_{pp} p^2 + P_{pq} pq + P_{pr} pr + P_{qq} q^2 + P_{qr} qr + P_{rr} r^2 + P_l L + P_m M + P_n N \quad (50)$$

$$\dot{q} = Q_{pp} p^2 + Q_{pq} pq + Q_{pr} pr + Q_{qq} q^2 + Q_{qr} qr + Q_{rr} r^2 + Q_l L + Q_m M + Q_n N \quad (51)$$

$$\dot{r} = R_{pp} p^2 + R_{pq} pq + R_{pr} pr + R_{qq} q^2 + R_{qr} qr + R_{rr} r^2 + R_l L + R_m M + R_n N \quad (52)$$

$$\dot{\psi} = q \sin \phi + r \cos \phi \cos \theta \quad (53)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (54)$$

$$\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta \quad (55)$$

$$\dot{x}_e = [u \cos \theta + (v \sin \phi + w \cos \phi) \sin \theta] \cos \psi - (v \cos \phi - w \sin \phi) \sin \psi \quad (56)$$

$$\dot{y}_e = [u \cos \theta + (v \sin \phi + w \cos \phi) \sin \theta] \sin \psi + (v \cos \phi - w \sin \phi) \cos \psi \quad (57)$$

$$\dot{H} = -\dot{z}_e \quad (58)$$

where

$$\dot{z}_e = -u \sin \theta + (v \sin \phi + w \cos \phi) \cos \theta \quad (59)$$

$$\dot{u} = \frac{F_x}{m} - qw + rv \quad (60)$$

$$\dot{v} = \frac{F_y}{m} + pw - ru \quad (61)$$

$$\dot{w} = \frac{F_z}{m} - pv + qu \quad (62)$$

and $P_{pp}, P_{pq}, P_{pr}, P_{qq}, P_{qr}, P_{rr}, P_l, P_m, P_n, Q_{pp}, Q_{pq}, Q_{pr}, Q_{qq}, Q_{qr}, Q_{rr}, Q_l, Q_m, Q_n, R_{pp}, R_{pq}, R_{pr}, R_{qq}, R_{qr}, R_{rr}, R_l, R_m, R_n$ are parameters dependent on the inertia values (Rauw, 2001).

Appendix B: Sequential quadratic programming

The non-linear optimization problem to obtain the trim point is solved using SQP (Fletcher, 1987; Schittkowski, 1986). Given the problem in (12)–(15), a quadratic programming (QP) subproblem based on a quadratic approximation of the Lagrangian function is formulated

$$\mathcal{L}(\bar{x}, \lambda_1, \lambda_2) = f(\bar{x}) + \lambda_1 g_1(\bar{x}) + \lambda_2 g_2(\bar{x}) \quad (63)$$

where λ_1 and λ_2 are Lagrange multipliers. At an iterate \bar{x}_k , the SQP algorithm defines an appropriate search direction d_k as a solution to the QP subproblem

$$\min_d \frac{1}{2} d^T \nabla^2 \mathcal{L}(\bar{x}_k, \lambda_{1,k}, \lambda_{2,k}) d + \nabla f(\bar{x}_k)^T d \quad (64)$$

subject to

$$\nabla g_1(\bar{x}_k)^T d + g_1(\bar{x}_k) = 0, \nabla g_2(\bar{x}_k)^T d + g_2(\bar{x}_k) = 0 \quad (65)$$

The subproblem is in standard QP form hence it can be tackled with any QP algorithm readily available in many numerical solver packages (Gill et al., 1984). The solution d_k to this problem is utilized to form a new iterate

$$\bar{x}_{k+1} = \bar{x}_k + \alpha_k d_k \quad (66)$$

where the step length α_k is obtained by an appropriate line search procedure so that an adequate decrease in the following merit function is obtained

$$\Psi(\bar{x}) = f(\bar{x}) + r_1 g_1(\bar{x}) + r_2 g_2(\bar{x}) \quad (67)$$

The penalty parameters r_1 and r_2 are constant values.

