Reliability Assessment for Low-cost Unmanned Aerial Vehicles

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Paul Michael Freeman

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Gary J. Balas, co-advisor
Peter J. Seiler, co-advisor

November 2014

Acknowledgements

Personal: TBD

This material is based upon work supported by the National Science Foundation under Grant No. 0931931 entitled *CPS: Embedded Fault Detection for Low-Cost, Safety- Critical Systems*; Dr. Ted Baker, Program Manager. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

Dedication

TBD

Abstract

Existing low-cost unmanned aerospace systems are unreliable, and engineers must blend reliability analysis with fault-tolerant control in novel ways. This dissertation introduces the University of Minnesota unmanned aerial vehicle flight research platform, a comprehensive simulation and flight test facility for reliability and fault-tolerance research. An industry-standard reliability assessment technique, the *failure modes and effects analysis*, is performed for an unmanned aircraft. Particular attention is afforded to the control surface and servo-actuation subsystem. Maintaining effector health is essential for safe flight; failures may lead to loss of control incidents. Failure likelihood, severity, and risk are qualitatively assessed for several effector failure modes. Design changes are recommended to improve aircraft reliability based on this analysis. Most notably, the control surfaces are split, providing independent actuation and dual-redundancy. The simulation models for control surface aerodynamic effects are updated to reflect the split surfaces using a first-principles geometric analysis.

The failure modes and effects analysis is extended by using a high-fidelity nonlinear aircraft simulation. A trim state discovery is performed to identify the achievable steady, wings-level flight envelope of the healthy and damaged vehicle. Tolerance of elevator actuator failures is studied using familiar tools from linear systems analysis. This analysis reveals significant inherent performance limitations for candidate adaptive/reconfigurable control algorithms used for the vehicle. Moreover, it demonstrates how these tools can be applied in a design feedback loop to make safety-critical unmanned systems more reliable.

Control surface impairments that do occur must be quickly and accurately detected. This dissertation also considers fault detection and identification for an unmanned aerial vehicle using model-based and model-free approaches and applies those algorithms to experimental faulted and unfaulted flight test data. Flight tests are conducted with actuator faults that affect the plant input and sensor faults that affect the vehicle state measurements. A model-based detection strategy is designed and uses robust linear filtering methods to reject exogenous disturbances, e.g. wind, while providing robustness to model variation. A data-driven algorithm is developed to operate exclusively on raw flight test data without physical model knowledge. The fault detection and identification performance of these complementary but different methods is compared. Together, enhanced reliability assessment and multi-pronged fault detection and identification techniques can help to bring about the next generation of reliable low-cost unmanned aircraft.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Stringent safety requirements have driven aircraft system design for decades. Aircraft manufacturers typically achieve high levels of safety via hardware redundancy throughout the flight control system. For example, the Boeing 777 employs triple redundancy for all hardware, including the flight computer system, electrical system, hydraulic actuation, and communication channels [1, 2] (Fig. 1.1). Pervasive hardware redundancy strategies can prevent or substantially mitigate loss of control (LOC) incidents. LOC incidents are the leading cause of aviation accidents, and system faults are the leading cause of in-flight LOC incidents [3, 4]. The aerospace industry additionally pioneered the use of now-standard reliability assessment tools, including failure modes and effects analysis (FMEA) and fault tree analysis (FTA). These tools are used in conjunction with hardware redundancy to achieve safety. The vehicle systems are designed to ensure that potential failure mode paths are mitigated prior to catastrophic failure by relying on backup (redundant) hardware whenever necessary.

Low-cost unmanned air systems (UAS) cannot typically accommodate pervasive hardware redundancy due to size, weight, power, and budget constraints. Much of the applications envisioned for low-cost UAS rely on several key characteristics such as manageable size, long endurance, and minimal maintenance. Hence, certifiable techniques to address UAS LOC incidents must, whenever feasible, rely on analytical over hardware redundancy to maintain the operational advantages of UAS. Advances are particularly important to the future of unmanned aviation, as the US Federal Aviation Administration has been required to safely integrate UAVs into the national airspace by 2015 [5]. To do so will require reliability certification procedures akin to those used for manned aviation. UAS certification tool development will require blending conventional reliability analysis, robust or adaptive control tools, and fault detection and isolation techniques. Existing reliability analysis tools are effective for hardware-redundant

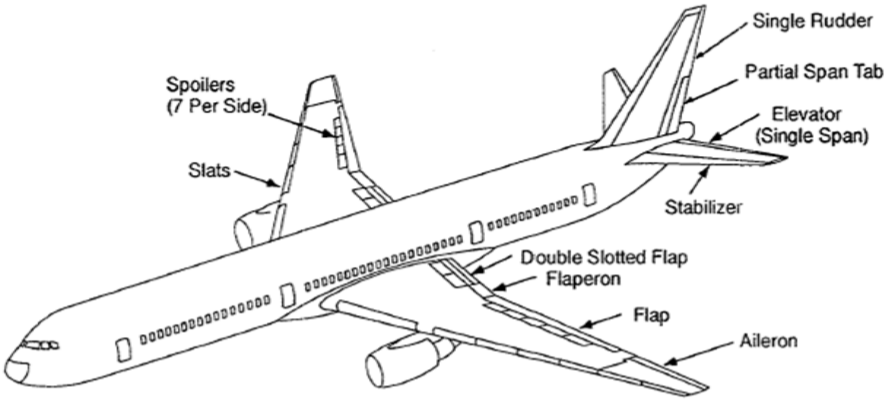


Figure 1.1: Boeing 777 flight control surfaces. The underlying computation, electrical, and hyraulic subsystems are triple redundant.

systems, yet are rarely applied to low-cost systems. One reason is that these traditional tools do not necessarily scale well to low-cost UAS without hardware redundancy.

This dissertation details an industry-standard failure modes and effects analysis for a small, low-cost unmanned aerial vehicle and The FMEA preliminarily assesses system vulnerabilities in a structured manner. The FMEA is extended by analyzing the effects of a single elevator actuator failure mode. A trim state discovery is performed to determine the achievable steady, wings-level flight envelope for the faulted and unfaulted vehicle. Linear analysis of the system dynamics reveals how the fault-tolerance and control authority of vehicle effectors changes throughout the flight envelope. Much existing literature regarding LOC trim state discovery focuses on non-UAS applications or places a greater emphasis on path-planning than on the overall system reliability [6, 7]. Other aircraft fault-tolerance works emphasize online (i.e., in-flight) parameter estimation for reconfiguration purposes [8, 9]. While both approaches have their utility, particularly for the design and validation of reconfiguration algorithms, they are not as useful for obtaining a high-level assessment of a vehicle's fault tolerance throughout a flight envelope. This research seeks to bridge the work of the guidance/navigation/control and health management communities by incorporating dynamics and controls analysis into the system reliability analysis. This detailed examination yields rich insight regarding UAV fault tolerance and raises significant questions about the feasibility of adaptive or reconfigurable control algorithms for low-cost UAS. Control surface impairments that do occur must be quickly and accurately detected. This dissertation also considers fault detection and identification for an unmanned aerial vehicle using model-based and model-free approaches and applies those algorithms to experimental faulted and unfaulted flight test data. Together, enhanced reliability assessment and multi-pronged fault detection and identification techniques can help to bring about the next generation of reliable low-cost unmanned aircraft. The dissertation is largely based on three major publications in these areas [10, 11, 12].

The dissertation is organized as follows. Chapter 2 introduces the University of Minnesota unmanned aerial vehicle flight research platform, a comprehensive, open-source flight test and simulation resource for reliability and guidance, navigation, and control research. The research described in this dissertation is based on aircraft and high-fidelity aircraft simulations within the flight research platform. Chapter 3 introduces industry-standard reliability tools such as fault tree analysis and failure modes and effects analysis. A thorough failure modes and effects analysis is conducted for a flight research platform aircraft, with particular attention paid to the failures affecting the control surfaces and actuators. Chapter 4 details reliability-focused design changes to the unmanned aircraft based upon the results of the failure modes and effects analysis. The changes to the design required updates to the simulation model, and the modeling changes and validation are presented in this chapter. Chapter 5 describes the trim state discovery procedure and shows the flight envelopes for flight research platform vehicles with different control surface architectures. A linear model set is obtained for each operating point, enabling linear analysis of how the dynamics vary throughout the flight envelope Chapter 6 details the model-based and data-driven fault detection and isolation module designs and results. Concluding remarks are provided in Chapter 7.

This dissertation contributes to the UAS reliability literature in three main ways. First, the full application of conventional FMEA to the aircraft provides important information for further development of a reliability-focused UAS platform within the UMN flight research platform architecture. These FMEA results inform design changes to both the airframe and the simulation used for model-based design and analysis. Additionally, the limitations of the traditional tools are better understood, opening the door to more rigorous analysis of failure modes and effects. Second, the trim state discovery tools and corresponding linear systems analysis extend the FMEA based on ideas from the dynamics and controls field. This demonstrates how the guidance, navigation, and control community and reliability engineering community can more closely integrate their efforts as applied to the emerging field of small UAS and obtain further insight regarding the vehicle flight dynamics in healthy and faulted conditions. This extension of the FMEA provides additional design feedback which can be used to improve vehicle safety and aid in the certification process. Finally, model-based and data-driven fault detection and isolation modules are developed using the flight research platform simulation and real vehicle flight test data to estimate control surface impairments

Chapter 2

UMN Flight Research Platform

The Department of Aerospace Engineering and Mechanics at the University of Minnesota has developed and maintained a comprehensive flight research platform (FRP) over the past decade. This chapter describes the FRP, which is used to support research activities including guidance, navigation, and control (GNC) algorithms, embedded fault detection methods, and system identification tools. The department originally sought to develop a high-fidelity, reliable platform to support algorithm design, proof of concept, and experimental demonstration in real operating environments. Such a platform is essential to successfully apply concepts from the theoretical domain to functional aerospace applications. Broadly, the FRP has two modules: the experimental flight test hardware and a software and simulation package. Each of these modules are now described in detail. [13, 14, 15]

2.1 Experimental Flight Test Hardware

This section describes the FRP airframes and R/C components, avionics package, sensor loadout, and the ground control station used for flight testing.

2.1.1 Airframes and R/C Components

The flight test hardware consists of several aircraft of different sizes carrying different avionics and and sensor payloads. The selected airframes are all commercial, off-the-shelf radio controlled aircraft purchased by the department. This was done to build a fleet of consistent airframes while minimizing overhead costs and extraneous design tasks. The FRP has three variants of the *Ultra Stick* fixed-wing electric aircraft. All of the FRP aircraft use conventional control surface configurations comprising a single vertical rudder with elevator, aileron, and

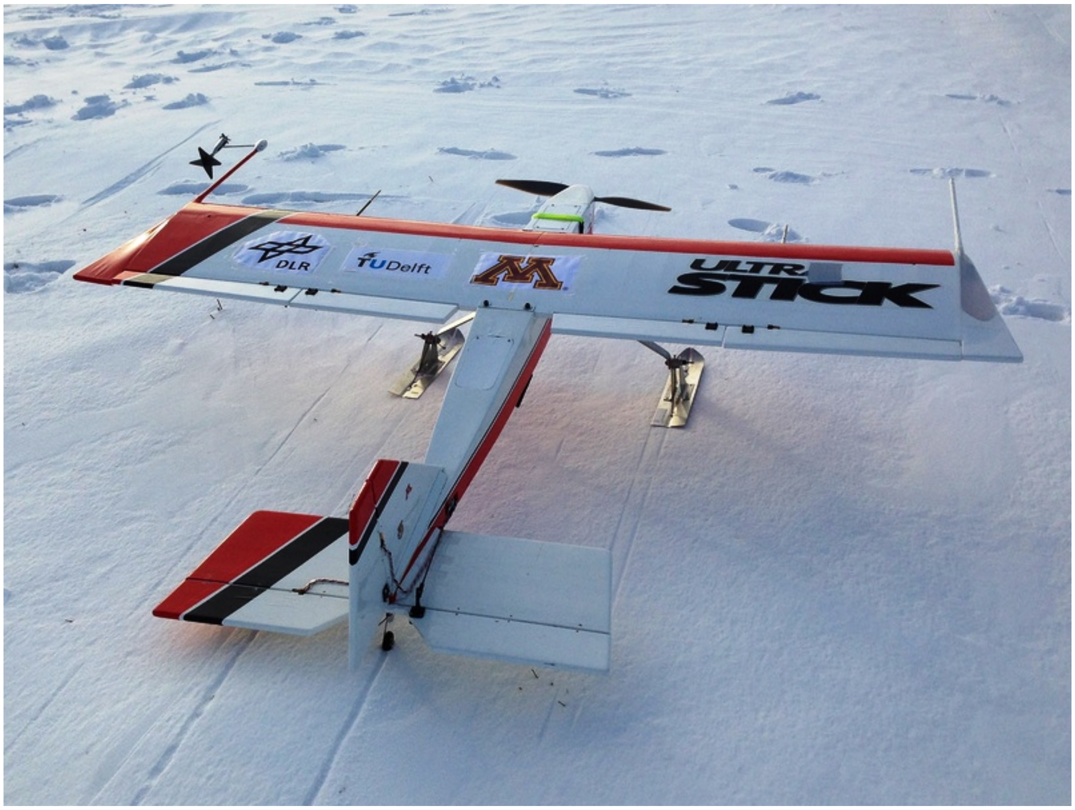


Figure 2.1: *Ultra Stick 120 'Ibis'* UAV at flight testing location. The landing gear wheels can be swapped with a pair of skis for flight testing during the winter months.

flap surfaces. Each surface is independently actuated by a single electric, hobby-grade servo with deflection limits of ±25 deg.

The *Ultra Stick 120* is the largest of the FRP vehicles. It has a symmetrical airfoil wing, spanning 1.92 m, and mass of 7.4 kg. Before being given to UMN, this aircraft was originally used at NASA Langley Research Center where it was called *FASER* (Free-flying Aircraft for Subscale Experimental Research) [16]. Capable of carrying over 2.5 kg of payload (the largest of the three FRP airframes), the *Ultra Stick 120* can employ a wide array of sensors for experimental data collection during flight tests. For *Ultra Stick 120*, all control surfaces are actuated by Hitec HS5625MG servos. The plane has a 1900W Actro 40-4 brushless electric motor with a Graupner 14 x 9.5 folding propeller. The motors is powered by two 5000mAh 5-cell lithium polymer batteries connected in serie, and the servos are powered by a separate 1350 mAh 3-cell lithium polymer battery. The primary *Ultra Stick 120* used for flight testing is named *Ibis*, and it is shown in Figures 2.1-2.2. Much of the research presented in this dissertation is based on *Ibis*.

Figure 2.2: *Ultra Stick 120* UAV with *Ultra Stick 25e* models. The department maintains several of the smaller *Ultra Stick 25e* test UAVs.

The *Ultra Stick 25e* is an approximately 66 % scale model of the *Ultra Stick 120*. It is conveniently sized, with a 1.27 m wingspan and 1.9 kg mass, and can accommodate the avionics payloads essential for most control algorithm flight testing. All control surfaces are actuated by Hitec HS-225BB servos. The plane has a 600W E-Flite Power 25 brushless electric motor with an APC 12 x 6 propeller. A 4200mAh 3-cell lithium polymer battery powers the motor and servos. Figure 2.2 shows the UAVs and their relative size.

The *Mini Ultra Stick* is an approximately 50% scale model of the *Ultra Stick 120*, with the same basic configuration. This aircraft is primarily used as as a wind tunnel model as it is the only one of the three airframes that fits inside the UMN Low-Speed Wind Tunnel. Because *Mini Ultra Stick* is an approximately scaled variant of the larger airframes, aerodynamic data collected from wind tunnel experiments can provide insight into the expected behavior of the other FRP vehicles. The aircraft is not equipped with the avionics necessary for autonomous operation and can only fly in a remotely piloted mode. The *Mini Ultra Stick* is shown in Figure 2.3.

2.1.2 Avionics

Each of the *Ultra Stick 120/25e* UAVs maintained by the department carry on board an avionics package that enables remote piloting and automatic piloting flight modes. This core avionics

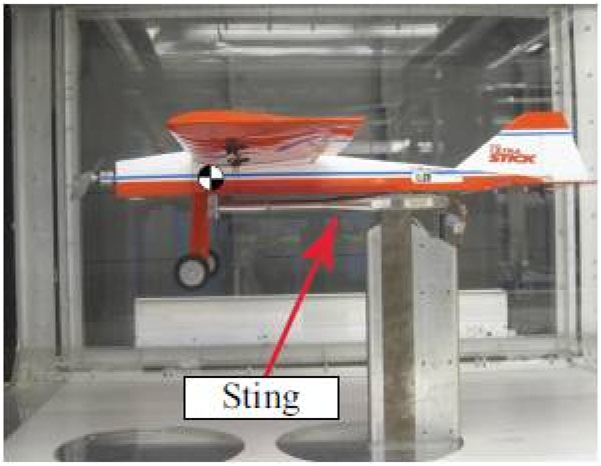


Figure 2.3: *Mini Ultra Stick* mounted on sting during experiments in UMN Low-Speed Wind Tunnel.

package is called the *Goldy Flight Control System*, and its design is standardized such that it is transferrable between all airframes within the the UMN UAV fleet.

The *Goldy* flight computer is a 32-bit PowerPC *phyCore MPC-5200B tiny* system on a module (SoM). The clock rate is 400 MHz, and the computer has 760MIPS of processing power with floating point computation. The flight computer runs the eCos real-time operating system, and the flight software is written and implemented in the C language. Flight software modules (e.g. different control laws, navigation algorithms, and fault detection filters) can be easily enabled or disabled to quickly perform flight test experiments and collect data using different modules. The flight software runs at a frame rate of 50 Hz and utilizes less than 2% of the CPU capacity.

The computer supports several input-output standards, including TTL, RS232 serial, SPI, I2C, Ethernet, and PWM. The computer acquires sensor data at 50 Hz and performs attitude and position estimation, executes flight control algorithms, logs relevant data, sends PWM servo commands to the actuators, and sends telemetry to a ground control station via a modem. A failsafe switching board is used to switch control of the aircraft between manual remote piloting (human R/C pilot) and the embedded automatic control. A high-level schematic of the avionics is shown in Figure 2.4. The hardware interface to the flight computer is handled via a custom-designed interface board (shown in Figure 2.5).

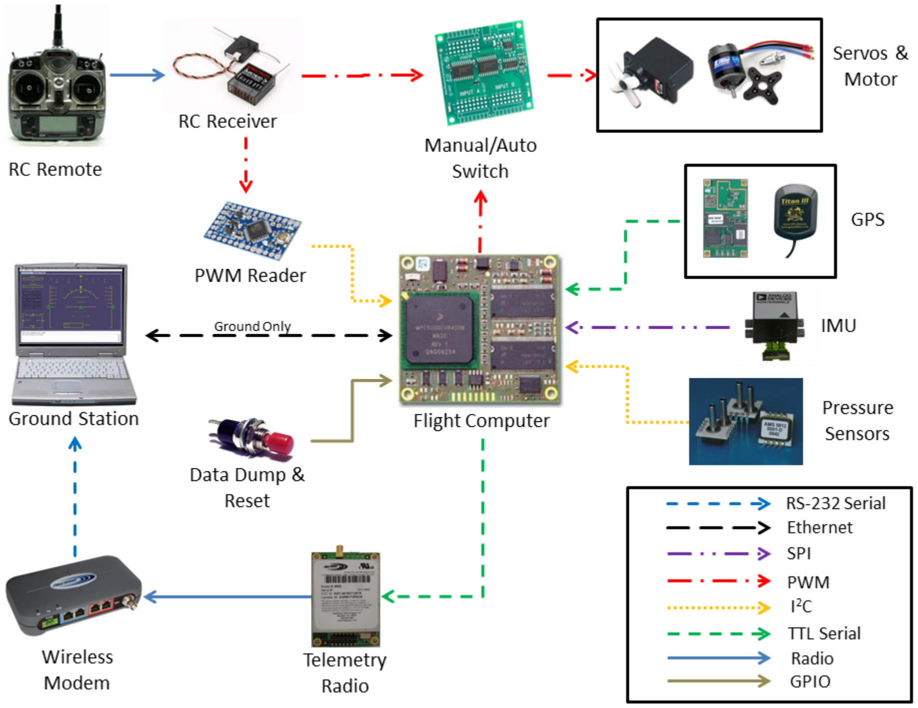


Figure 2.4: Core UAV avionics architecture.