Appendix C: Loop shaping

The controller design for the inner loop attitude controller is carried out using the loop-shaping procedure. If the shaped plant is factored as

$$\mathbf{G}_s = \mathbf{M}^{-1} \mathbf{N} \quad (68)$$

then any perturbed plant can be written as

$$\mathbf{G}_\Delta = (\mathbf{M} + \Delta_{\mathbf{M}})^{-1} (\mathbf{N} + \Delta_{\mathbf{N}}) \quad (69)$$

where $\Delta_{\mathbf{M}}$ and $\Delta_{\mathbf{N}}$ are stable and unknown transfer functions that represent uncertainties in the nominal plant. The objective of the robust controller design is to stabilize by a

controller \mathbf{K} , not only the nominal plant but also the family of perturbed plants defined as

$$\mathbf{G}_\varepsilon = \left\{ (\mathbf{M} + \Delta_{\mathbf{M}})^{-1} (\mathbf{N} + \Delta_{\mathbf{N}}) : \|\Delta_{\mathbf{M}}, \Delta_{\mathbf{N}}\|_\infty < \varepsilon \right\} \quad (70)$$

For robust stability, internal stability must be achieved for the nominal and perturbed plant. If there exist a \mathbf{K} such that $(\mathbf{M}, \mathbf{N}, \mathbf{K}, \varepsilon)$ is robustly stable, then $(\mathbf{M}, \mathbf{N}, \varepsilon)$ is said to be *robustly stabilizable* with stability margin ε (Chen and Desoer, 1982). For robust stability, the following must be satisfied

$$(\mathbf{I} - \mathbf{G}\mathbf{K})^{-1}, \mathbf{K}(\mathbf{I} - \mathbf{G}\mathbf{K})^{-1}, (\mathbf{I} - \mathbf{G}\mathbf{K})^{-1} \mathbf{G}, (\mathbf{I} - \mathbf{K}\mathbf{G})^{-1} \in \mathbf{RH}_\infty, \det(\mathbf{I} - \mathbf{G}\mathbf{K})(\infty) \neq 0 \quad (71)$$

$$\inf_{\mathbf{K}} \left\| \begin{bmatrix} \mathbf{K}(\mathbf{I} - \mathbf{G}\mathbf{K})^{-1} \mathbf{M}^{-1} \\ (\mathbf{I} - \mathbf{G}\mathbf{K})^{-1} \mathbf{M}^{-1} \end{bmatrix} \right\|_\infty \leq \varepsilon^{-1} \quad (72)$$

where the infimum is taken over all stabilizing controllers. The H_∞ optimization problem allows ε^{-1} to be chosen as small as possible. For actual implementation, the robust stabilization problem can be converted to a more suitable formulation. Let

$$\mathbf{P} \triangleq \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} 0 \\ \mathbf{M}^{-1} \\ \mathbf{M}^{-1} \end{pmatrix} & \begin{pmatrix} \mathbf{I} \\ \mathbf{G} \\ \mathbf{G} \end{pmatrix} \end{bmatrix} \quad (73)$$

$$\mathcal{F}_L(\mathbf{P}, \mathbf{K}) \triangleq \mathbf{P}_{11} + \mathbf{P}_{12} \mathbf{K} (\mathbf{I} - \mathbf{P}_{22} \mathbf{K})^{-1} \mathbf{P}_{21} \quad (74)$$

Then Equation (72) can be seen to be equivalent to

$$\inf_{\mathbf{K}} \|\mathcal{F}_L(\mathbf{P}, \mathbf{K})\|_\infty \leq \varepsilon^{-1} \quad (75)$$

where \mathbf{K} is gain chosen over all stabilizing controllers and \mathbf{P} is a plant of standard form for H_∞ optimization problem (Glover and McFarlane, 1989). The final controller to be used is then computed as

$$\mathbf{K}_{\text{final}} = \mathbf{W}\mathbf{K} \quad (76)$$