2.1.3 Sensors

As stated previously, *Ibis* carries the largest payload, and therefore, the entire sensor suite that is available for UMN FRP vehicles. The sensor suite on *Ibis* consists of the following components:

- Inertial Measurement Unit (IMU): Analog Devices iSensor ADIS16405
- GPS: Hemisphere GPS Crescent OEM board with Titan III antenna
- Air data probe: Goodrich 0858 mini 5-hole probe
- Pressure transducers: AMSYS AMS 5812: -0150-B, -003D, -003D-B (two)
- 2 angle of attack/angle of sideslip (AOA/AOS) vanes (one per wingtip) via NASA Langley Research Center
- 8 CTS 250 series 100 kΩ rotary potentiometer deflection sensors (one per control surface, one per AOA vane)
- 2 SX8724C analog-to-digital converter (ADC) chips for potentiometers

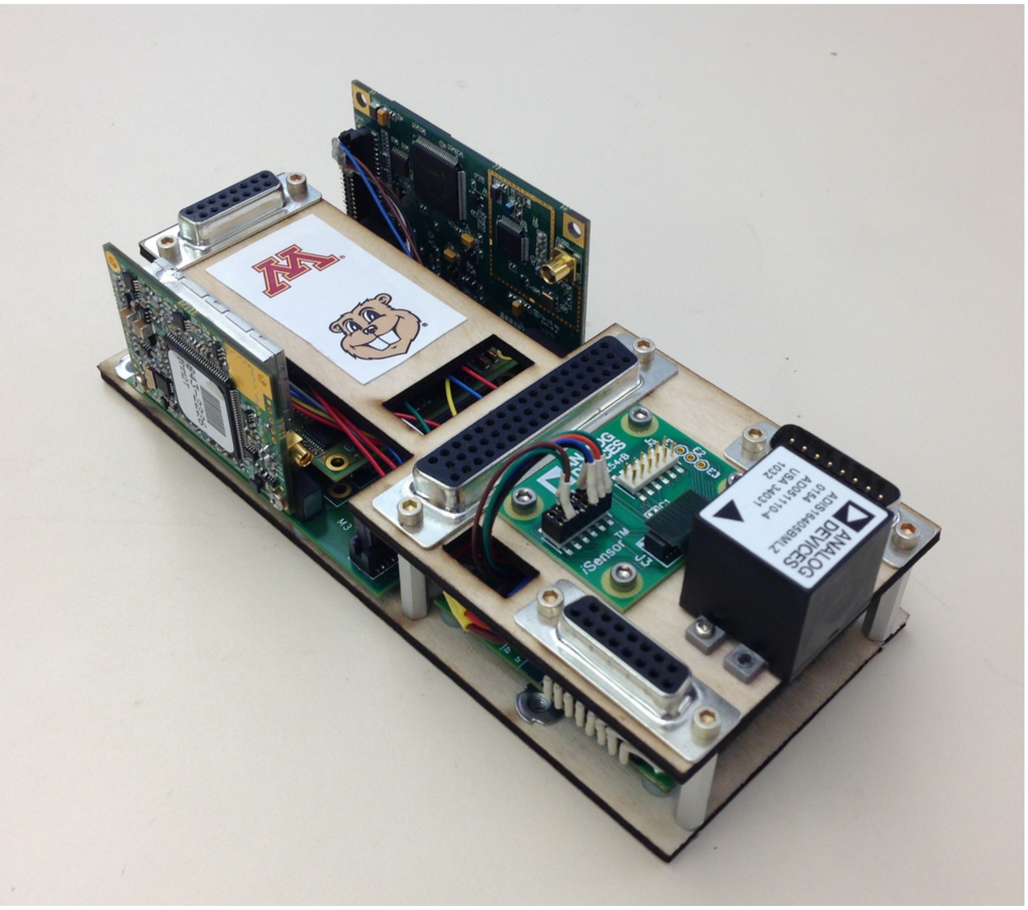


Figure 2.5: *Goldy* Flight Control System avionics pallet.

- Camera
- Camera battery
- Voltage regulator for camera

The *Ultra Stick 25e* models do not carry the wingtip AOA/AOS vanes, camera hardware or surface deflection potentiometer hardware.

2.1.4 Ground Station

The Ground Control Station (GCS) is used during flight testing to provide vital real-time flight information to assess the flight performance and maintain situational awareness. It consists of a laptop computer running the GCS software connected via serial to a data modem that receives UAV telemetry. The GCS software includes a Heads-Up Display, a moving map showing the location of the aircraft, commanded actuator positions, and indicators to display flight control mode information. The software is a Java-based program inspired by the Open Source Glass Cockpit Project.

2.2 Software and Simulation Package

The *Freyja* software and simulation infrastructure complements the flight test hardware within the UMN FRP. *Freyja* includes a nonlinear, modular simulation that allows researches to accurately model the physics of the UAVs. Additionally, *Freyja* incorporates Software-in-the-Loop (SIL) and Hardware-in-the-Loop (HIL) capabilities. These tools allow researchers to perform verification and validation testing natively within the nonlinear simulation while connected to the *Goldy* Flight Control System. The code for the flight software modules can be autogenerated from the simulation and loaded onto the flight computer for experimental flight testing. Data collected from the flight tests is used to update the simulation models when necessary.

2.2.1 Nonlinear Simulation

The core of the *Freyja* FRP package is a high-fidelity, six degree-of-freedom (DOF) nonlinear simulation of FRP aircraft dynamics. The FRP nonlinear simulation is a comprehensive prototyping platform and incorporates the following submodels:

- Instrumented airframe mass and inertia properties

- Propulsion (motor and propeller) dynamics
- Servo-actuator dynamics
- Computational time delays
- Identified aerodynamic force and moment coefficients (*Ultra Stick 25e*) or lookup tables (*Ultra Stick 120*)
- Sensor noise properties
- Environmental effects (e.g., standard atmosphere, winds, turbulence, gravity, magnetic field)

The simulation model computes the net forces and moments due to aerodynamics, propulsion, and environmental conditions (e.g. atmospheric conditions, gravity, wind) and uses numerical integration to solve the rigid body equations of motion.

The simulation is modular; the physical properties unique to each FRP UAV can easily be enabled. The aerodynamic models were derived and are implemented differently for each airframe. The *Ultra Stick 120* aerodynamic model is derived from wind tunnel experiments at NASA Langley Research Center. The coefficients are provided in a set of extensive lookup tables based on those tests. Details regarding this model are provided in Chapter 4. The *Ultra Stick 25e* model was obtained at a single, nominal flight condition using open-loop flight tests with a frequency domain system identification approach. The aerodynamic model was validated for this flight condition, and the coefficients are constant. Hence, flight tests using *Ultra Stick 25e* airframes are generally conducted starting the UAV at the nominal flight condition. Finally, the *Ultra Stick Mini* aerodynamic model is entirely derived from wind tunnel data.

The emphFreyja simulation also includes automated trimming and linearization routines to obtain a linear plant model. The linear model is useful for understanding system behavior and is typically a starting point for the design of new flight control laws or fault detection algorithms. After achieving satisfactory performance with the linear plant, the algorithm can be placed in feedback with the nonlinear dynamics for verification.

2.2.2 Software-in-the-Loop Simulation

Instead of using a Simulink implementation of the flight control laws (as in the full nonlinear simulation), the SIL places the auto-generated flight control code (written in C) in feedback with the nonlinear UAV dynamics. This is done by including the flight control software as a

S-function in Simulink. The SIL resulting SIL model can be used to verify the correctness of the flight software auto-generation from Simulink (a mathematical discrete-time model) to C.

2.2.3 Hardware-in-the-Loop Simulation

The HIL module allows for nonlinear simulations with the auto-generated flight software running on the actual flight computer. The MathWorks Real Time Windows Target toolbox allows the simulation to run in real-time on departmental PCs while communicating with the flight control hardware. During HIL simulations, a modified version of the flight software is used to use data from the nonlinear simulation in lieu of actual sensor data. Additionally, the actuator commands generated by the flight computer sent to the nonlinear simulation as inputs. The HIL simulation module is used to verify that all hardware-software compatibility issues are resolved prior to flight testing. By using linear, nonlinear, SIL, and HIL simulations, any flight software algorithm can buy its way onto a FRP aircraft for flight testing.

2.3 FRP Contribution to Research Community

The UMN FRP infrastructure is designed to be an open-source, low-cost platform for aerosystems basic research and technology demonstration at UMN and beyond. The research group at UMN promotes open-source development of both the experimental and software FRP modules. Extensive resources are made freely available at the group website `http://www.uav.aem.umn.edu`, including:

- Airframe specifications and parts lists
- Subversion repository of *Freyja* software and simulation package
- Thorough documentation of the hardware and software
- Archived flight data
- Media
- Peer-reviewed publications

This collection of resources can serve as a benchmark experimental platform for other researchers to design and validate their own flight software algorithms and compare performance with UMN designs [17].

Chapter 3

Failure Modes and Effects Analysis

This chapter provides a failure modes and effects analysis (FMEA) for the UMN *Ibis* UAV introduced in Chapter 2. A FMEA can be used to obtain an understanding of potential operational liabilities and can be an important tool for risk mitigation and fault tolerant design. An overview of the FMEA methodology is provided, and a thorough analysis for the *Ibis* follows. This scope of this chapter is mostly limited to failure modes affecting the UAV control surfaces and actuators, as those failures and effects are of primary importance to the control engineer. Further details regarding the FMEA for all *Ibis* subsystems can be found in Appendix A.

3.1 FMEA Overview

There are two broad classifications of system reliability analysis tools in use today: inductive (bottom-up) and deductive (top-down) procedures. Deductive reliability analysis requires identifying high-level system failure events and determining all lower-level events which could directly cause such a failure event. This procedure is repeated, flowing causes of failure events down to the lowest-level components. A benefit of this approach is the limited focus on one or more undesirable events and their possible causes. Fault tree analysis (FTA) is a deductive technique that is prevalent in aerospace and other safety-critical industries. One reason that FTA is favored is that extending the FTA with probabilistic risk assessment is straightforward when failure event rates are well-understood. For a detailed treatment of FTA as applied to aerospace systems, see [18].

For inductive reliability analysis, the failure modes of system components are determined at the lowest level possible, and the effects of those failures upon higher-level subsystems are traced forward to higher subsystems. This repeated for different initiating causes until analysis of all predictable failure modes are considered. The end consequences can vary according to the selected initiating cause. Eventually, the initiating causes and associated failure paths leading to total system loss can be determined. An inductive analysis has the benefit of encouraging system-wide understanding of potential failure modes and effects. For systems that have not undergone detailed reliability assessment, such as the FRP UAVs, the inductive approaches are a necessary starting point.

A FMEA is a particular inductive methodology used to analyze and discover: (1) all potential failure modes of a system, (2) the effects of these failures on the system and (3) how to correct and/or mitigate the failures or effects on the system. The correction and mitigation priority is usually based on a ranking of the severity and probability of the failure [19]. The FMEA is typically limited in scope to system hardware failures and their consequences. Human factors and software failures can be excluded from this analysis, although these considerations may remain essential to ensuring total system reliability.

A FMEA is most commonly performed during the early stages of system development in order to understand as many of the potential failure modes as possible and to design strategies to mitigate those risks. Reliability concerns can often be more easily addressed during preliminary system development. For more mature systems, a FMEA may be performed as part of a continuous product improvement effort and may be useful in identifying necessary hardware upgrades or design modifications for future design variants. For safety-critical systems, with which reliability is a paramount design feature, reliability must be analyzed throughout all phases of system development.

The FMEA procedure can be summarized into several large steps. First, the functions of all system components must be documented and the various system operational environments must be understood. Based on this information, all potential failure modes should be identified for each component. Next, the effects of each failure mode must be considered, with special attention paid to any common-mode system failures. Potential failures and effects are frequently assigned a qualitative severity (or criticality) ranking according to their damage potential and/or probability of occurrence. Several different severity criteria are employed according to application and industry. Finally, the information garnered throughout the FMEA process can be used to revise the system design to better mitigate the identified risks. A flowchart of this procedure is shown in Figure 3.1.

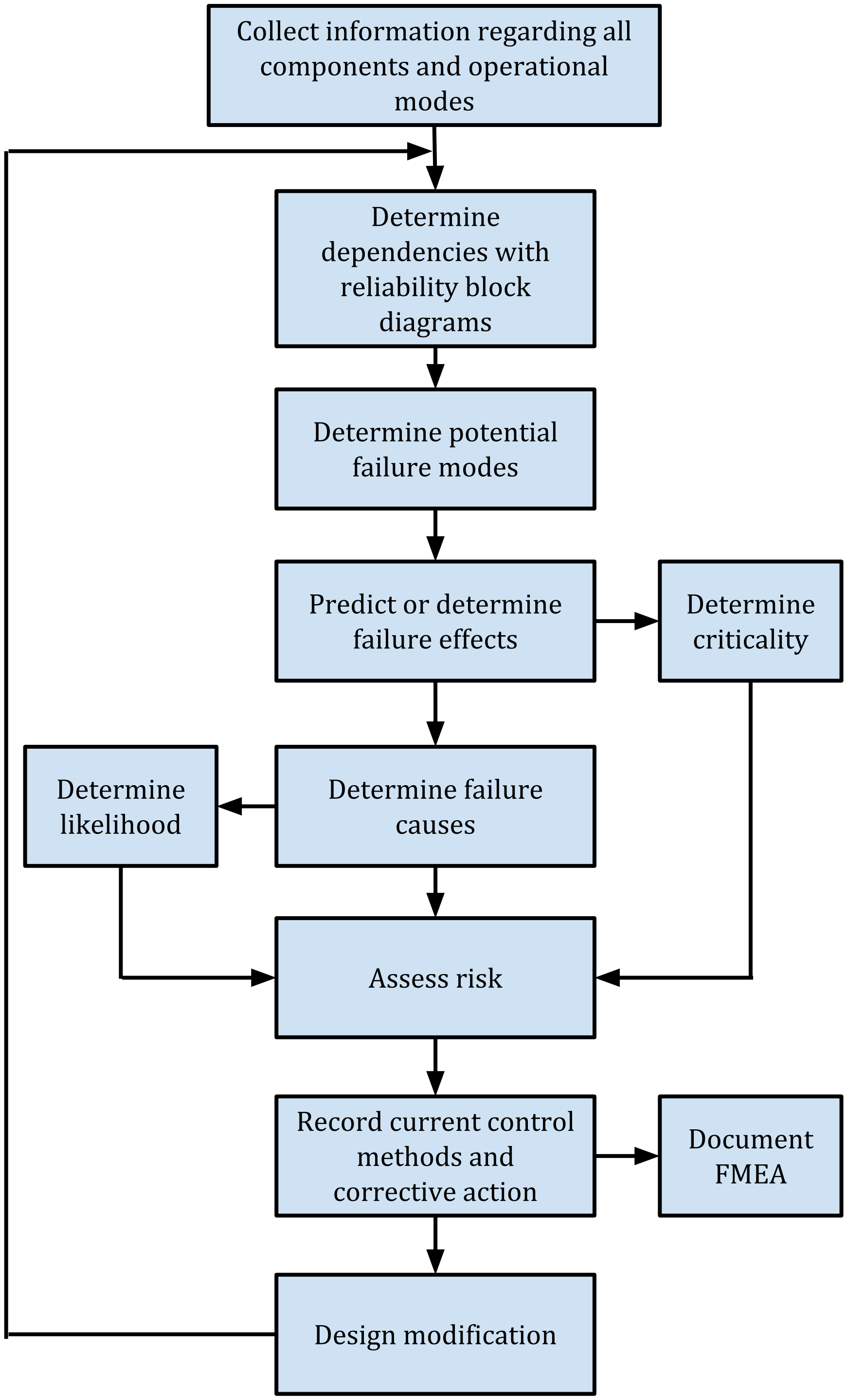


Figure 3.1: Flowchart of typical FMEA procedures

Figure 3.2: *Ibis* UAV in flight

3.2 *Ibis* Hardware Elements

This section sets up the FMEA by detailing the hardware onboard the *Ibis* UAV, organized by broad subsystem. Reliability block diagrams are presented to clarify hardware dependencies.

3.2.1 Airframe Subsystem

The aircraft has a conventional fixed-wing configuration with a single fuselage, a symmetric airfoil wing, and a vertical tail with horizontal stabilizers (Fig. 3.2). The landing gear wheels are mounted upon a dual-strut fixed external undercarriage. The wheels are of a dense foam material and hence do not utilize inflatable tires.

3.2.2 Powerplant Subsystem

The *Ibis* uses an electric powerplant system which comprises all the components necessary to produce thrust (Fig. 3.3). Two Turnigy 5000 mAh 2S 20C lithium polymer batteries provide the current for the propulsion system. Those batteries are connected in series to a Castle Creations ICE2 HV80 electronic speed controller. The speed controller receives commands via a AcroName Robotics Rx multiplexer (RxMux) which allows control of a set of servos or motors from two different signal sources. The aircraft may be piloted via R/C or with an onboard autopilot, and the RxMux enables this pilot mode switching. The speed controller regulates the voltage transmitted to the electric motor, a brushless 1900 W Actro 40-4 model. The motor shaft connects to the propeller hub and a spinner with dual Graupner 14 x 9.5 folding propeller blades.

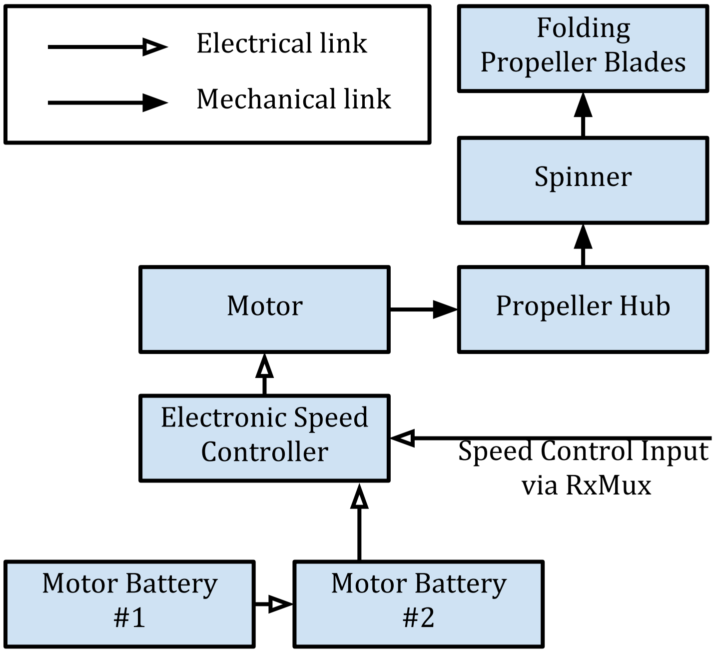


Figure 3.3: Reliability block diagram for powerplant components.

3.2.3 Effectors Subsystem

The control surface configuration on *Ibis* consists of an elevator, rudder, left/right ailerons, and left/right flaps. Each of the six control surfaces uses a single Hitec HS-5625MG digital electric servo actuator. The servos are powered by a 1350 mAh 3-cell lithium polymer battery with a Castle Creations Phoenix battery eliminator circuit (BEC). Another Castle Creations ICE2 HV80 electronic speed controller interfaces with the RxMux to relay pulse width modulation (PWM) position commands to the servos. The effectors subsystem is shown in Fig. 3.4.

3.2.4 Avionics and Sensors Subsystem

The *Ibis* uses several components in its avionics suite for guidance, navigation, control, data acquisition, and telemetry. The onboard flight computer is a Phytec MPC5200B microcontroller with a daughterboard. The computer is mounted to a interface board designed by UMN which is responsible for the power and communication interface to other avionics components . The flight computer sends telemetry data to a Free Wave MM2-T 900 MHz Modem which then broadcasts the data to the ground station. A Spektrum AR7010 radio receiver takes manual pilot commands from the ground and they are relayed to the RxMux and an Arduino Pro Mini 328 PWM reader. The read PWM data is then transmitted to the flight computer. As previously

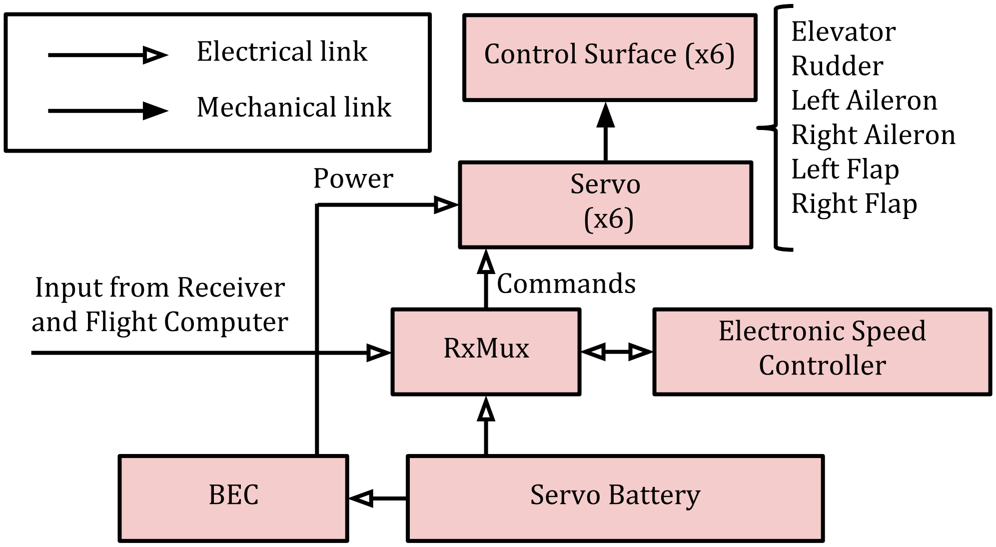


Figure 3.4: Reliability block diagram for effectors components.

described, receiver commands transmitted to the RxMux are relayed to the servos and motor speed controller during manually-piloted flight.

The sensor suite carried by *Ibis* is detailed in Section 2.1.3. At the time of this analysis, several of the sensor measurements are used merely for data acquisition and have no role in the guidance, navigation, or control of the aircraft. The critical sensors which are most important to the FMEA are the IMU, GPS, air data probe, and the pressure transducers. The IMU, GPS, pressure transducers, rotary potentiometers, motor speed sensor, camera, and voltage regulator each require power and use the servo battery. All sensor measurements are transmitted to the flight computer for data acquisition and, depending on the measurement, vehicle control. Note that planned future research will involve different sensors such as the camera (for visual-based guidance, navigation, control). The avionics and sensors components are shown in Fig. 3.5.

3.3 Failures Modes and Effects Analysis of *Ibis* UAV

Using the high level model of *Ibis* described in Sec. 3.2, the FMEA of the vehicle is performed at the component level. The function of each component is identified. A failure mode is defined as the manner or mechanism in which a component, subsystem, or system could potentially fail to meet or deliver the intended function. Table 3.1 summarizes the subsystems and components considered in the analysis.

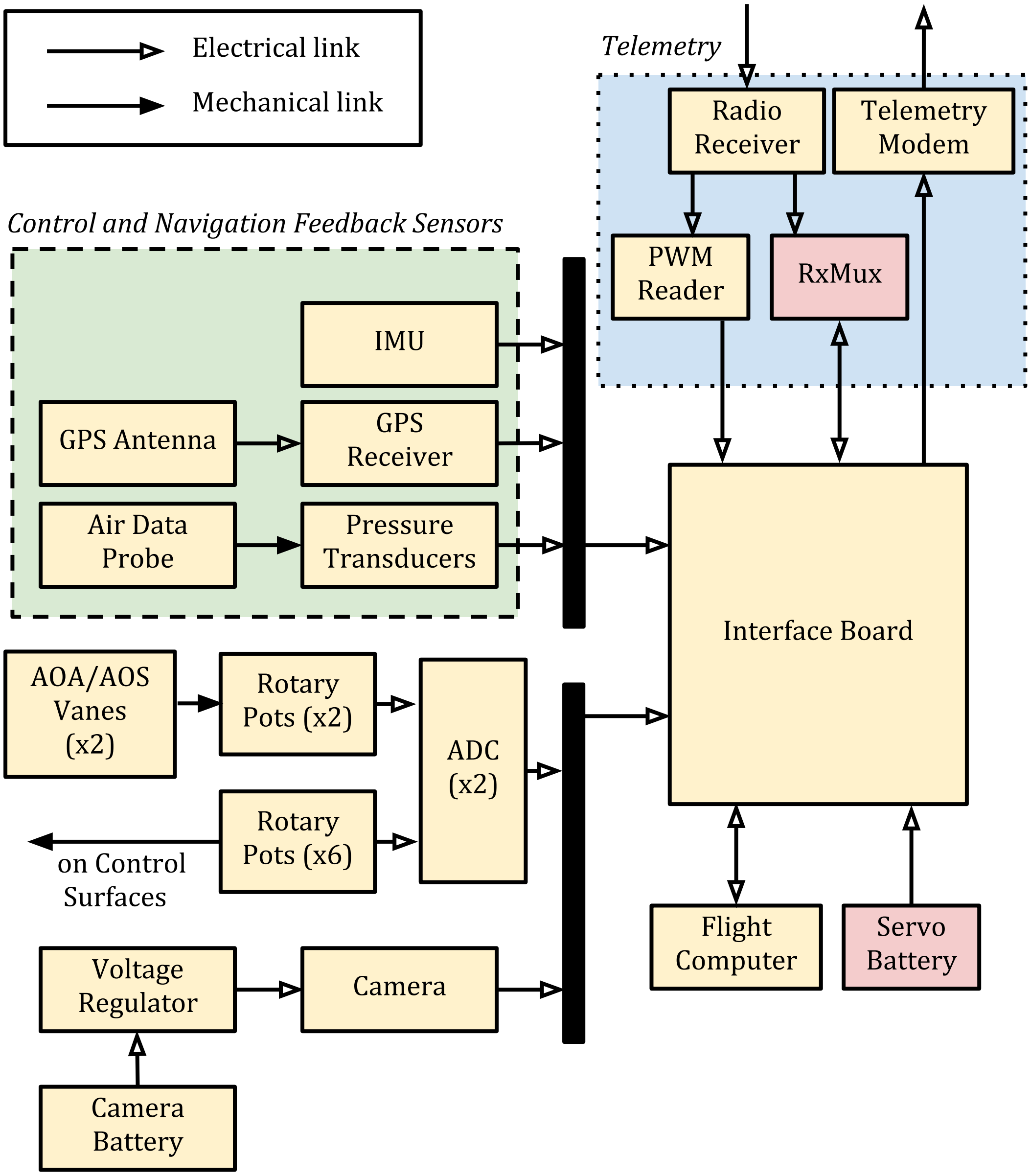


Figure 3.5: Reliability block diagram for avionics and sensors components.

Table 3.1: *Ibis* UAV Hardware Subsystems and Components

Subsystem	**ID Number**	**Component**
A. Airframe	A.1	Fuselage
	A.2	Wing
	A.3	Landing Gear
B. Powerplant	B.1	Motor Battery (x2)
	B.2	Electronic Speed Controller
	B.3	Motor
	B.4	Propeller Hub
	B.5	Spinner
	B.6	Folding Propeller Blades
C. Effectors	C.1	Servo Battery
	C.2	RxMux
	C.3	Electronic Speed Controller
	C.4	BEC
	C.5	Servo (x6)
	C.6	Control Surfaces (x6)
D. Avionics and Sensors	D.1	Flight Computer
	D.2	Interface Board
	D.3	Modem
	D.4	Radio Receiver
	D.5	PWM Reader
	D.6	IMU
	D.7	GPS
	D.8	Air Data Probe
	D.9	Pressure Transducer (x??)
	D.10	AOA/AOS Vanes (x2)
	D.11	Rotary Potentionmeters (x8)
	D.12	ADC (x2)
	D.13	Camera Battery
	D.14	Camera Voltage Regulator
	D.15	Camera

3.3.1 Evaluation Metrics

Each identified failure mode is qualitatively classified according to its *likelihood* of occurrence, *criticality* of failure effects, and overall *risk* to the system based on the previous two factors. The failure likelihood is evaluated on a simple three-point scale of low, medium, and high. For one-off research and development platforms such as *Ibis*, the likelihood of individual failure modes may not be known with confidence. Hence, a coarse rating scale is appropriate to help represent likelihoods of failure.

Table 3.2: Failure Likelihood Categories

Rating	Description
High	Failure likely to repeatably occur
Medium	Failure likely to occur occasionally
Low	Unlikely that failure would occur

The effects of each failure mode are assigned to a criticality category. As this is an aviation platform, failure effects can range from solely affecting a component, to the entire vehicle system, to other systems, people, or property. The failure criticality is chosen for each failure mode based upon the worst-case consequence of the failure to develop a conservative understanding of the set of potential failures and effects. The criticality categories taken from the NASA standards for flight platform FMEA [20] and are defined as follows:

Category 1 *Catastrophic* - Failure modes that could result in damage to the vehicle, property damage, serious injury, or loss of life.

Category 1R *Catastrophic* - Failure modes of identical or equivalent redundant hardware items that, if all failed, could result in Category 1 effects

Category 2 *Critical* - Failure modes that could result in loss of one or more mission objectives

Category 2R *Critical* - Failure modes of identical or equivalent redundant hardware items that could result in Category 2 effects if all failed.

Category 3 *Significant* - Failure modes that could cause degradation to mission objectives

Category 4 *Minor* - Failure modes that could result in insignificant or no loss to mission objectives

Unlike with the likelihood metric, the worse-case effects of a given fault mode should be relatively well-understood by the engineer. Thus, fault criticality can be rated on a more discriminating scale than likelihood.

The likelihood of a given failure and the associated effects of that failure's occurrence can be combined to provide some qualitative estimate of the risk associated with the failure mode. Broadly, critical fault modes that are likely to occur pose the utmost risk to the system and must be top priorities when enhancing system reliability. Rare and minor faults, however, pose little risk and hence, little may be gained by targeting reliability improvement efforts at mitigating these failure conditions. Because risk, like failure likelihood, is an imprecise measure, a three-point high, medium, and low rating scale is sufficient to identify system weaknesses and prioritize improvements.

Table 3.3: FMEA Risk Matrix

	Criticality					
Likelihood	4	3	2R	2	1R	1
High	M	M	H	H	H	H
Medium	L	M	M	M	H	H
Low	L	L	M	M	M	M

3.3.2 Airframe

Potential fault modes affecting the airframe are major structural failures that would lead to a significant vehicle damage and loss of mission, if not outright loss of vehicle. Little can be done to mitigate the occurrence of these failures, and only regular maintenance and inspection can prevent them from occurring. The failure modes are included in the FMEA for completeness. Table A.1 contains a full FMEA summary for the airframe subsystem.

3.3.3 Powerplant

Failures affecting the powerplant subsystem can have severe consequences for the aircraft. Each powerplant component has failure modes that could lead to a total loss of thrust, which could result in loss of mission, loss of control, and/or loss of the vehicle itself. The electronic components pose a higher risk, in general, than the mechanical components, as they are more prone to sudden failures that are harder to predict. Table A.1 contains a full FMEA summary for the powerplant subsystem.

3.3.4 Avionics and Sensors

The avionics and sensors system contains several components that are essential for safe flight, especially the flight computer (required for autopilot modes of operation) and R/C components (required for manual piloting). Failures affecting these components are most critical, but other components that provide important GNC measurements (i.e. air data probes, pressure transducers, GPS) must also remain functioning properly in most cases. Even if the vehicle does not lose control due to a particular failure, important mission data may be lost or mission objectives may become impossible. Table A.1 contains a full FMEA summary for the avionics and sensors subsystem.

3.3.5 Effectors

The subsystem of the greatest interest to the control engineer is the effectors subsystem. The effectors are the means by which the vehicle is controlled as desired, and effector failures will directly lead to unintended forces and moments acting upon the vehicle. Moreover, such faults are of interest in the aerospace fault-tolerant control research community. Detecting and assessing UAV tolerance to such servo-actuator faults is the focus of the latter part of this dissertation. A brief description of major servo actuator fault modes follows.

Bias

Here, the control surface has a constant (or slowly changing) difference between the commanded and actual deflection. This can arise due to poor rigging, slippage of servo actuator gears, or bent control linkages. This is a common fault mode for small UAVs with exposed servos and difficult handling and operating conditions. A controller can often correct for the effects of a bias and ensure continued operation, although this depends on the bias severity.

Oscillatory or Increased Deadband

The servo actuator remains operative, yet its dynamics are changed and control effectiveness is reduced. These failure modes may have either software or hardware causes. A well-designed control system may be robust to these disturbances, but the effects could lead to a loss of mission (LOM), loss of control (LOC), and loss of vehicle (LOV).

Floating Surface

The surface freely floats about its hinge due to aerodynamic moments and may cause uncommanded, time-varying disturbances to the system. All control authority in the channel is lost.

Stuck-at and Hardover

The control surface is locked at a single position. This can be caused by a broken control linkage, broken servo gears, or an unbalanced control surface. A hardover is the most extreme of all the failures because the surface is fixed at one of its extreme positions. Hence, it imposes the maximum uncommanded control forces and moments on the system and can induce dangerous uncommanded maneuvers.

Table A.1 contains a full FMEA summary for the effectors subsystem.

Chapter 4

Redesigning *Ibis* for Improved Reliability

The reliability assessment for *Ibis* revealed several opportunities to make design changes to the airframe to improve system reliability. This chapter describes those changes to the airframe and the corresponding changes to the flight research platform simulation.

4.1 *Baldr*: Reliability-focused *Ibis* Variant

A UMN senior design team studied the overall reliability of the *Ibis* aircraft and recommended several changes to improve system reliability. The changes to *Ibis* are primarily designed to increase the physical (hardware) redundancy of the aircraft for safety-critical components. Upgrades to other components were also suggested. With these ideas in mind, the design team sought to build a reliability-focused variant of the *Ibis* UAV from an original, unaltered *Ultra Stick 120* airframe. The changes described throughout Section 4.1 were the work of the senior design team, and full details can be found in their report [21]. The UAV resulting from these modifications is named *Baldr*, and it is referred to as such herein.

4.1.1 Increased Physical Redundancy

Based on a FMEA and fault tree analysis, critical components were identified as candidates for increased redundancy in the *Baldr* build. The selections were made according to the vehicle's sensitivity to failures of those components and the ability to bolster the redundancy in a cost-effective manner. The elevator surface, the rudder surface, and the avionics battery were the components chosen for modification.

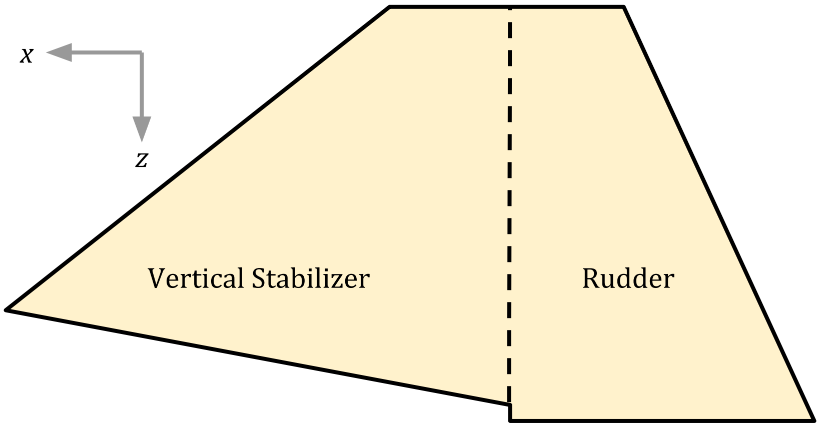


Figure 4.1: Rudder and vertical stabilizer on *Ultra Stick 120* (not to scale). The directions of the body-fixed axes are indicated.

Split Elevator Design

In a conventional configuration of the *Ultra Stick 120*, such as *Ibis*, the elevator is a single control surface with a single actuator. Because the vehicle was determined to be sensitive to elevator impediments or failures, the decision was made to split the elevator into two surfaces (left and right) and provide each surface an independent servo-actuator. The surface was split symmetrically with respect to the aircraft body x-axis (i.e., lengthwise). The split elevator should leave some degree of functional pitch control authority in the event that a single elevator servo loses effectiveness or fails entirely.

Split Rudder Design

Like the elevator, the rudder on *Ibis* is a single surface with a single servo-actuator. The rudder, however, is not symmetrical relative to the lateral-directional body axes as shown in Figure 4.1. Thus, deciding where to split the rudder is not trivial. The goal is to split the rudder into two surfaces – top and bottom rudder – such that each surface can be actuated independently and produce an equivalent side force and yawing moment for a given deflection. To do this, the combined vertical stabilizer and rudder surface shown in Figure 4.1 must be modeled as a two dimensional wing (i.e., the stabilizer) with a flap (i.e., the rudder). An arbitrary candidate cut line is drawn horizontally, and the mean aerodynamic chord (MAC) is computed for the resultant rudder partitions (Figure 4.2). The MAC of each section has a vertical stabilizer component and a rudder (flapped) component. The lift coefficient per unit span can be expressed as a function of the rudder deflection and the ratio of flapped chord to total chord length. This

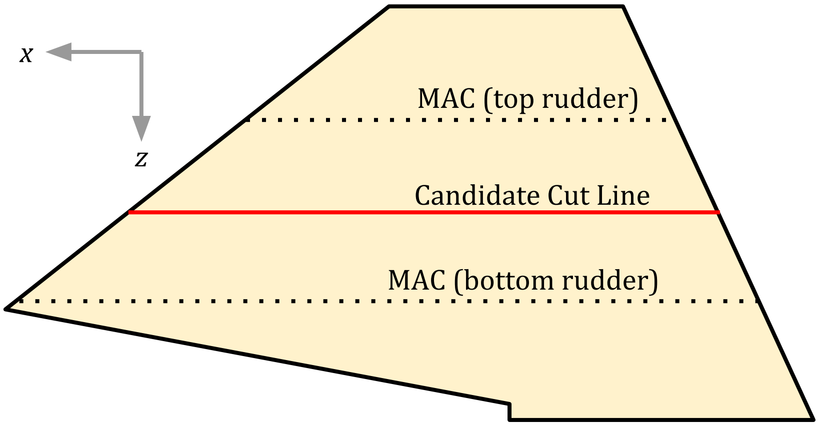


Figure 4.2: Rudder splitting with a candidate cut line and computed mean aerodynamic chords (MAC) for the resultant partitions.

can be used to compute the total lift force on each partition as a function of rudder deflection. The ideal horizontal cut location was determined to be at $z = -9.78\,\text{cm}$ from the lower aft corner of the rudder. At this location, the top and bottom rudders produce equal side force and yaw moment for equal deflection. Refer to [21] for the full details of this calculation. The rudder surface was cut in this position during the construction of *Baldr*.

Redundant Avionics Battery and Upgraded Receiver

A disruption to the power supply of the vehicle R/C components (e.g., receiver, servos, failsafe switch) could prove catastrophic. As a result, the addition of a second avionics battery was recommended to provide redundancy. Adding a second battery, however, would have required the design of a battery isolation circuit. A more advanced receiver, the Spektrum AR12120, was determined to have built-in battery isolation and had four satellite receivers (*Ibis* had 1). Three of the satellite receivers were to be powered via the main power hub, but the fourth receiver was redundant. Thus, upgrading the receiver provided satellite receiver redundancy and solved the problem of battery isolation.

A complication of the upgrade was that powering the AR12120 and the daughterboard with only two batteries was infeasible. The proposed solution was to use three batteries. Two batteries would power the receiver, which in turn would power the servos and failsafe switch. The third battery would power the daughterboard and other avionics components. The result is a complete decoupling of the power supply to the flight computer and avionics from that of the critical R/C components. To ensure that the voltages to the receiver were properly regulated, two Castle Creations battery eliminator circuits were included between the two receiver batteries and the receiver.

4.1.2 Other Modifications due to Splitting Surfaces

Splitting the elevator and rudder introduced two new surfaces, which required two additional servos and surface deflection potentiometers (CTS 250 series 100 kΩ). The potentiometer signals require conversion, and the *Ibis* carries three Semtech SX8724C analog-to-digital converters (ADC). Each ADC supports three channels, but on *Ibis*, only two channels per ADC were used for convenience reasons. The addition of the two additional potentiometers required the addition of a fourth ADC on *Baldr*.

4.2 *Baldr* Simulation

The hardware modifications and upgrades selected for *Baldr* needed to be reflected in the corresponding FRP simulation. The most significant changes to the model required were accurately accounting for the split control surface aerodynamic effects. This section describes the original aerodynamic model for *FASER* and *Ibis* and develops an extended model of split control surface aerodynamic effects for the reliability variant *Baldr*.

Fixed-wing aircraft conventionally use coupled left and right horizontal control surfaces. This coupling can be either mechanical or analytical. As previously described, *Ibis* has a mechanically coupled elevator. On many other aircraft, however, the elevator typically comprises a left and right surface and uses direct analytical coupling. Hence, the flight computer commands each surface to deflect equivalently, but the surfaces are independently actuated. Flaps exhibit similar analytical coupling. Ailerons tend to be inversely coupled to affect the rolling motion of the vehicle.

The design changes described in Section 4.1 left the elevator and rudder mechanically decoupled on *Baldr*, resulting in eight independently actuated control surfaces. Understanding the aerodynamic effects for each of these individual surfaces is useful for the development of control reconfiguration or reallocation algorithms.

4.2.1 Original *Ibis* Aerodynamic Model

The *Ibis* aerodynamic model was developed using *FASER* aircraft wind tunnel data collected at NASA Langley Research Center [22, 16, 23]. The model uses lookup tables that describe the non-dimensional aerodynamic force and moment coefficients for the vehicle throughout a large flight envelope. The model considers three main categories of aerodynamic effects:

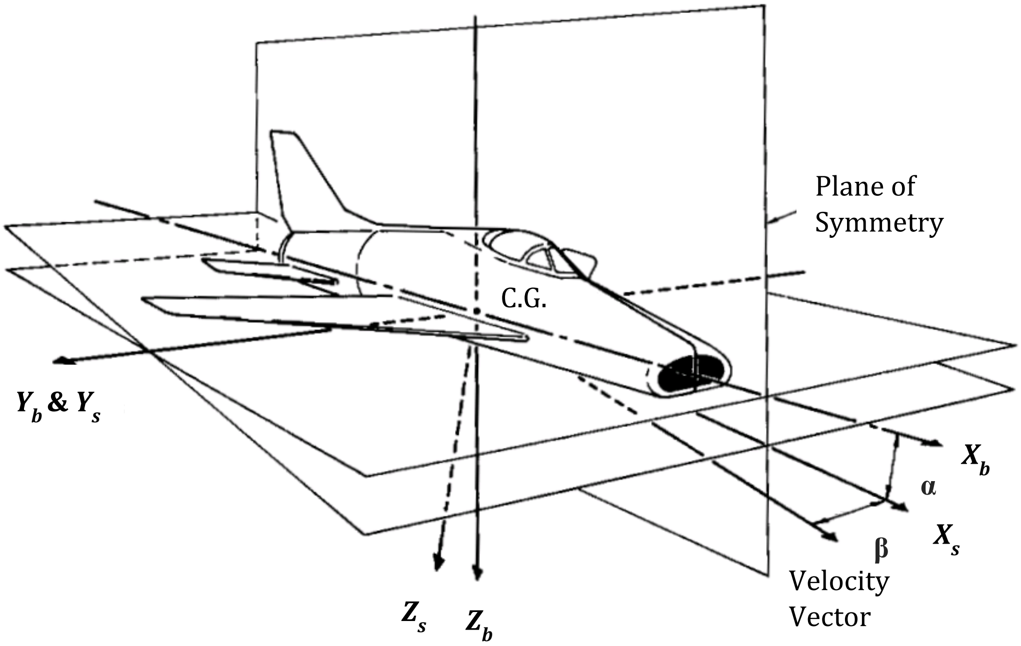


Figure 4.3: Body and stability frame definitions for example aircraft.

(i) the basic airframe effects; (ii) the control surface effects; and (iii) the dynamic derivative effects.

The force and moment coefficients for all aerodynamic effects are expressed in the vehicle stability frame. The stability frame is used to align the X-body-axis (X_b) with the oncoming flow direction. The stability frame is simply a rotation of the the body frame about the Y-body-axis (Y_b) by the trim angle of attack α. The body and stability frame conventions are shown in Figure 4.3 for an example aircraft. The forces of interest are the drag force D along the X_s-axis, the side force Y along the Y_s axis, and the lift force L along the Z_s-axis. The corresponding moments of interest are the rolling moment l, the pitching moment m, and the yawing moment n.

The independent variables for the *FASER* wind tunnel tests were angle of attack α, sideslip angle β, power level, elevator deflection δe, aileron deflection δa, rudder deflection δr, and flap deflection δf. The response variables were non-dimensional aerodynamic coefficients for drag, side, and lift forces (C_D, C_Y, and C_L) and rolling, pitching, and yawing moments (C_l, C_m, and C_n). Flight envelope sweeps were conducted using a one-factor-at-a-time (OFAT) approach. Each data point produced measured values for all independent and response variables.

Table 4.1: Modeled Force and Moment Coefficients for *Ibis*/*FASER* Control Effects

Surface or Rate	**Modeled Coefficients**
Elevator (δe)	C_D, C_L, C_m
Rudder (δr)	C_Y, C_n
Aileron (δa)	C_l
Flaps (δf)	C_D, C_L, C_m
Roll rate (p)	C_Y, C_l, C_n
Pitch rate (q)	C_m
Yaw rate (r)	C_Y, C_l, C_n

Coefficient values obtained from the OFAT approach were refined using modern experimental design techniques and multivariate orthogonal functions [22].

The way the basic airframe affects the overall aerodynamic force and moment coefficients is a nonlinear function of vehicle's orientation with respect to the relative wind. These baseline aerodynamic coefficients are denoted $C_{i_0}(\alpha, \beta)$ for $i = D, Y, L, l, m, n$ [24].

The net control surface effects are denoted $C_{i_\delta}(\alpha, \beta, \delta, J)$, where δ represents all control surface deflections and $i = D, Y, L, l, m, n$.

The dynamic derivative effects are denoted $C_{i_\Omega}(\alpha, \beta, \Omega)$ where Ω is the stability axis rotation rate and $i = D, Y, L, l, m, n$. The coefficients are obtained using two methods. First, DATCOM estimates of linear dynamic derivatives are used for the pitch damping coefficient (C_{m_q}). Next, the effects of roll and yaw rate on C_Y, C_l, and C_n are modeled as nonlinear functions of α and the respective nondimensional angular rate based on forced oscillation wind tunnel data [23].

The effects of control surfaces and rates are modeled for only those coefficients revealed to be significant in wind tunnel experiments. These significant relationships are listed in Table 4.1. Force and moment coefficients not listed for a given effector or angular velocity do not depend on changes to the respective parameter and are assumed to be zero.

The contributions of individual control effects are summed to yield net control effects force and moment coefficients. For example, $C_{D_\delta} = C_{D_{\delta e}} + C_{D_{\delta f}}$, the sum of the respective elevator and contributions to the net drag force coefficient. The net aerodynamic force and moment coefficients are obtained by summing the baseline, control surface, and dynamic derivative effects:

$$C_i = C_{i_0} + C_{i_\delta} + C_{i_\Omega} \tag{4.1}$$

The *Ibis* simulation uses these non-dimensional coefficients to compute the net forces and moments acting on the vehicle and numerically integrates the equations of motion.

4.2.2 Enhanced *Baldr* Aerodynamic Model

Because *Baldr* has a physically split rudder and elevator, along with analytically decoupled ailerons and flaps, the *Ibis* coefficient models shown in Table 4.1 no longer describe all relevant aerodynamic effects. Hence, extensions to the existing lookup table data needed to be incorporated into the *Baldr* model; these extensions would allow the new model to characterize the effect of any individual surface (e.g., left elevator, left aileron) on the six force and moment coefficients. A first principles geometric analysis was performed to enhance the aerodynamics lookup tables. A summary of that analysis and the validation of the enhanced models are now presented. For the *Baldr* aircraft, the individual control surfaces are denoted as follows: left elevator (eL); right elevator (eR); top rudder (rT); bottom rudder (rB); left aileron (aL); right aileron (aR); left flap (fL); right flap (fR).

Drag Force Coefficient (C_D) Modeling

The drag force control surface effects model for *Ibis* is of the form

$$C_{D_\delta} = C_{D_{\delta e}}(\alpha, \delta e, J) + C_{D_{\delta f}}(\alpha, \delta f, J) \tag{4.2}$$

for a single elevator and directly coupled flaps. Splitting and decoupling the control surfaces requires changes to the model. The lookup table values for the elevators and flaps were split equally between the left and right surfaces as shown in Equations 4.3-4.6

$$C_{D_{\delta eL}}(\alpha, \delta eL, J) = 1/2 \cdot C_{D_{\delta e}}(\alpha, \delta eL, J) \tag{4.3}$$

$$C_{D_{\delta eR}}(\alpha, \delta eR, J) = 1/2 \cdot C_{D_{\delta e}}(\alpha, \delta eR, J) \tag{4.4}$$

$$C_{D_{\delta fL}}(\alpha, \beta, \delta fL, J) = 1/2 \cdot C_{D_{\delta f}}(\alpha, \beta, \delta fL, J) \tag{4.5}$$

$$C_{D_{\delta fR}}(\alpha, \beta, \delta fR, J) = 1/2 \cdot C_{D_{\delta f}}(\alpha, \beta, \delta fR, J) \tag{4.6}$$

Figure 4.4 shows drag coefficient effect of left elevator deflection ($C_{D_{\delta eL}}$) versus α using the updated lookup tables. Wind tunnel data exists for $\delta e = \{-10, 0, 20\}$ deg; the other values are interpolated within the range of deflection. The Simulink model can linearly extrapolate for δe outside the [-10, 20] range, but it is not necessary to validate the correctness of the enhanced models. As Figure 4.4 shows, the drag coefficient correctly increases as α increases (due to the longitudinal component of the increased lift force) and as δeL becomes more positive (which

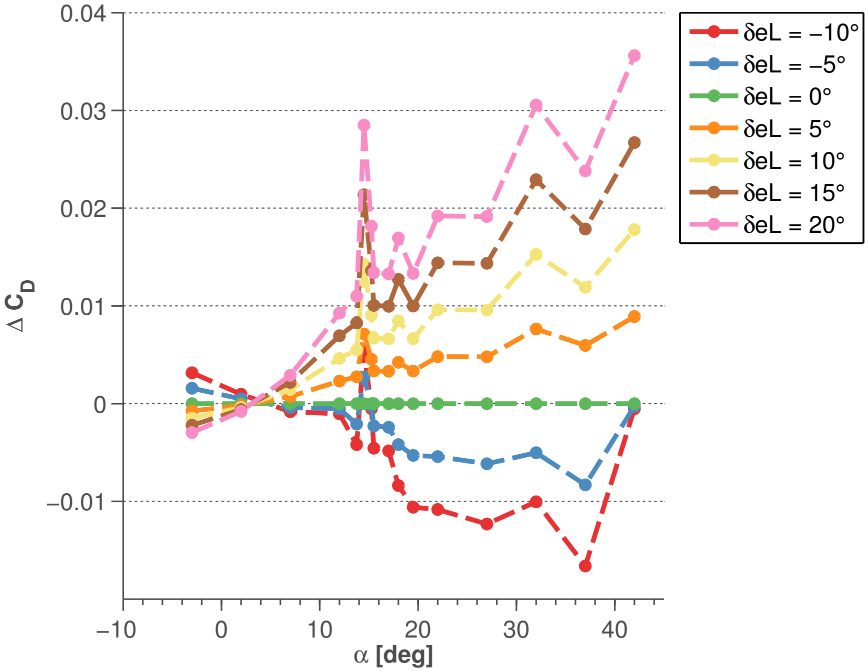


Figure 4.4: Drag force coefficient for left elevator ($C_{D_{\delta eL}}$). The coefficient for the right elevator is identical.

increases the effective α of the surface). Not pictured is the data for $C_{D_{\delta eR}}$, which matches that of Figure 4.4 as expected.

Figure 4.5 similarly shows the drag coefficient for left flap deflection, $C_{D_{\delta fL}}$. Wind tunnel data only exists for $\delta f = \{0, 30\}$ deg, so the model is interpolated such that the effects are modeled on the $\delta fL \in [-25, 25]$ deg interval. Given any deflection δfL, it is assumed that the drag coefficient is equivalent for deflection $-\delta fL$; the lookup tables for $C_{D_{\delta fL}}$ are extended accordingly. Like the elevators, the right flap drag coefficient is identical to the left.

The effects of ailerons on drag are unmodeled for *Ibis*; hence, there is no drag penalty for use of ailerons. For *Baldr*, which may use unusual control allocation to gain a reliability advantage over *Ibis*, this effect should be modeled more carefully. Because the aileron is roughly of the same dimensions as the flap, and because both surfaces are longitudinally equidistant from the nose of the vehicle, the simplifying assumption was made that:

$$C_{D_{\delta aL}} = 1/2 \cdot C_{D_{\delta f}}(\alpha, \beta, \delta aL, J) \tag{4.7}$$

Thus, the effects of the aileron deflection are determined as if it was an equivalent flap deflection. This relationship holds for the right aileron and right flap, and plots of these relationships are equivalent to the data shown in Figure 4.5. It should be noted that, although the flaps

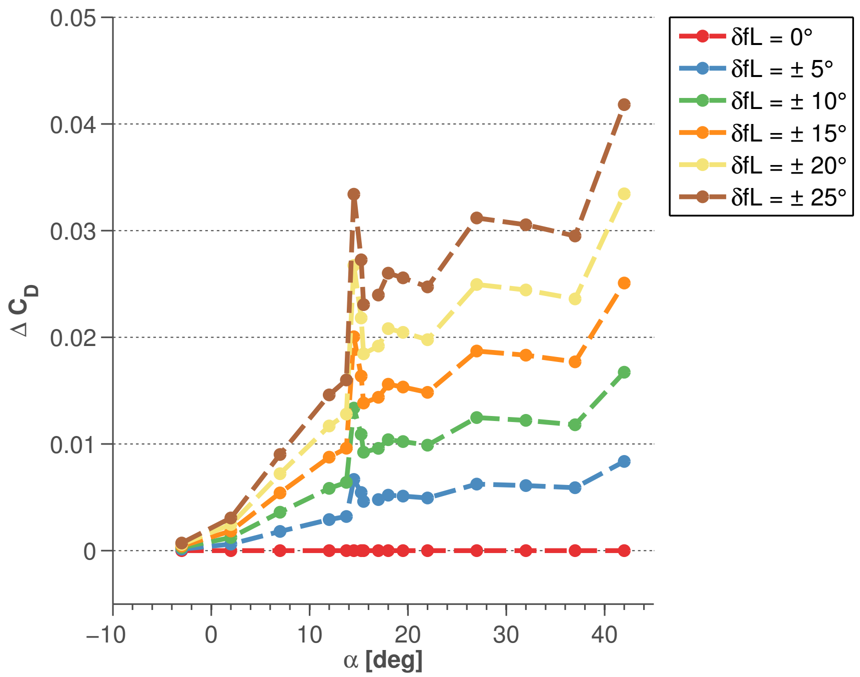


Figure 4.5: Drag force coefficient for left flap ($C_{D_{\delta fL}}$). The coefficients for the right flap, left aileron, and right aileron are identical.

lookup table takes β as an input, the flap force and moment coefficients do not depend on β. The inclusion of the β parameter is to ensure that the aileron lookup tables have consistent dimensions ($C_{l_{\delta a}}$ depends on β).

Tunnel data for rudder drag does not exist, so we assume that $C_{D_{\delta rT}} \equiv C_{D_{\delta rB}} \equiv 0$. The lookup tables in the *Baldr* simulation are populated with zeros for relationships where good data is absent. This will enable dynamic updates to the lookup tables based upon flight data collected in future system identification experiments.

Side Force Coefficient (C_Y) Modeling

All *Baldr* horizontal surfaces are assumed to contribute zero side force. Hence, modeling C_Y is restricted to the top and bottom rudders. The original lookup table data for $C_{Y_{\delta r}}$ is only defined for $\delta r \in \{-20, -5, 0\}$. The lookup tables must be expanded such that

$$\forall \delta r > 0, C_{Y_{\delta r}}(\alpha, \beta, \delta r, J) = C_{L_{\delta f}}(\alpha, -\beta, -\delta r, J) \tag{4.8}$$

With this modification, the lookup tables can be further modified to account for the split rudder.

As described in Section 4.1.1, the top and bottom rudders are sized such that they induce equal side force for equal deflection. Thus, the values in the *Ibis* lookup tables corresponding

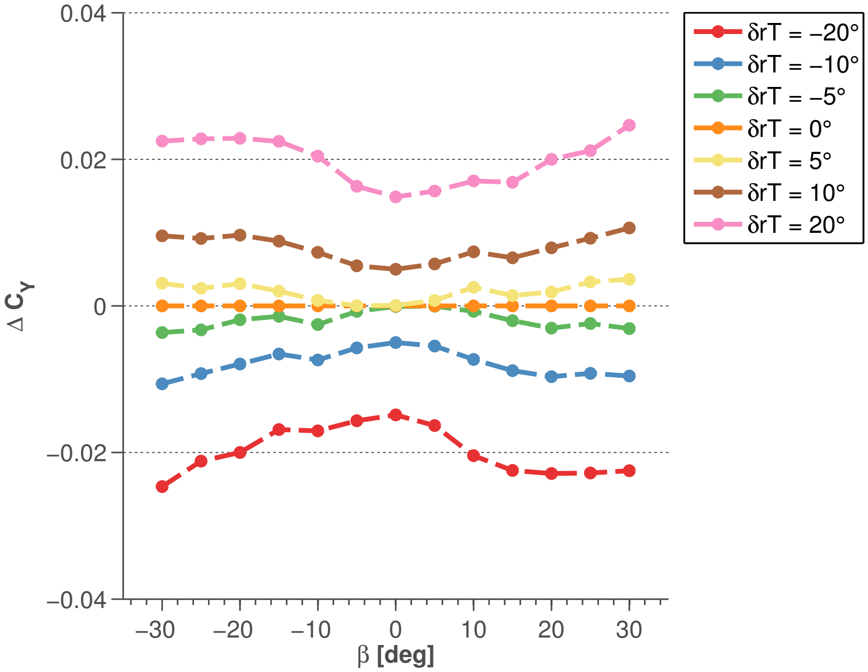


Figure 4.6: Side force coefficient for top rudder ($C_{Y_{\delta rT}}$). The coefficient for the bottom rudder is identical.

to rudder drag can be split equally between the top and bottom rudder as shown in Equations 4.9-4.10.

$$C_{Y_{\delta rT}}(\alpha, \beta, \delta rT, J) = 1/2 \cdot C_{Y_{\delta r}}(\alpha, \beta, \delta rT, J) \tag{4.9}$$

$$C_{Y_{\delta rB}}(\alpha, \beta, \delta rB, J) = 1/2 \cdot C_{Y_{\delta r}}(\alpha, \beta, \delta rB, J) \tag{4.10}$$

Figure 4.6 shows the contribution of the top rudder to the side force coefficient.

Lift Force Coefficient (C_L) Modeling

Recall that the original lookup table data for $C_{L_{\delta f}}$ is defined for $\delta f \in \{0, 30\}$. The lookup tables must be expanded such that

$$\forall \delta f < 0, C_{L_{\delta f}}(\alpha, \beta, \delta f, J) = C_{L_{\delta f}}(\alpha, \beta, -\delta f, J) \tag{4.11}$$

With this modification, the lift force coefficient modeling can be extended using the same approach used for the drag force coefficient. The C_L values from the *Ibis* lookup tables are split equally for the left and right elevator and flaps (Equations 4.12-4.15).

$$C_{L_{\delta eL}}(\alpha, \delta eL, J) = 1/2 \cdot C_{L_{\delta e}}(\alpha, \delta eL, J) \tag{4.12}$$

$$C_{L_{\delta eR}}(\alpha, \delta eR, J) = 1/2 \cdot C_{L_{\delta e}}(\alpha, \delta eR, J) \tag{4.13}$$

$$C_{L_{\delta fL}}(\alpha, \beta, \delta fL, J) = 1/2 \cdot C_{L_{\delta f}}(\alpha, \beta, \delta fL, J) \tag{4.14}$$

$$C_{L_{\delta fR}}(\alpha, \beta, \delta fR, J) = 1/2 \cdot C_{L_{\delta f}}(\alpha, \beta, \delta fR, J) \tag{4.15}$$

The C_L model for the left/right ailerons is the same as that for the left/right flaps. Note that the aileron sign convention, for the purposes of this work, departs from the usual convention. Ordinarily, a positive aileron deflection occurs with the right aileron's trailing edge down and with the left aileron's trailing edge up. Because *Baldr* uses decoupled ailerons, both ailerons are defined as having positive deflections with their trailing edges down. This simplifies the modeling of longitudinal aerodynamic effects. These changes are reflected by Equations 4.16-4.17.

$$C_{L_{\delta aL}} = 1/2 \cdot C_{L_{\delta f}}(\alpha, \beta, \delta aL, J) \tag{4.16}$$

$$C_{L_{\delta aR}} = 1/2 \cdot C_{L_{\delta f}}(\alpha, \beta, \delta aR, J) \tag{4.17}$$

Figures 4.7-4.8 show the lift force coefficient for left elevator and left flap, respectively. The model displayed Figure 4.8 is identical to that of the right flap and both ailerons.

Roll Moment Coefficient C_l

The aileron is the only *Ibis* surface for which the roll moment effects are modeled. The lookup tables are defined for $\delta a \in \{0, 5, 25\}$ deg, and they must be expanded such that

$$\forall \delta a < 0, C_{l_{\delta a}}(\alpha, \beta, \delta a) = C_{l_{\delta a}}(\alpha, -\beta, -\delta a) \tag{4.18}$$

Note that new table entries with negated β and δa breakpoints must be established. With this modification, the lookup table can be further modified to account for the decoupled ailerons on *Baldr* (Equations 4.19-4.20).

$$C_{l_{\delta aL}}(\alpha, \beta, \delta aL) = -1/2 \cdot C_{l_{\delta a}}(\alpha, \beta, \delta aL) \tag{4.19}$$

$$C_{l_{\delta aR}}(\alpha, \beta, \delta aL) = 1/2 \cdot C_{l_{\delta a}}(\alpha, \beta, \delta aR) \tag{4.20}$$

Note the sign change for the left aileron due to the different sign conventions used in the tunnel experiments and in this dissertation. Figure 4.9 shows the roll moment coefficient for the left aileron; the coefficient for the right aileron is the negative of the data shown.

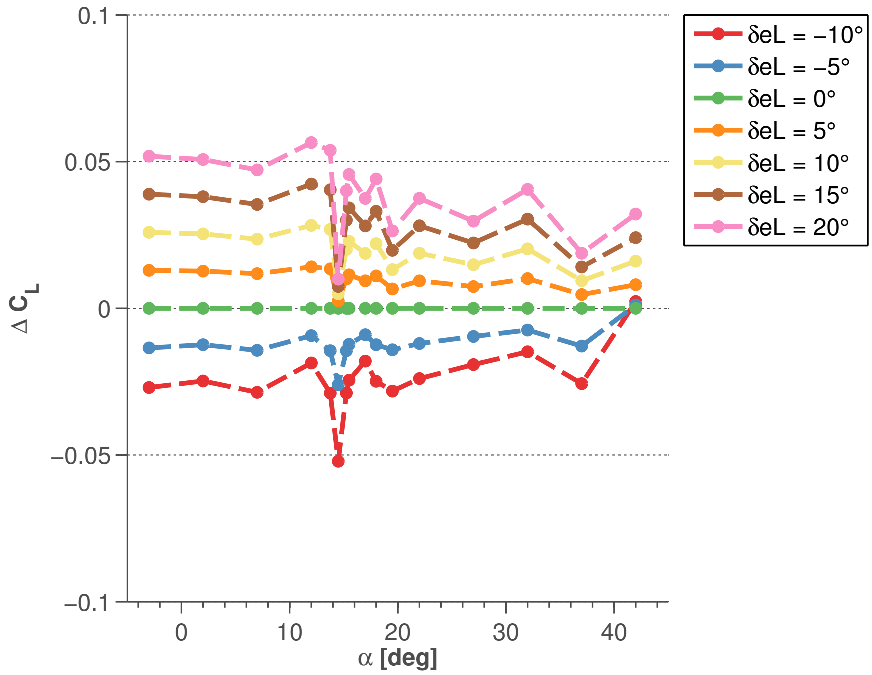


Figure 4.7: Lift force coefficient for left elevator ($C_{L_{\delta eL}}$). The coefficient for the right elevator is identical.

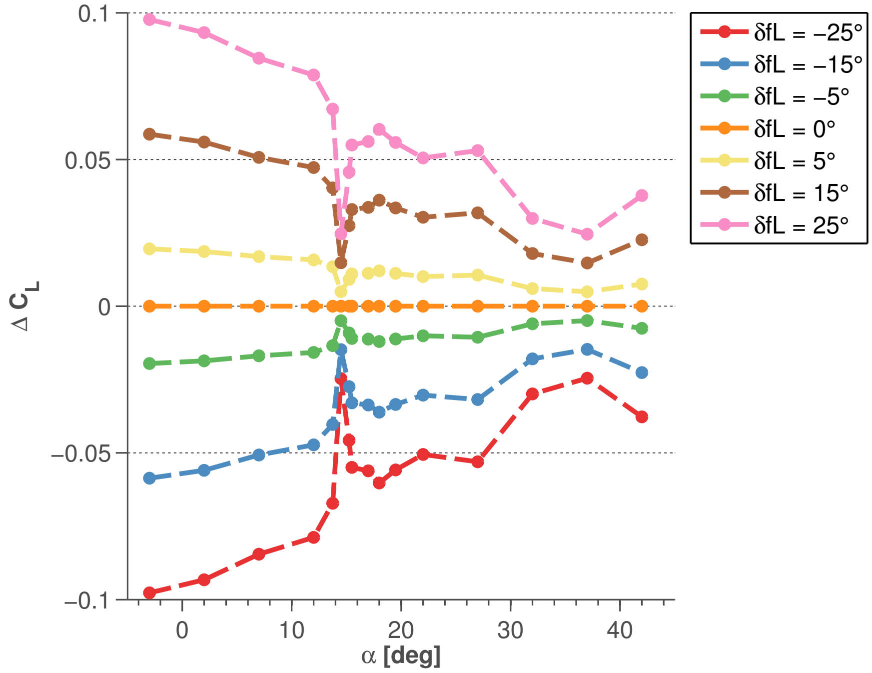


Figure 4.8: Lift force coefficient for left flap ($C_{L_{\delta fL}}$). The coefficients for the right flap, left aileron, and right ailerons are identical.

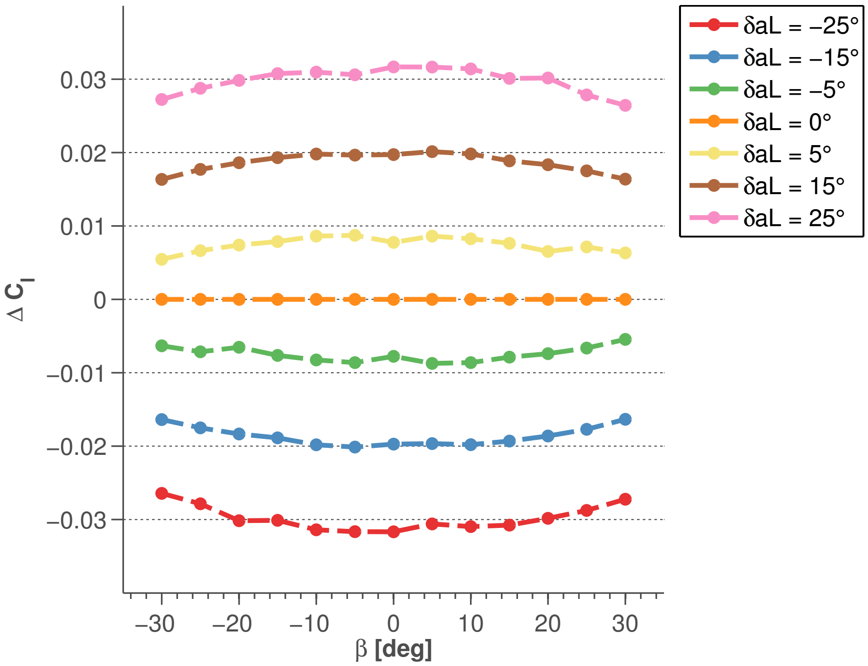


Figure 4.9: Roll moment coefficient for left aileron ($C_{l_{\delta aL}}$). The coefficient for the right aileron is the negative of the data shown here.

The roll moment coefficient for the left and right elevators is unmodeled for *Ibis*. Asymmetric deflections can induce a rolling moment, however, so these effects must be characterized. The roll moment is modeled as the lift force acting upon each elevator surface multiplied by the roll moment arm length

$$C_{l_{\delta eL}}(\alpha, \delta eL, J) = C_{L_{\delta eL}}(\alpha, \delta eL, J) \cdot \frac{\bar{y}_{hstab}}{b} \tag{4.21}$$

$$C_{l_{\delta eR}}(\alpha, \delta eR, J) = -C_{L_{\delta eR}}(\alpha, \delta eR, J) \cdot \frac{\bar{y}_{hstab}}{b} \tag{4.22}$$

where b is the aircraft wingspan and $\bar{y}_{hstab}$ is the moment arm length (i.e., the distance from the left horizontal stabilizer center of pressure to X_b). The center of pressure is assumed to be the centroid of the surface. Scaling $\bar{y}_{hstab}$ by b preserves the non-dimensionality of C_l. For $\delta eL > 0$, $C_{L_{\delta eL}} > 0$; the increased lift force on the left wing induces a positive rolling moment. For $\delta eR > 0$, however, a negative rolling moment is induced (hence, the sign difference in Equation 4.22).

A similar approach is used to model the roll coefficients for the rudders and flaps, as shown

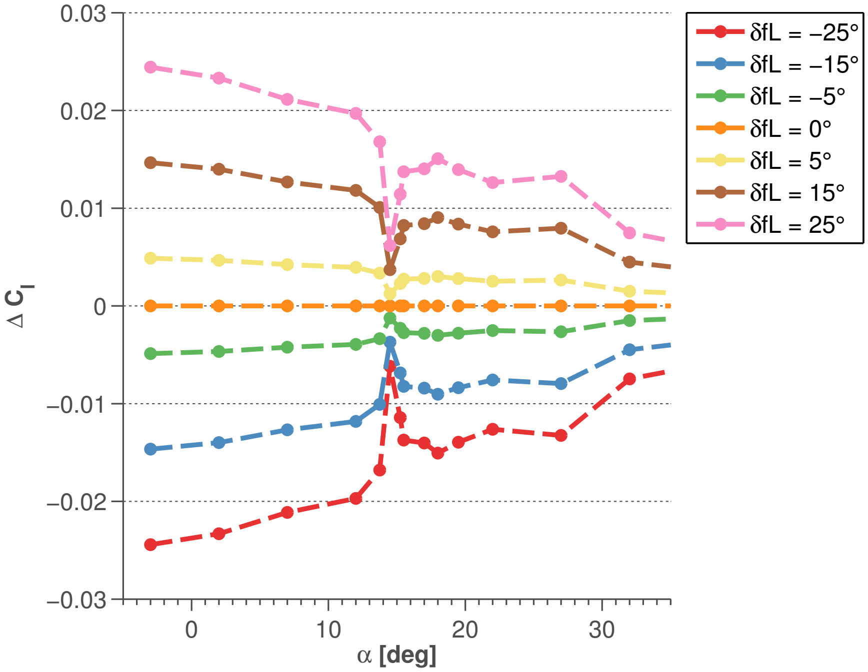


Figure 4.10: Roll moment coefficient for left flap ($C_{l_{\delta fL}}$). The coefficient for the right flap is the negative of the data shown here.

in Equations 4.23-4.26.

$$C_{l_{\delta rT}}(\alpha,\beta,\delta rT,J) = C_{Y_{\delta rT}}(\alpha,\beta,\delta rT,J)\cdot\frac{\bar{z}_{rT}}{b} \tag{4.23}$$

$$C_{l_{\delta rB}}(\alpha,\beta,\delta rB,J) = -C_{Y_{\delta rB}}(\alpha,\beta,\delta fB,J)\cdot\frac{\bar{z}_{rB}}{b} \tag{4.24}$$

$$C_{l_{\delta fL}}(\alpha,\beta,\delta fL,J) = C_{L_{\delta fL}}(\alpha,\beta,\delta fL,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.25}$$

$$C_{l_{\delta fR}}(\alpha,\beta,\delta fR,J) = -C_{L_{\delta fR}}(\alpha,\beta,\delta fR,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.26}$$

$\bar{z}_{rT}$ and $\bar{z}_{rB}$ are the distance from the rudder surface center of pressure to X_b for the top and bottom rudder, respectively. Note that Equations 4.23-4.24 use C_Y to compute the rolling moment induced by the split rudders. In Equations 4.25-4.26, $\bar{y}_{wing}$ is the distance from the wing center of pressure to X_b.

Figure 4.10 shows the roll moment coefficient for the left aileron; as indicated in Equation 4.26, the coefficient for the right aileron is the negative of the data shown. Roll moment coefficient figures for the top and bottom rudders are omitted; their roll effects are 1-2 orders of magnitude smaller, respectively, than the aileron surfaces.

Pitch Moment Coefficient C_m

The *Ibis* simulation includes C_m models for both the elevator and the flaps. The lookup table values can be split to describe the contributions of the left and right elevator and flaps as shown in Equations 4.27-4.32.

$$C_{m_{\delta eL}}(\alpha, \beta, \delta eL, J) = 1/2 \cdot C_{m_{\delta e}}(\alpha, \beta, \delta eL, J) \tag{4.27}$$

$$C_{m_{\delta eR}}(\alpha, \beta, \delta eR, J) = 1/2 \cdot C_{m_{\delta e}}(\alpha, \beta, \delta eR, J) \tag{4.28}$$

$$C_{m_{\delta fL}}(\alpha, \beta, \delta fL, J) = 1/2 \cdot C_{m_{\delta f}}(\alpha, \beta, \delta fL, J) \tag{4.29}$$

$$C_{m_{\delta fR}}(\alpha, \beta, \delta fR, J) = 1/2 \cdot C_{m_{\delta f}}(\alpha, \beta, \delta fR, J) \tag{4.30}$$

The pitching moment due to ailerons is related to the lift and drag forces induced by the surface deflection. Because the ailerons and flaps are assumed to have the same lift and drag effects, it follows that the C_m model for the decoupled flaps (Equations 4.29-4.32) also describes the C_m effects of the decoupled ailerons:

$$C_{m_{\delta aL}}(\alpha, \beta, \delta aL, J) = C_{m_{\delta fL}}(\alpha, \beta, \delta aL, J) \tag{4.31}$$

$$C_{m_{\delta aR}}(\alpha, \beta, \delta aR, J) = C_{m_{\delta fR}}(\alpha, \beta, \delta aR, J) \tag{4.32}$$

Figures 4.11-4.12 show the pitch moment coefficients for the left elevator and left flap, respectively. The data shown in Figure 4.12 is the same as for the left and right aileron pitch moment coefficients.

The pitching moment due to rudder deflection is assumed to be zero.

Yaw Moment Coefficient C_n

The *Ibis* simulation only includes C_n models for $\delta r \in \{-20, -5, 0\}$, and the lookup tables must be expanded such that

$$\forall \delta r < 0, C_{n_{\delta r}}(\alpha, \beta, \delta r, J) = C_{n_{\delta f}}(\alpha, -\beta, -\delta r, J) \tag{4.33}$$

With this modification, the lookup table can be futher modified to account for the split rudder (Equations 4.34-4.35). Because the split rudders are designed to produce equal side force for equal deflection, the yawing moment models are equivalent.

$$C_{n_{\delta rT}}(\alpha, \beta, \delta rT) = 1/2 \cdot C_{n_{\delta r}}(\alpha, \beta, \delta rT) \tag{4.34}$$

$$C_{n_{\delta rB}}(\alpha, \beta, \delta rB) = 1/2 \cdot C_{n_{\delta r}}(\alpha, \beta, \delta rB) \tag{4.35}$$

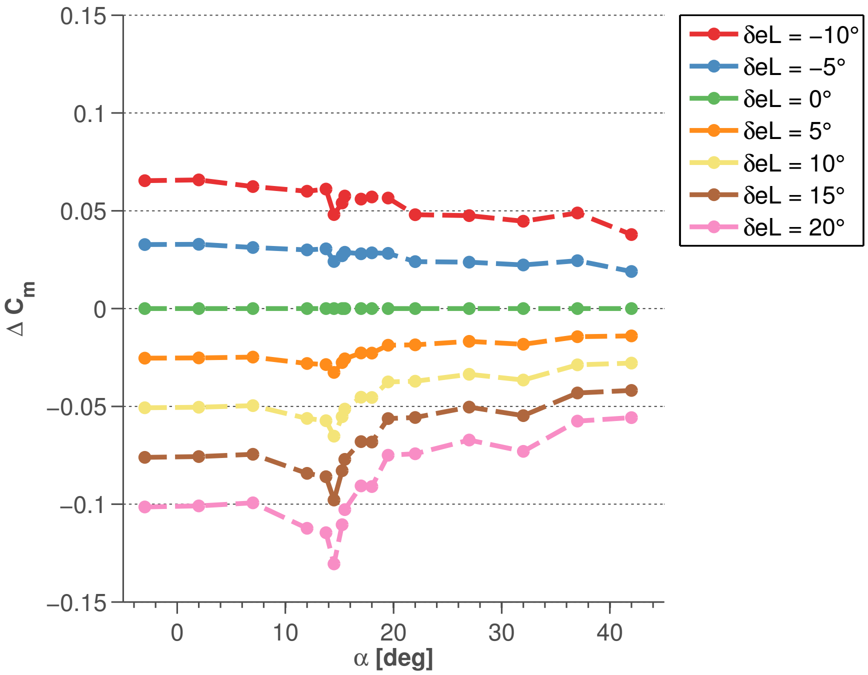


Figure 4.11: Pitch moment coefficient for left elevator ($C_{m_{\delta eL}}$). The coefficient for the right elevator is identical.

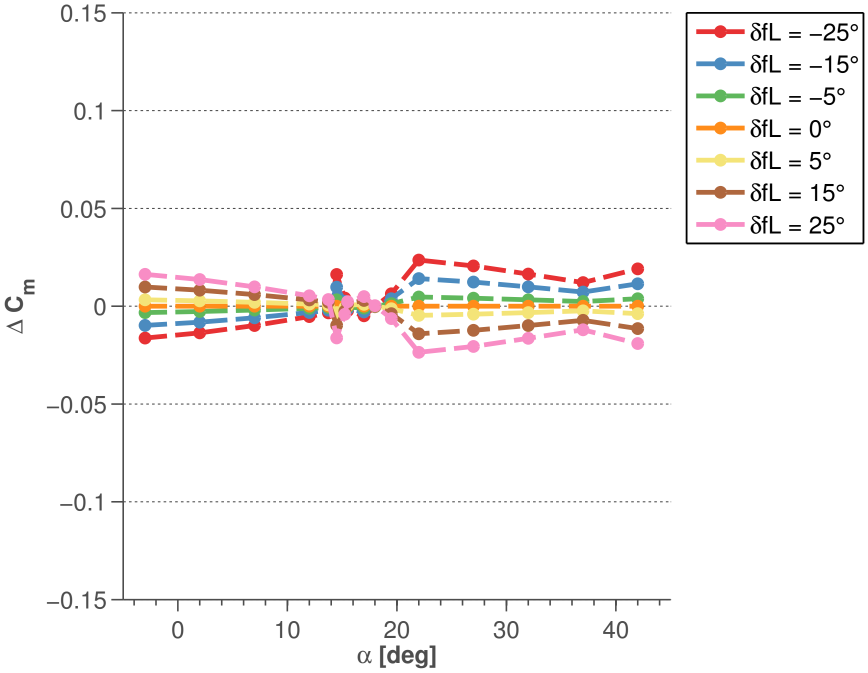


Figure 4.12: Pitch moment coefficient for the left flap ($C_{m_{\delta fL}}$). The coefficients for the right flap, left aileron, and right aileron are identical.

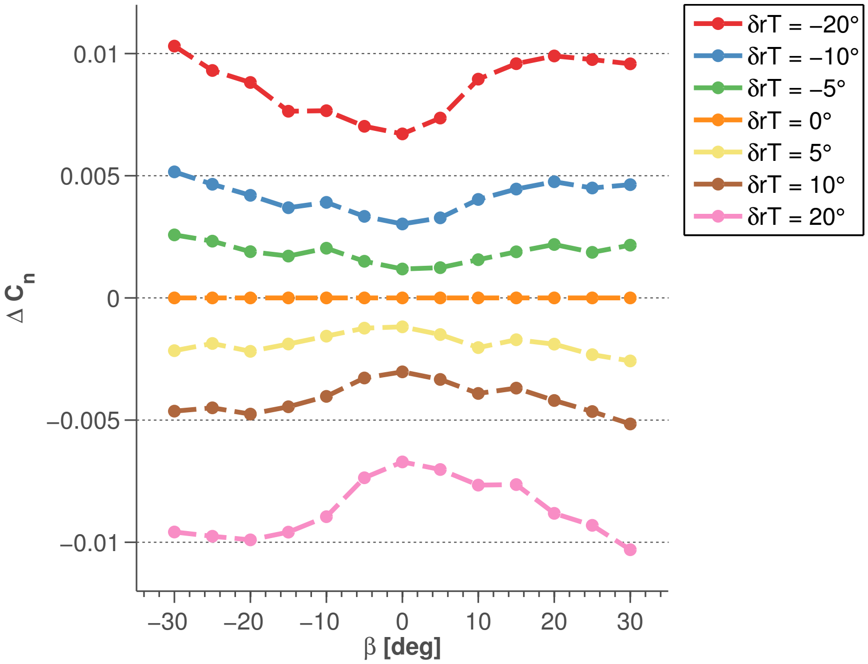


Figure 4.13: Yaw moment coefficient for top rudder ($C_{n_{\delta rT}}$). The coefficient for the bottom rudder is identical.

Figure 4.13 shows the yaw moment coefficient for the top rudder; the coefficient for the bottom rudder is identical.

Similar to the methods used for modeling C_l, C_n is computed by determining the drag force on the relevant surface and multiplying by the yaw moment arm length – the distance to Z_b. This approach is followed for each of the six horizontal surfaces according to Equations 4.36-4.41.

$$C_{n_{\delta eL}}(\alpha,\beta,\delta eL,J) = -C_{D_{\delta eL}}(\alpha,\beta,\delta eL,J)\cdot\frac{\bar{y}_{hstab}}{b} \tag{4.36}$$

$$C_{n_{\delta eR}}(\alpha,\beta,\delta eR,J) = C_{D_{\delta eR}}(\alpha,\beta,\delta eR,J)\cdot\frac{\bar{y}_{hstab}}{b} \tag{4.37}$$

$$C_{n_{\delta fL}}(\alpha,\beta,\delta fL,J) = -C_{D_{\delta fL}}(\alpha,\beta,\delta fL,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.38}$$

$$C_{n_{\delta fR}}(\alpha,\beta,\delta fR,J) = C_{D_{\delta fR}}(\alpha,\beta,\delta fR,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.39}$$

$$C_{n_{\delta aL}}(\alpha,\beta,\delta aL,J) = -C_{D_{\delta aL}}(\alpha,\beta,\delta aL,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.40}$$

$$C_{n_{\delta aR}}(\alpha,\beta,\delta aR,J) = C_{D_{\delta aR}}(\alpha,\beta,\delta aR,J)\cdot\frac{\bar{y}_{wing}}{b} \tag{4.41}$$

From Equations 4.5-4.7, it is apparent that Equations 4.38 and 4.40 are equivalent relations;

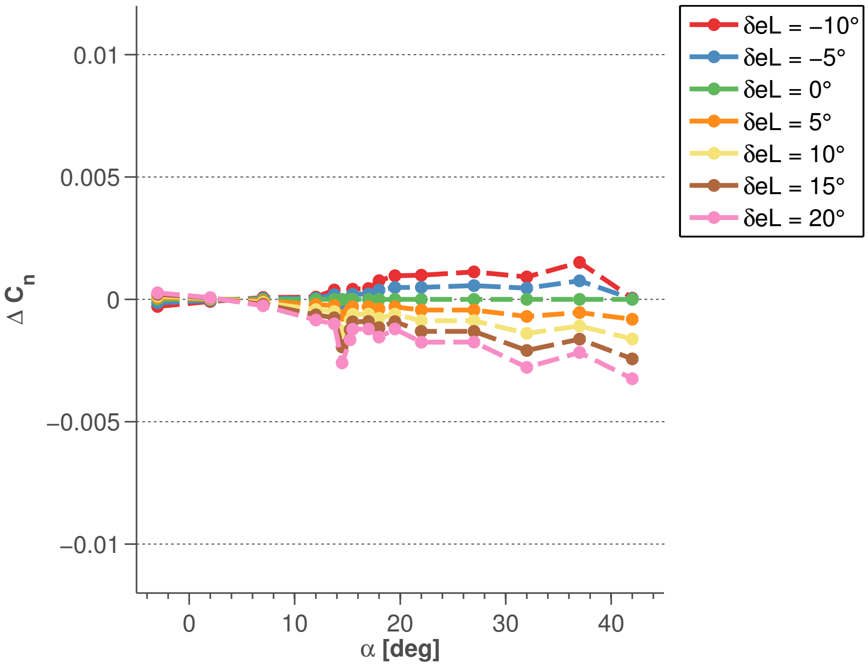


Figure 4.14: Yaw moment coefficient for left elevator ($C_{n_{\delta eL}}$). The coefficient for the right elevator is negative of the data shown.

this also holds for Equations 4.39 and 4.41.

Figures 4.14-4.15 show the yaw moment coefficient for the left elevator and left flap, respectively. α is chosen as the independent variable for these figures (as opposed to β) because these coefficients do not vary with β according to the models. This phenomenon is likely a deficiency in the model, but the β dependence can be further evaluated in future flight testing. The data shown in Figure 4.15 is identical to that for the left aileron, and the negative of the right flap and right aileron. Note that deflection of the flaps and ailerons induce significantly larger yawing moments than do the elevators (but smaller than the rudders).

Summary

This section has highlighted the deficiencies of the original *Ibis* aerodynamic model for simulations using split and/or analytically decoupled control surfaces. Based on the physical modifications to the rudder and elevator that were made on *Baldr*, an enhanced aerodynamic model was developed which characterized the most significant effects for each independent control surface. These characterizations were made using a first principles geometry-based approach, with the goal of establishing a new baseline aerodynamic model that is more representative

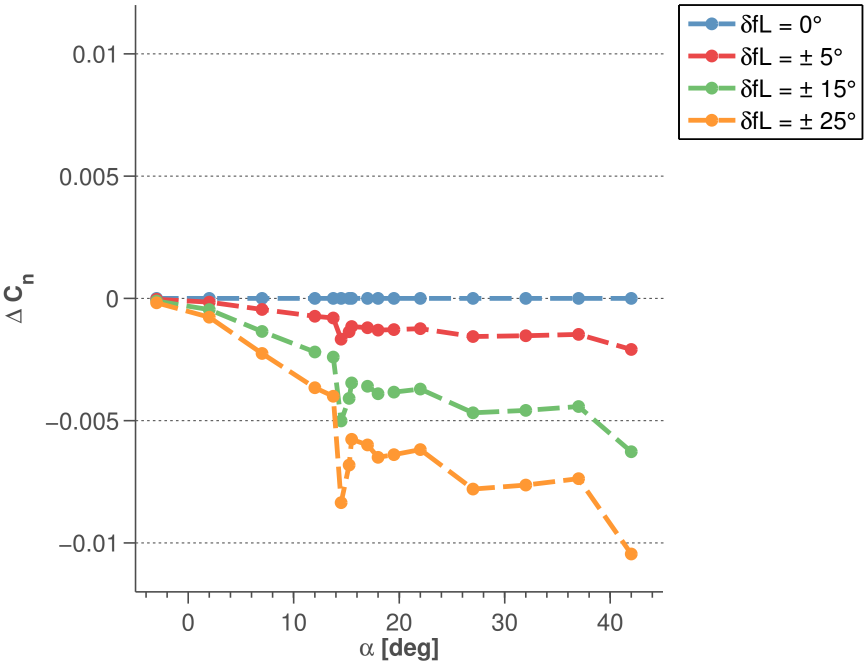


Figure 4.15: Yaw moment coefficient for left flap ($C_{n_{\delta fL}}$). The coefficient for the left aileron is identical, while the coefficients for the right flap and right aileron are the negative of the data shown.

off the real effects than the original. A visual summary of those changes is presented in Table 4.2, where maroon denotes the relationships originally modeled for *Ibis* from wind tunnel data, and gold denotes the extensions to those models for split and decoupled surfaces. The lookup tables for *Baldr* were configured such that all relationships, including the unmodeled relationships marked with gray, could be dynamically updated using flight test data from system identification experiments. Improving this aerodynamic model enables further research in the areas of fault-tolerant reconfigurable control using the UMN *Freyja* simulation and *Baldr* UAV.

Table 4.2: Coefficients in Original (Maroon) and Enhanced (Gold) Aerodynamic Models

	C_D	C_Y	C_L	C_l	C_m	C_n
δe	🟥	⬜	🟥	⬜	🟥	⬜
$\delta eL, \delta eR$	🟨	⬜	🟨	🟨	🟨	🟨
δa	⬜	⬜	⬜	🟥	⬜	⬜
$\delta aL, \delta aR$	🟨	⬜	🟨	🟨	🟨	🟨
δr	⬜	🟥	⬜	⬜	⬜	🟥
$\delta rT, \delta rB$	⬜	🟨	⬜	🟨	⬜	🟨
δf	🟥	⬜	🟥	⬜	🟥	⬜
$\delta fL, \delta fR$	🟨	⬜	🟨	🟨	🟨	🟨

Chapter 5

Reliability Assessment Using Trim State Discovery

Conventional industry reliability approaches do not scale well to small UAS. The FMEA traditionally assumes worst-case effects of failures and can be overly conservative. Typically, pervasive hardware redundancy with thorough FMEA and FTA is sufficient to achieve the high levels of reliability that have become standard in the commercial aerospace industry. Where adaptive or reconfiguration strategies are used, they are limited and can be crude. Small UAS, however, do not have the luxury of pervasive hardware redundancy. For truly reliable small UAS, analytical redundancy approaches must be developed that can quickly and accurately detect failures, isolate failures, and reconfigure the control strategy to safely accommodate failures. Extending the FMEA using detailed control systems analysis complement traditional tools by determining which flight conditions or operational modes may be achievable for a healthy and faulted system. Developing a deeper understanding of a system's physical performance limitations can provide a richer view of failure modes, their effects and criticality, design changes necessary to prevent or mitigate overall risk, and insight regarding appropriate certification procedures.

This section describes the determination of the achievable flight envelope for *Ibis* and *Baldr* under different control coupling configurations. Splitting the control surfaces, as recommmended for *Baldr* in Chapter 4 provides hardware redundancy for surface actuation failures. Further reconfiguration by decoupling the control commands to each surface pair provides further analytical redundancy. Control action can sometimes be reallocated to different effectors to compensate for the loss of control authority in a failed channel; this capability is necessary for many active adaptive control techniques.

A trim state discovery analysis is used to determine the flight envelope for a single flight condition – steady wings-level flight. After identifying the operating points within the achievable flight envelope, a linear model can be obtained for each operating point. Several familiar linear analysis tools can be applied to this linear model set, and the resulting data will reveal the extent to which the vehicle can accommodate faults such as those described in Chapter 3 and Appendix A.

5.1 Trim State Discovery Method

The vehicle model has implicit state equations in the general form $f(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}) = 0$, where f is an array of nonlinear functions, $\mathbf{x}$ is the system state vector, and $\mathbf{u}$ is the input vector. An equilibrium point of the nonlinear system is achieved when these state equations are satisfied by $\dot{\mathbf{x}} \equiv 0$, $\mathbf{u} \equiv 0$ (or constant). For an aircraft, this is said to be a *stick-fixed, steady-state* flight condition.

The Earth position equations of motion can be decoupled and excluded from consideration in the trimming procedure. A steady-state flight condition can be described using the remaining states. The flight condition of interest is steady, wings-level flight (SLF), and the following constraints must hold for SLF:

$$\dot{p}, \dot{q}, \dot{r} \text{and } \dot{u}, \dot{v}, \dot{w} \text{ (equivalently } \dot{V}, \dot{\beta}, \dot{\alpha}) \equiv 0 \tag{5.1}$$

$$\phi, \dot{\phi}, \dot{\theta}, \dot{\psi} \equiv 0 \Leftrightarrow p, q, r \equiv 0 \tag{5.2}$$

where p, q, and r are the body-frame angular velocities, u, v, and w are the body-frame velocities, and ϕ, θ, and ψ are the roll, pitch, and yaw angles respectively. SLF is chosen as the reference flight condition because it is such a common flight mode during cruise phase and for flying from waypoint to waypoint. A vehicle that is unable to maintain SLF would, in most instances, be unable to complete its mission and would be at risk for loss of control or total loss of vehicle.

A particular SLF condition is chosen by specifying the trim targets airspeed $V^* = 23\,\text{m/s}$, flight path angle $\gamma^* = 0\,\text{deg}$, sideslip $\beta^* = 0\,\text{deg}$, and altitude $h^* = 100\,\text{m}$. This SLF condition represents the nominal experimental flight test condition for most *Ibis* research. A trim solution of the form $f(\mathbf{x}^*, \mathbf{u}^*) = 0$ is computed via numerical gradient-descent optimization (`fmincon`), in which $\mathbf{x}^*$ and $\mathbf{u}^*$ are the trimmed state and control input vectors, respectively. The trim solution for this nominal, unfaulted flight condition is denoted $(\mathbf{x}_\mathbf{0}^*, \mathbf{u}_\mathbf{0}^*)$ [25].

A stuck elevator primarily affects the aircraft longitudinal dynamics, for example. The objective of the trim state discovery is to determine the set of feasible SLF trim conditions across

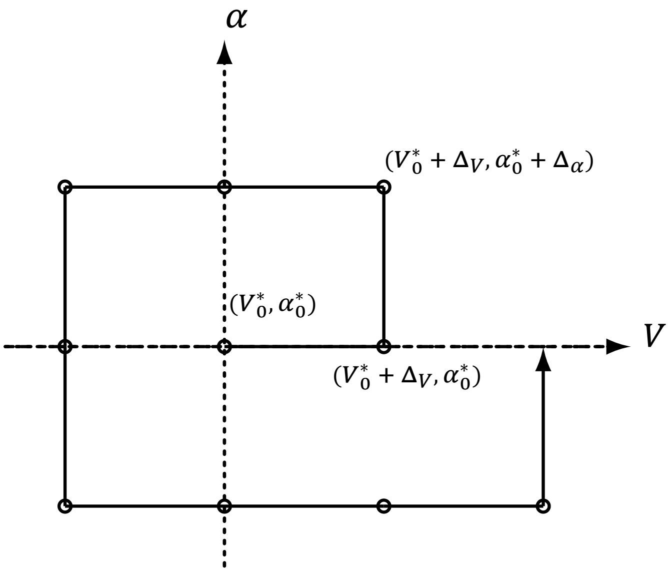


Figure 5.1: Counterclockwise spiral trim target selection over airspeed - angle of attack grid.

the V-α plane (i.e., sideslipping conditions are ignored and $\beta \equiv 0$). To achieve this, trim attempts are made at locations throughout the V-α plane that are defined by a rectangular grid centered at $(V_0^*, \alpha_0^*) = (23, 4.5)$ with grid increments $(\Delta_V, \Delta_\alpha) = (0.34, 0.25)$.

The numerical optimization algorithm used for trimming is sensitive to the initial conditions provided to the nonlinear simulation model. The algorithm performs more effectively when the initial inputs, outputs, and states are 'close' to the solution. The results from a previously determined trim condition are used as initial conditions to better condition subsequent optimizations. Three features of the trim state discovery algorithm help to condition the optimizations for the best results. First, fine grid increments are selected to minimize the changes due to nonlinear effects between grid locations. Second, the extent of the viable flight envelope is determined by successively moving outward from the nominal trim condition in a counterclockwise spiral path (Fig. 5.1). The full rectangular grid is covered by the spiral path, and trim attempts are made at each point along the path. This quasi-polar approach allows for an exploration of the trim space with a monotonically-increasing distance from the well-understood nominal trim condition. Finally, the distance in the V-α plane to all previously-determined successful trim conditions is obtained at each spiral path location. The nearest previously-determined successful trim condition results are selected to initialize the nonlinear model trim optimization. Together these safeguards ensure a well-conditioned optimization and improved quality of results. Additionally, the realistic input saturation limits are characterized in the optimization specifications; all control surfaces are restricted to deflections on the [-25, 25] deg interval, and the throttle is limited to nondimensional values on [0, 1]. These saturation limits are accurate for UMN FRP aircraft.

The trim state discovery procedure can be summarized as follows:

1. Configure nonlinear model in *Freyja* simulation package for aircraft of interest.
2. Set trim target for reference flight condition of interest
3. Trim aircraft at this target to obtain initial operating point
4. Generate spiral trim target path in V-α plane as shown in Fig 5.1
5. Attempt trim at each location in spiral path
   (a) Compute distances to all previously-determined successful operating points in V-α space and select nearest operating point
   (b) Initialize nonlinear model with values from nearest operating point
   (c) Attempt trim; if successful, save results.

5.2 Achievable Steady Wings-level Flight Envelope for *Ibis*

The trim state discovery procedure is first applied for the *Ibis* aircraft using the conventional control surface configuration described in Section 4.1. Thus, the flaps are directly coupled while the ailerons are inversely coupled. A total of 1,298 (out of 10,201) operating conditions satisfying the SLF constraints were identified over the V-α trim target grid. Fig. 5.2 shows these operating points projected onto the V-α plane. The SLF envelope is contiguous when viewed in the V-α plane. At the slowest airspeeds, the velocity contribution to lift force is reduced and higher angles of attack (with corresponding increases in lift coefficient) are required to achieve sufficient lift for SLF. As the airspeed increases, the velocity contribution to the lift force outweighs increases in the lift coefficient – allowing for SLF with reduced angles of attack. Note that the vehicle is limited to angles-of-attack on the [0, 15] deg interval, while the airspeeds vary between [12, 37] m/s.

Fig. 5.3 displays the same operating points projected in the δe-α plane, revealing that nearly all discovered trim conditions have negative elevator deflection. An alternative way to interpret Fig. 5.3 is the set of achievable SLF conditions with a stuck elevator fault of varying magnitudes. Each vertical slice of the flight envelope indicates the achievable angle of attack range when the elevator is stuck at a given position. Hence, *Ibis* could not be trimmed in SLF with an

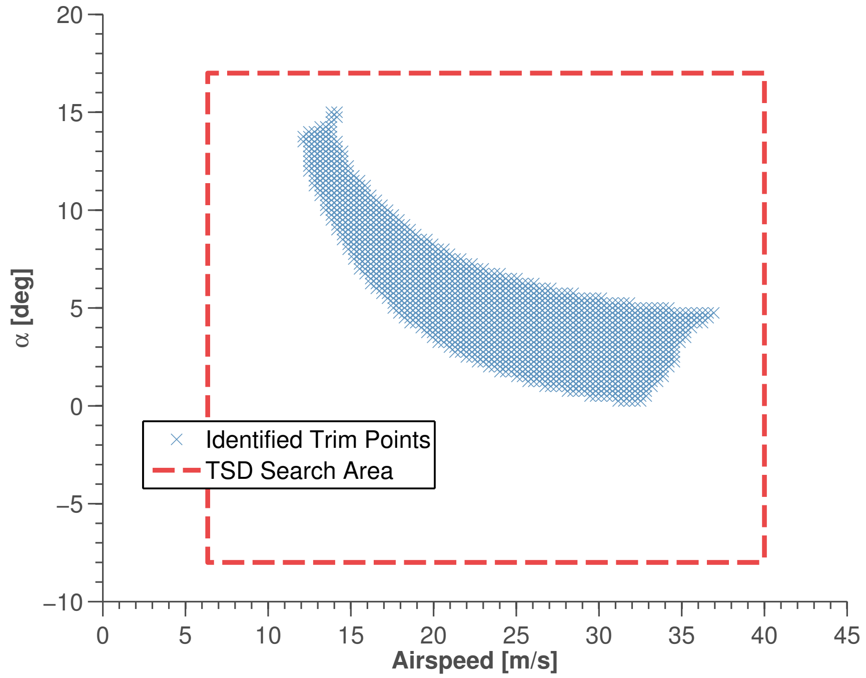


Figure 5.2: Achievable SLF conditions identified via trim state discovery for *Ibis* aircraft.

elevator stuck at any positive deflection. A small number of operating points are achievable for $\delta e <$ -15 deg, but these points are not contiguous with the rest of the flight envelope. One could infer that attempting to operate at $\delta e <$ -15 deg may be difficult when facing environmental and other disturbances in a real operating environment. Also note that there are few achievable trim conditions with δe near 0. This suggests that SLF likely cannot be maintained with a floating surface elevator failure. Thus, a large portion of the [-25, 25] deg allowable elevator deflection range falls outside of the envelope where SLF is achievable.

The values of all inputs across the SLF flight envelope can be viewed in a similar format as Fig. 5.3. Those plots are omitted in this dissertation, but *Ibis* requires significant aileron and flap deflections as well as large throttle inputs to reach many locations within the SLF flight envelope. Hence, even though much of the available elevator control authority is unused, the same cannot be said for all inputs.

One way to check the results of the trim state discovery is to examine experimental data from prior *Ibis* research flight tests. Flights where the aircraft was mostly flying at SLF, or near-SLF, conditions were selected, and the data was extracted for when the vehicle was flying with the autopilot engaged. This flight test data can be plotted in the same planes as the operating points identified via trim state discovery to provide a comparison between the simulation and

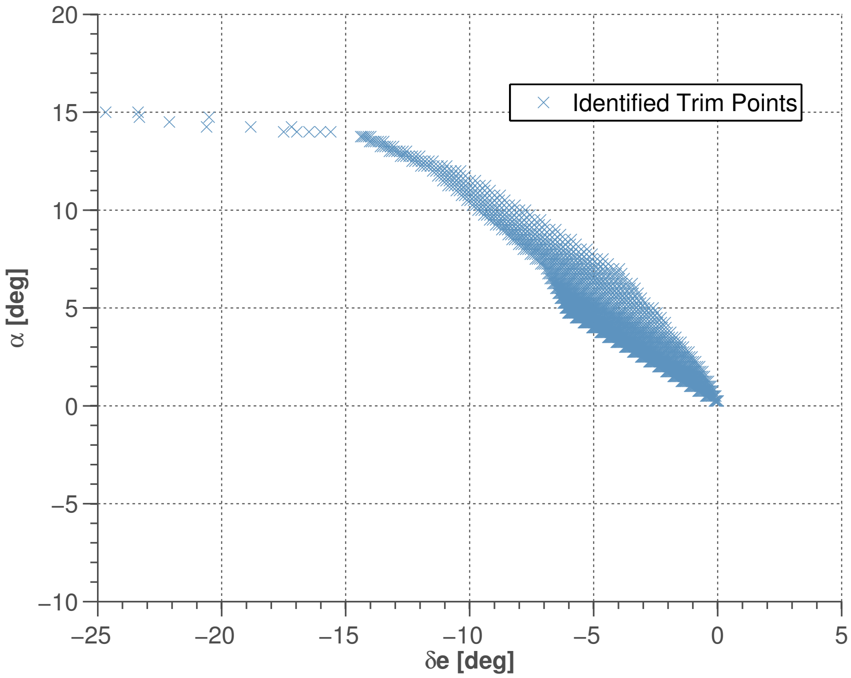


Figure 5.3: Achievable SLF conditions identified via trim state discovery for *Ibis* aircraft projected in δe-α plane.

experimental results. Fig. 5.4 shows the behavior of *Ibis* in the V-α plane. Note that the flight data is aggregated; some of the data points are from flight conditions other than SLF. Data obtained at some of these other conditions is likely responsible for some of the flight envelope excursions shown in Fig. 5.4. In general, there is large agreement between the experimental data and the flight envelope obtained via trim state discovery simulation. As stated previously, most *Ibis* flight tests are conducted at $V = 23\,\mathrm{m/s}$, and consequently, the flight test data is clustered near that airspeed. Further explanations for disagreements between the simulation and experimental data could include (i) model or simulation inaccuracy; (ii) unmodeled environmental effects encountered during flight tests (e.g., wind gusts); (iii) sensor inaccuracy for the angle-of-attack vane and pitot-static probes used to collect measurements of α and V. Another way to examine the question of system vulnerability to elevator faults is to view the relative incidence of different elevator deflections during SLF. Fig. 5.5 shows the experimental flight test data in the δe (commanded) - α plane. The commanded elevator positions are predominately clustered in the interval [-11, -2] deg, albeit at a lower angle of attack than the trim state discovery predicted would be possible. The data suggests that if an elevator is going to get stuck in SLF, it is likely to get stuck in a position where SLF could plausibly be maintained.

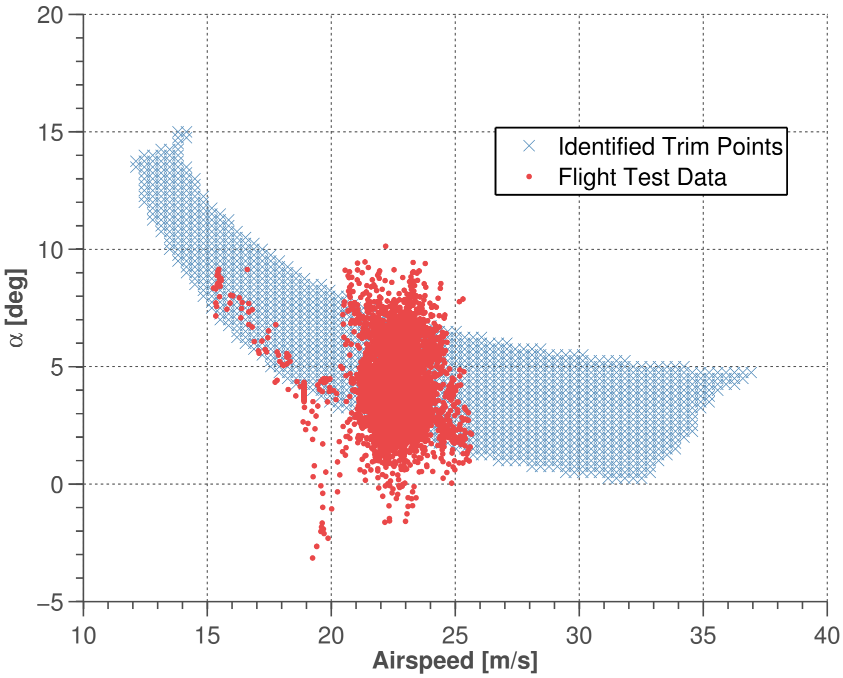


Figure 5.4: *Ibis* fllight envelope excursions during flight tests. Note that the flight data is aggregate, and includes some data from flight conditions other than SLF.

While servos are most likely to stick at their last functioning position due to an internal gear failure, other events may cause servos to become stuck outside their typical range of operation. In these rare cases, reconfiguration or adaptation to maintain SLF may not be possible, and alternative emergency measures must be considered.

When the experimental data is visualized in this space, flight envelope excursions are more evident. It may be the case that the trim routine built into the trim state discovery yields a conservative estimate of the flight envelope due to numerical issues associated with the trim optimization. That is, the procedure may to successfully trim the vehicle at some locations where it is actually capable of flying. There is, perhaps, no efficient way to identify the entire flight envelope via experimentation, so the information obtained via simulation becomes the best available information regarding the vehicle's achievable operating regime. To reiterate, however, the comparisons shown in Fig. 5.4-5.5 are indirect; the flight data is not exclusively obtained at trimmed SLF conditions and likely has its own margin of error.

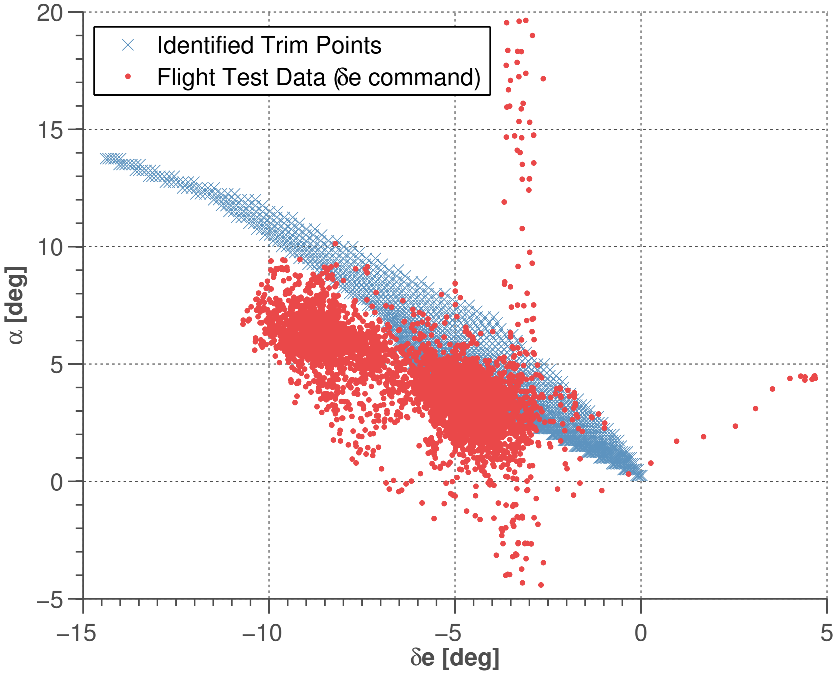


Figure 5.5: *Ibis* flight envelope excursions during flight tests. The flight data indicated is for the commanded elevator deflection rather than the actual position (which was not measured during flight testing). Note that the flight data is aggregate, and includes some data from flight conditions other than SLF.

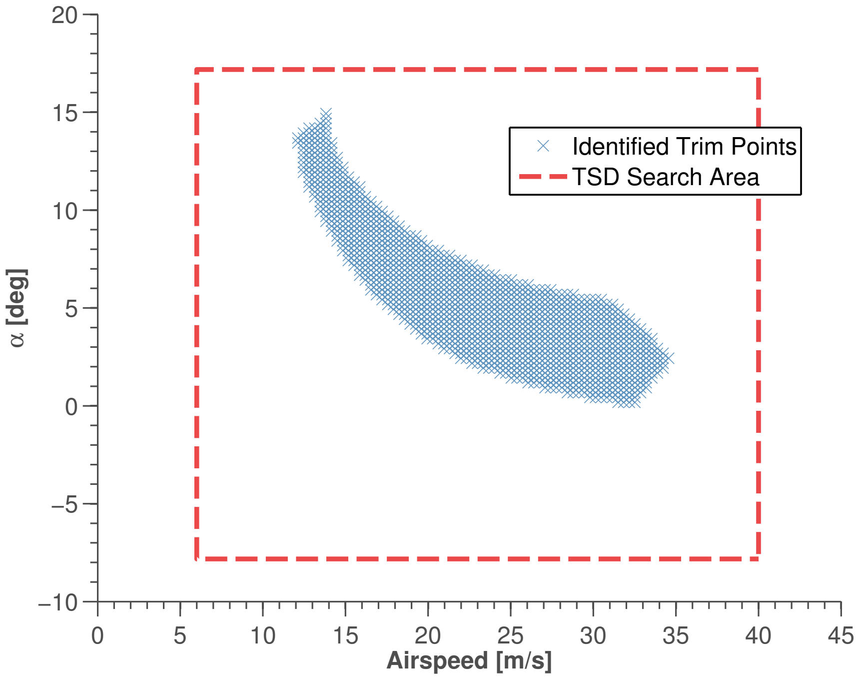


Figure 5.6: Achievable SLF conditions identified via trim state discovery for *Baldr* aircraft.

5.3 Linear Analysis

While significant insight can be gained by simply visualizing the shape of the achievable flight envelope, a linear analysis of the system dynamics at each operating point can reveal even more information. The nonlinear equations of motion are linearized about all operating points to obtain a linear model set, where each model in the set corresponds to a particular SLF trim condition.

For the linear analysis, a new trim state discovery is first conducted for *Baldr* to take advantage of the enhanced aerodynamic modeling described in Section 4.2. A particular control surface failure mode is configured for the trim state discovery procedure to see the effects upon the achievable flight envelope. In this case, each trim target is set such that $\delta eL \equiv 0$, which represents a floating surface, or alternatively, a stuck surface. Fig. 5.6 shows the 1,212 achievable SLF conditions identified for *Baldr* with a failed left elevator. The overall shape of the flight envelope is similar to that of *Ibis* (Fig. 5.2) when viewed in the V-α plane. At higher airspeeds, *Baldr* has slightly fewer achievable operating points than *Ibis*. It turns out, however, that control inputs to the flaps and ailerons are significantly smaller than for the *Ibis* flight envelope due to the ability to decouple all control surface surfaces and make finer trim adjustments.

5.3.1 Open-loop Analysis

Gap Metric and Plant Changes Throughout Flight Envelope

A linear model is generated for each of these operating points. One question worth investigating is how the aircraft dynamics vary across the flight envelope. The gap metric is a mathematical measure of the dissimilarity between two systems with equivalent input and output dimensions [26, 27], and it can be used to ascertain variations within the linear model set. The gap metric values are nondimensional, ranging from 0 to 1. A small value implies that any controller that can stabilize system G_1 is likely to stabilize system G_2, with similarity between the closed-loop gains of the two systems. A gap value of 0 indicates that $G_1 \equiv G_2$, while a gap value of 1 implies that G_1 and G_2 are dissimilar and are unlikely to be stabilized by the same controller [28].

During flight testing, the UMN FRP aircraft typically use a classical, loop-at-a-time autopilot designed for SLF with V = 23 m/s and h = 100 m (the same condition used to initialize the trim state discovery procedure). The linear model associated with this preferred, nominal flight testing condition (denoted P_0) can be used as the baseline system for the gap metric computation throughout the linear model set. The resulting gap values between P_0 and $\{P_1, P_2...P_n\}$, where n is the number of linear models in the set, will measure the dissimilarity of the system dynamics across the identified flight envelope. Moreover, it will provide insight regarding how well the baseline classical controllers may be expected to perform in the event that the aircraft must operate at some off-nominal SLF condition.

Figure 5.7 shows the computed gap values across the SLF envelope. There is a discernible 'sweet spot' near the nominal SLF condition where the linear models are all roughly equivalent. For models obtained at airspeeds outside the [20, 25] m/s interval, however, there is a precipitous increase in the gap values. At the slowest airspeeds, the model dynamics are starkly different and the gap values approach 1.

Frequency Response and Relative Efficacy of Control Inputs

By examining the frequency response characteristics across the linear model set, one can obtain a direct view of how significantly the input-output relationship can change inside the flight envelope. The magnitude frequency responses are computed for each model, and the pitch control authority of two longitudinal effectors (i.e., right elevator, throttle) is compared to that of the right aileron. Recall that the ailerons, usually inversely coupled, were decoupled for the

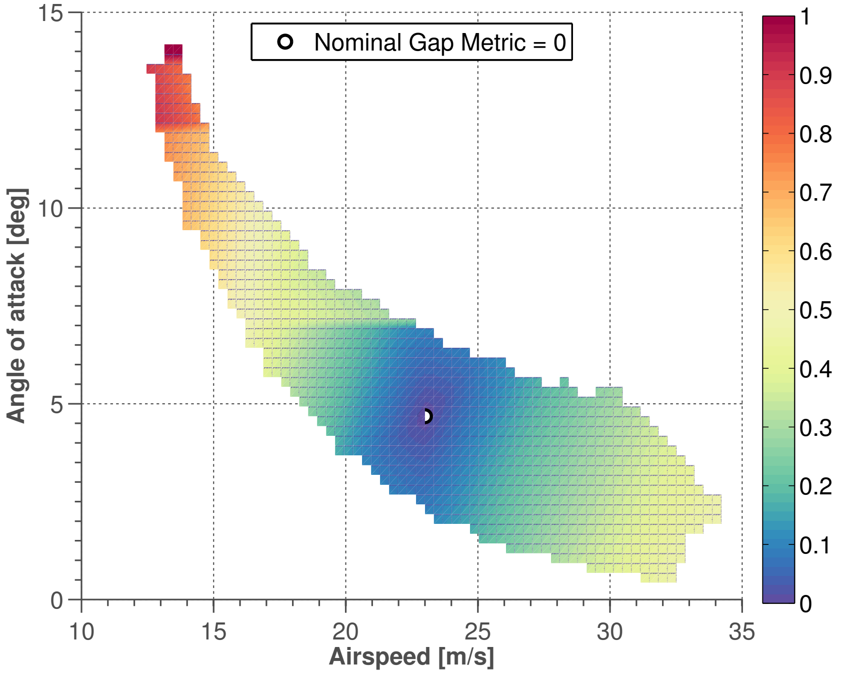


Figure 5.7: Gap values throughout the SLF envelope when compared to nominal SLF baseline condition.

Baldr trim state discovery. A reconfigured control law may attempt to create additional longitudinal control authority by decoupling the ailerons so they may act as flaperons or spoilerons. Fig. 5.8 shows the frequency responses from $[\delta eR, \delta t, \delta aR]$ to θ in the frequency band relevant for vehicle control, (ω = 0.1-10 rad/s). The magnitudes for each frequency response are scaled according to the remaining allowable control action in each channel at the respective flight condition. For example, the throttle would have more available control authority with a trim value of 0.5 versus 0.95 due to the impending saturation in the latter case. This adjustment is reflected in the responses plotted in Fig. 5.8.

For the relevant frequency band, the throttle and aileron both have significantly poorer longitudinal control authority than the elevator, and the control authority is even poorer in the pitch rate channel. Hence, with a disabled elevator, there can be little active damping of pitch rate disturbances using alternative effectors. This result suggests that any candidate adaptive control algorithm may have serious difficulty successfully controlling the vehicle if the elevator becomes completely disabled.

A close examination of Fig. 5.8 shows that there are discrete changes in model dynamics in different regions of the flight envelope. By visualizing the magnitude frequency response

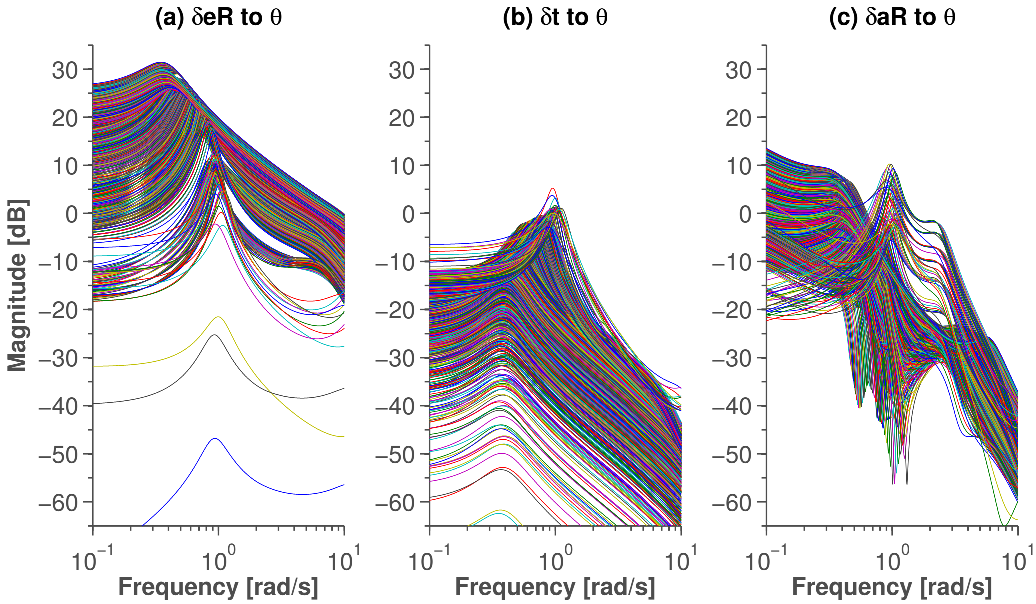


Figure 5.8: Magnitude frequency response from (a) right elevator to pitch angle; (b) throttle to pitch angle; and (c) right aileron to pitch angle. The magnitudes are normalized according to the remaining allowable input before saturation

data differently, those changes can be localized to those different parts of the flight envelope. A simple way of quantifying pitch control authority for a given model in the set is to compute the H_∞-norm from the input channel of interest to the pitch angle output within the relevant bandwidth. The H_∞-norm is the peak gain over this frequency domain, and the greater the norm, the better control authority that input-output channel exhibits over the system. Using this approach, it is straightforward to assess the relative pitch control authority for each effector at each location in the flight envelope.

Fig. 5.9 shows the ratio of throttle pitch control authority to right elevator pitch control authority (i.e., the ratio of their respective H_∞ norms). Throughout almost the entire flight envelope, the throttle channel has significantly less pitch control authority than the elevator. Only at the lowest airspeeds does throttle exhibit parity with the elevator, and it is ineffective at airspeeds near or above that of nominal flight testing condition. At high airspeeds the throttle exhibits almost no pitch control authority at all. These are notable observations because the throttle is the secondary longitudinal effector after the elevator. In the event of a fully disabled elevator, it seems unlikely that active throttle control would be effective for maintaining SLF conditions, much less more demanding maneuvers.

Similarly, the ratio of right aileron pitch control authority to right elevator pitch control

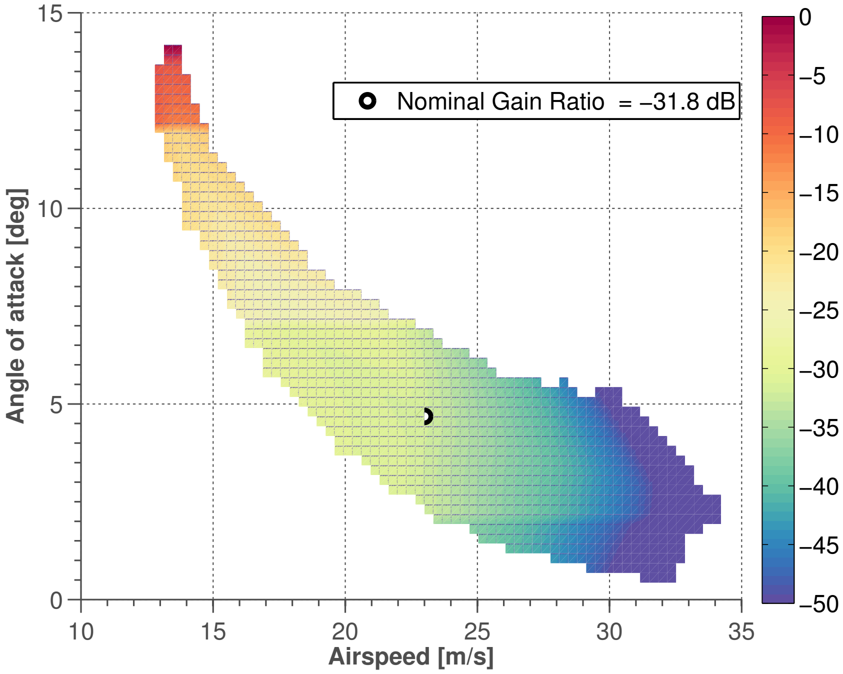


Figure 5.9: Relative pitch control authority,

authority can be viewed (Fig. 5.10). The data indicates that the ailerons also have the greatest pitch control authority at low airspeeds, and also begin to exhibit greater authority at the highest achievable speeds. In the center of the flight envelope, where flight tests are typically conducted, neither the aileron nor the throttle has significant longitudinal control authority; much of the achievable trim space has no feasible control reconfiguration if the entire elevator is stuck. Displaying this information in this fashion highlights the difficulty of using adaptive/reconfigurable control and suggests that there are some fundamental physical limitations to these approaches that cannot be ignored.

5.4 Summary

This chapter demonstrated that trim state discovery can be used to complement a conventional failure modes and effects analysis. Determining the achievable operational regime in various states of system health can provide detailed information regarding specific failure effects and system vulnerabilities. Additionally, familiar tools from linear systems analysis can be applied to predict how the system dynamics will change throughout the flight envelope. This analysis indicates that there is significant vulnerability to elevator faults, especially for the coupled single-redundancy architecture implemented for *Ibis*. *Baldr* fares better with its

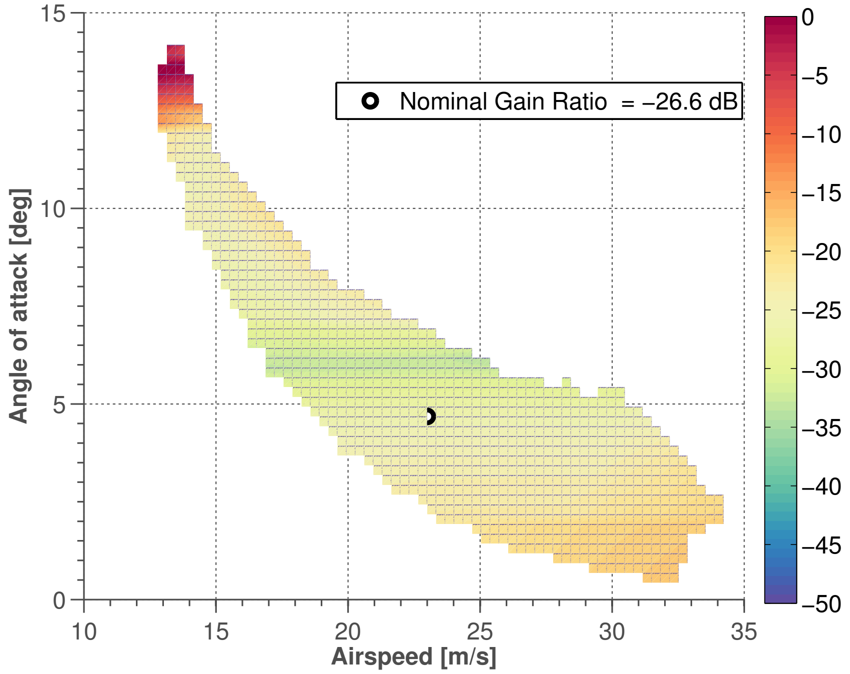


Figure 5.10: Relative pitch control authority,

independently-actuated, decoupled surfaces, but the *Ultra Stick 120* airframe appears to have limited potential for exotic reconfiguration approaches. This result is likely common for low-cost, hobby-grade vehicles that do not have advanced airframe design or hardware redundancy. Dynamics that are more suitable for adaptive control may need to be 'designed-into' the original airframe for the best chances of success.

The results of this analysis reveal several avenues to improve FRP vehicle reliability and extend research efforts at UMN. The enhanced modeling and trim state discovery tools should be valuable for prototyping and implementing reliability algorithms such as flight envelope protection that can be tailored to phase of flight and assessing the feasibility of active reconfiguration algorithms. Moreover, this type of analysis can serve as an important feedback loop to the FMEA and FTA and inform future design choices.

The trim state discovery tools designed for this research are fully compatible with the UMN FRP open-source simulation package, and researchers can freely use them for their own work. While the analysis in this chapter was limited to a single class of flight conditions, steady wings-level flight, any class of reference flight conditions can be studied (e.g. coordinated turns, steady descent, steady climb) in conjunction with any control surface impairments of interest.

Finding ways to better integrate reliability engineering tools with control system tools will be increasingly important as the demand for low-cost, reliable UAS soars.

Chapter 6

Fault Detection and Isolation for Control Surface Impairments

Designing for improved reliability can help make UAS more reliable and viable, but there will inevitably be in-flight failures that must be detected and accommodated for truly safe operation. Designing fault detection and isolation (FDI) algorithms that are suitable for UAS applications is an important plank in a comprehensive strategy to make UAS more fault-tolerant. Exacting reliability standards for safety-critical systems spanning the energy, biomedical, and transportation industries necessitate novel approaches to system monitoring and FDI. Advances are particularly important to the future of unmanned aviation, as the US Federal Aviation Administration has been required to safely integrate UAVs into the national airspace by 2015 [5]. To do so will require reliability certification procedures akin to those used for manned aviation.

Manned aircraft have stringent reliability requirements, allowing less than a single catastrophic malfunction per 10^9 hours of operation. For decades, manufacturers have achieved this reliability via a strategy of hardware redundancy (HR) throughout the flight control system design, encompassing multiple control surfaces, actuation systems, sensors, and flight computers [1, 2]. Voting algorithms typically check for consistency of behavior among HR subsystems and usually provide direct fault isolation. HR, however, carries a cost, payload, and power consumption penalty. The resources consumed by HR in the interest of fault-tolerance may not allow sufficient resources for other performance objectives (e.g. payload capacity, endurance, maneuverability). These constraints, coupled with the increasing acceptance of UAV technology by regulators, drives an increasing demand for low-cost, fault-tolerant aerospace systems.

An alternative to HR is analytical redundancy (AR), in which health monitoring software uses a system process model to generate residuals (the differences between estimates and measurements of the system states). A residual should be a good representation of the fault of interest yet insensitive to process and measurement noises. Residual evaluation algorithms involve processing and decisioning logic leading to a fault diagnosis [29]. Common techniques for residual generation include parity-equations, state-observers, or parameter estimations [30]. Data-driven approaches to FDI eschew model information in favor of signal processing techniques. Data-driven techniques are often used to detect discrete changes in a process. See [30, 31, 29] for a detailed treatment of model-based and model-free fault detection methods. A certifiable AR strategy may incorporate layered model-based and data-driven FDI approaches with system-level evaluative and reconfiguration procedures. Current vehicle health management approaches increasingly use prognostic algorithms to estimate useful component life, enabling condition-based maintenance to preemptively replace degraded components prior to failure. These techniques are primarily built on statistical, data-driven algorithms [32]. Prognostic approaches may be useful for fleet management and can reduce the incidence of certain faults, but any post-fault control reconfiguration strategy still requires rapid, accurate FDI.

The objective of the work described in this chapter is to design and apply robust model-based residual generation and data-driven anomaly detection approaches to a small, low-cost UAV platform. Extensive nominal and faulted closed-loop experimental flight tests are conducted with various control surface actuation fault modes selected for study. Closed-loop validation of the algorithm is performed because FDI research is often performed for open-loop systems which ignores significant influences that controller robustness may have on the detection performance [33]. Furthermore, using field-collected data enables practical evaluation of detection approaches in the presence of realistic sensor noise, exogenous disturbances, and unmodeled dynamics. Few model-based FDI algorithms have been applied to real data. This work expands the literature to include validation of model-based and data-driven techniques for both simulation and experimental data.

A robust observer residual generation algorithm based on the H_∞ model-matching architecture [34, 35] is developed using a linear model of the UAV. Next, a statistics-based, data-driven algorithm is developed to operate exclusively on raw flight test data without knowledge of the system dynamics. The main contribution of this work is that experimental data allows for a side-by-side comparison of FDI techniques arising from different philosophies of system monitoring within a realistic operational environment. Existing literature has not shown the experimental application of such techniques for comparable systems, and recent regulatory changes have made practical research efforts in this area imperative. This work further provides a FDI benchmark problem for a small UAV with simulation and real flight data for other researchers to test their algorithms. Conclusions about strengths and weaknesses of each approach for the UAV platform and thoughts on future FDI development can be derived from this research.

6.1 Experimental Scope

The aircraft used in the FDI design and experiments is the FRP *Ultrastick 25e* [36]. Note that the *Ultra Stick 25e* airframe uses the conventional control surface configuration described in Chapter 2 rather than the split/decoupled surfaces described later in this dissertation.

The research focus is limited to the lateral-directional aircraft dynamics; commanded aircraft maneuvers and injected faults were chosen to excite these dynamics. Hence, the flight data signals of interest are (1) roll angle command, ϕ_{cmd} [deg]; (2) roll angle response measurement, ϕ_m [deg]; (3) aileron deflection command, $\delta_{a_{cmd}}$ [deg]; (4) roll rate, p [deg/s]; (5) yaw rate, r [deg/s]. Data acquisition in-flight is at a sampling rate of 50 Hz. The aileron and rudder control surfaces are actuated by separate, identical servos. Two controllers are used for the FRP: a H_∞ pitch-tracking autopilot controls the longitudinal motion of the aircraft and a linear quadratic optimal roll angle controller described in detail in [37]. The pitch tracking controller takes pitch angle θ [deg], pitch rate q [deg], and airspeed TAS [m/s] in negative feedback. The roll angle controller takes as inputs the roll angle reference command ϕ_{cmd} as well as the three state measurements p_m, r_m, and ϕ_m (each in negative feedback). A state-space representation of the controller is given in [38].

The control and model-based FDI techniques used in this work require a linear model of the trimmed aircarft. The steady, level reference flight condition selected has an indicated airspeed of 17 m/s and angle of attack of 4.2 deg. The six degree-of-freedom nonlinear model is linearized about this condition and the longitudinal dynamics are decoupled. This yields a four state, two input, three output linear approximation of the FRP lateral-directional dynamics, G. The four states are (1) lateral velocity in body-frame, v [ft/s]; (2) roll rate, p [deg/s]; (3) yaw rate, r [deg/s]; (4) roll angle, ϕ [deg]. The two system inputs are deflections of the aileron δ_a [deg] and rudder δ_r [deg]. The three measurements of interest are p_m, r_m, and ϕ_m.

To accurately characterize the FRP, the linear UAV dynamics are augmented with linear component models in the simulation architecture (Fig. 6.1). The surface actuators with time delay, Act, are modeled with a 2x2 block diagonal, first-order system with 8 Hz bandwidth and a 40 ms first-order Pade time delay. The roll-angle controller is denoted C, and the FDI filter to be designed in Section 6.4 is denoted F. Details regarding the uncertainty model and H_∞ input/output signal weighting are presented in Section 6.4.

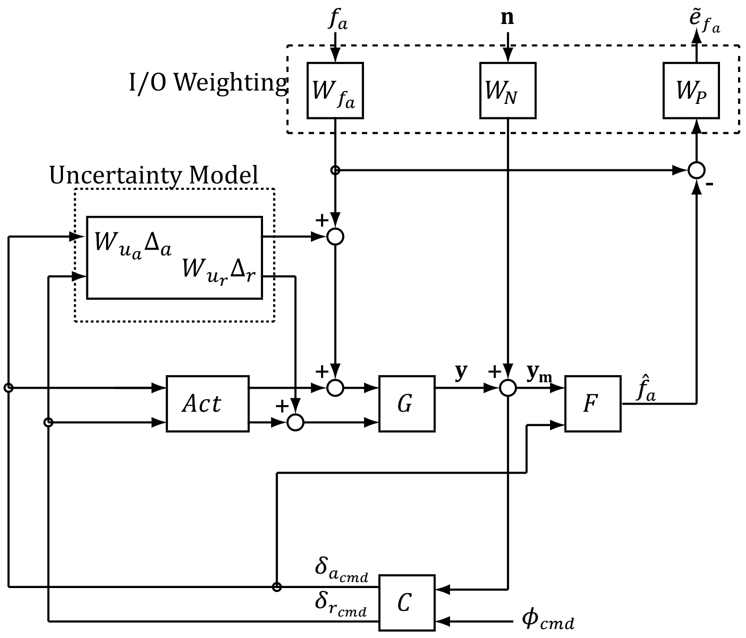


Figure 6.1: System Interconnection for *Ultrastick* UAV

6.2 Fault Scenarios Considered

Actuators and sensors provide the physical system inputs and measurement feedback, respectively, required for desired vehicle operation. Thus, undetected faults affecting these subsystems can make aircraft control difficult or impossible. This work focuses on detection of aileron control surface actuation impairments. A set of faults are chosen for this research that excite the lateral-directional dynamics of the aircraft and provide insight into the effects of aileron actuation faults. Specifically, four aileron faults were considered: (1) 5 deg aileron ramp; (2) 10 deg aileron ramp; (3) 5 deg aileron bias; (4) 10 deg aileron bias. A bias can be modeled as erroneous, instantaneous step in the control surface position. Input biases can cause rapid changes in aircraft behavior. An injected ramp fault involves a linear increase in the surface deflection over some time interval and represents a non-instantaneous degradation in the actuation subsystem. Ramps are slow-acting in comparison to bias faults and are more difficult to detect. Two fault magnitudes are selected in order to better understand magnitude influences on fault manifestation and detection.

6.3 Flight Testing

A flight test procedure is developed enabling repeatable aircraft maneuvers in the faulted and unfaulted flight conditions. The experimental data can be used by model-based and model-free detection techniques to provide vehicle health information. The reference flight condition described in Sec. **??** is chosen for the flights. The test maneuver is a 20 deg roll doublet. The ability of the autopilot to successfully track this reference signal will vary depending on which fault (if any) is engaged. The sequence of maneuvers commanded by the flight computer – beginning with time t_0 is as follows:

1. t_0: controller engaged, ϕ_{cmd} = 0 deg, θ_{cmd} = 5 deg (constant throughout flight)
2. t_0 + 2 sec: 20 deg doublet with 4 sec period
3. t_0 + 6 sec: doublet complete
4. t_0 + 8 sec: fault injection starts
5. t_0 + 10 sec: 20 deg doublet with 4 sec period
6. t_0 + 16 sec: 20 deg doublet with 4 sec period
7. t_0 + 20 sec: ϕ_{cmd} = 0; ramp fault reaches final value

Each aileron fault mode received four flights (except the 10 degree aileron bias with three flights), and a complete sequence constitutes a single flight. The roll angle responses during these maneuvers are shown in Fig. 6.2.

Subplot 6.2(a) shows the roll doublet tracking for four test flights of the unfaulted UAV. Note the variation in the response of the bank angle ϕ during the unfaulted flight. Any fault detection filter, model-based or data-driven, must be robust to these variations as they are not due to a faulty vehicle. This is part of the challenge of applying fault detection algorithms to real flight data. Subplots 6.2(b)-(e) show tracking for the 5 deg ramp, 10 deg ramp, 5 deg bias, and 10 deg bias fault scenarios, respectively. Note that for ramp faults the tracking degradation does not occur until several seconds after the fault is injected at $t = 8$ s. The bias faults have an adverse impact immediately after the fault injection, though the flight control system begins to effectively reduce the magnitude of the tracking error throughout the remainder of the flight duration. Finally note that, as expected, the larger magnitude faults more adversely impact the fault tracking performance than do smaller magnitude faults.

The linear model described in Sec. **??** is an accurate approximation of the nonlinear flight dynamics of the FRP at the flight condition of interest. Fig. 6.3 compares the simulated and

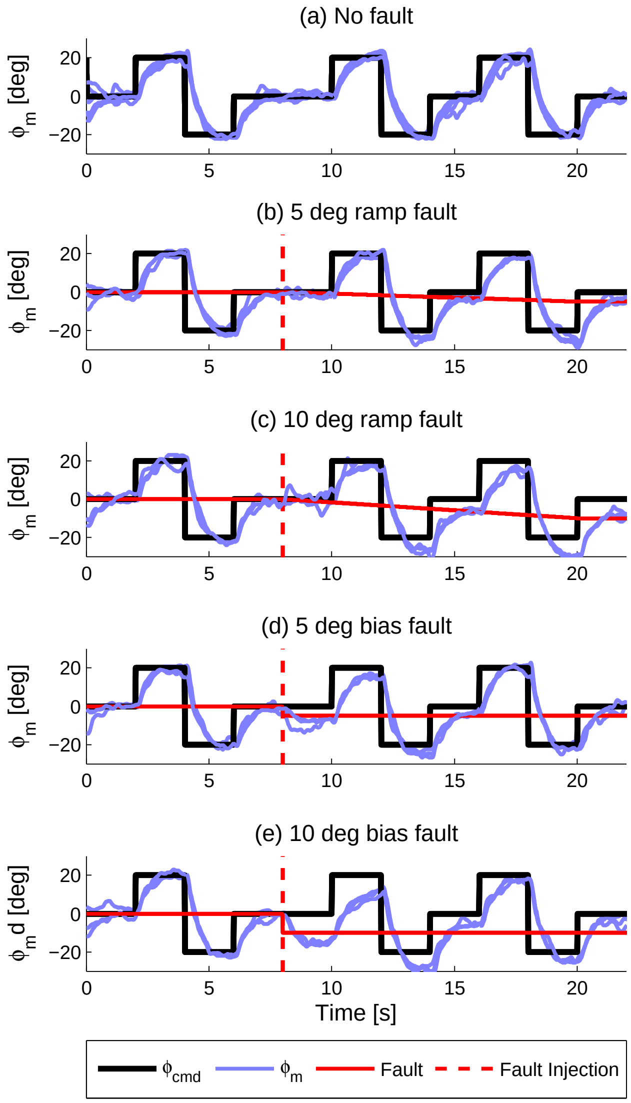


Figure 6.2: Experimental UAV Flight Test: Roll rate response for multiple flights with unfaulted and fault cases.

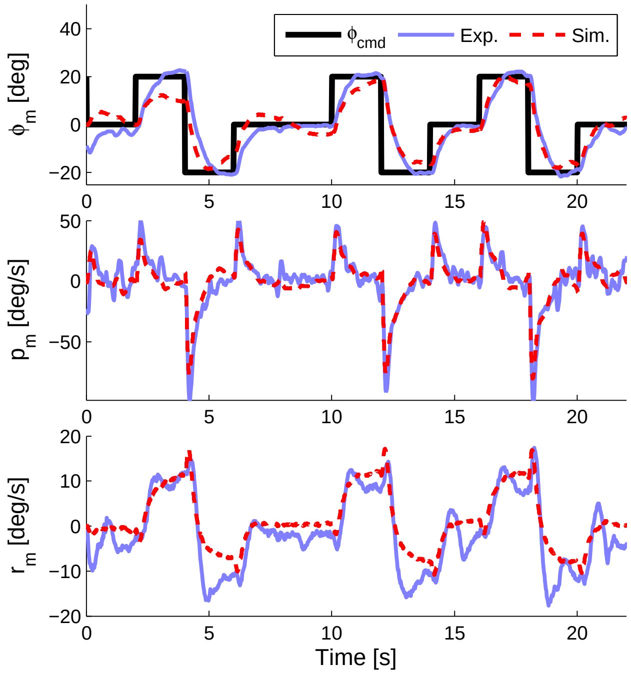


Figure 6.3: Experimental UAV Flight Test vs. Linear Simulation: Output measurements for unfaulted flights.

actual aircraft output measurements for a single unfaulted test flight. Note that the actual roll angle and aileron commands used during the flight test are provided as inputs to the simulation. There is broad agreement in the measurement outputs, and much of the disparities that do exist are the result of a slightly out-of-trim aircraft at the initialization of the roll doublet commands. An aircraft operating near a steady, level reference flight condition is not a highly nonlinear system; hence, a linear approximation of the vehicle dynamics will suffice for FDI algorithm development. The experimental flight data is important to constructing and comparing the model-based and model-free fault detection methods described in the subsequent sections. Moreover, the online availability of flight data recorded from these flight tests and other FRP research activities is an essential component to the UAV FDI benchmark problem.

6.4 Model-based Fault Detection

Several observer-based FDI techniques have been proposed and validated using simulation results [39, 35], yet there is a dearth of existing literature regarding application of these techniques to data from real, operating engineering systems. In particular, the aerospace flight control community has not yet undertaken concerted efforts in this regard. Model-free health management approaches are increasingly being applied for prognostics purposes that may not provide the post-fault information necessary for automatic control reconfiguration strategies [32, **?**, 40]. Data-driven anomaly detection for single autonomous vehicles has not received focused attention. Hence, the application of model-based and data-driven FDI techniques to an experimental autonomous system such as the UMN FRP is a positive addition to the existing literature. This section details the factors motivating the choice of model-based detection algorithm, an overview of the robust filtering methodology, the mathematical model representing the FRP, and the synthesis of the FDI filter.

6.4.1 Design Considerations

There are three broad model-based design approaches.

Integrated design

The controller and filter are designed simultaneously to satisfy objectives. Because the controller and filter may be optimized to meet particular objectives, integrated design can frequently yield excellent results. For many existing systems with fixed control laws, however, an engineer may not be able to alter the controller. Hence, the integrated approach is often infeasible.

Discrete closed-loop design

The controller and filter are designed separately. This approach is often used when integrated design is not possible since it allows for filter design with existing closed-loop systems. The filter is developed using a closed-loop system with the controller behavior is embedded into the FDI filter. This approach was applied for air data sensor faults in [41, 42, 43].

Discrete open-loop design

The controller and filter are designed separately, and the filter is synthesized based upon the open-loop system dynamics. This approach may not achieve the same performance as the previous methods, but it does provide the advantage of flexibility: any controller can be used for a given plant since the filter is synthesized independently. This work develops a discrete open-loop design that will provide flexibility for future control research with the FRP.

For a well-performing FDI system, a candidate residual generator should exhibit the following desirable attributes:

- Sensitivity to certain faults and insensitivity to others
- Robustness to modeling uncertainty
- Good attenuation of external disturbances and noise at the FDI filter output
- Ability to distinguish faults that occur simultaneously
- A non-zero steady-state filter output in response to a non-zero fault; zero output otherwise

These attributes account for many of the advantages of model-based techniques in general.

Recall that the FRP is a small, low-cost UAV. Such platforms are often outfitted with low-quality sensors and the system dynamics are not necessarily well-known as high fidelity models can be expensive to develop. Hence, robustness to model uncertainty, noise, and disturbances is important for the FRP and similar platforms. Robust H_∞ model-based techniques [44, 35] explicitly address model uncertainty concerns and are applicable for open-loop or a closed-loop systems. These approaches are well-suited for the UAV FDI problem.

6.4.2 H_∞ FDI Formulation

The standard H_∞ model-matching approach [34, 35] is a robust filtering method used to design the FDI filter. Other robust filtering approaches have been used in [45, 46]. This technique seeks to minimize the H_∞ norm of error signals of interest over the convex set of controllers (or an observer, in this case), $F(s)$.

The specific design objective is to minimize the infinity norm (i.e worst-case gain) of the transfer function matrix from aileron fault $\mathbf{f}$ and sensor noise inputs $\mathbf{n_p}$, $\mathbf{n_r}$, $\mathbf{n_\phi}$ to the weighted fault estimation error, $\tilde{\mathbf{e}}_\mathbf{f} = W_P(\mathbf{f_a} - \hat{\mathbf{f}})$, where $\hat{\mathbf{f}}$ is the aileron fault estimate produced by the FDI filter F (Fig. 6.1). Many standard algorithms to solve this problem are available in the literature [34, 47]. Weighting functions are used to shape the frequency content of disturbances, errors, and the filter performance objective across frequency. This fault detection architecture is easy for the control engineer to adopt and adapt given its similarity to H_∞ control. Moreover, its residual generation satisfies many of the attributes desired for a FDI system.

1) Uncertainty Modeling: The dynamics of the UAV vary by flight condition. Any FDI algorithm must provide accurate estimates despite these changing dynamics. The UAV can have small airspeed changes due to disturbances, maneuvers, or unmodeled effects, hence the FDI filter should be insensitive to small airspeed variations not resulting from an aileron fault. An input multiplicative uncertainty model is used in the H_∞ framework to overbound the dynamic uncertainty in the linear model at different equilibrium flight conditions ranging from V = 16-20 m/s, with the nominal equilibrium at V = 17 m/s. For more significant changes in airspeed, model accuracy and filter sensitivity to real faults are reduced. Thus, the quality of the diagnosis is degraded, and a filtering approach that incorporates scheduling may prove more effective [42, 43].

The plant model set can be represented by $G_\Delta(s) = G(s)[I + W_u(s)\Delta(s)]$ where $G(s)$ is the nominal plant dynamics and $\Delta(s) := \{\mathbf{\Delta} : \Delta \in \mathbf{C}_{2x2}, \bar{\sigma}(\Delta) \leq 1\}$ is the set of all complex, norm-bounded uncertainty. Note that the uncertainty in each channel is modeled as being entirely independent of the other. $W_u(s) = \text{diag}([W_{u_a}(s) W_{u_r}(s)])$ is a weighting function that describes the upper bound on percentage of system gain variation from the corresponding control input across all frequency. The UAV dynamics are linearized for straight, level flight for airspeeds of 16-20 m/s at 1 m/s intervals, yielding a set of linear, time-invariant models. The frequency responses of this model set from the inputs to the outputs of interest (p,r,ϕ) are examined, and an appropriate W_u is computed to overbound the model set. The most significant changes in the aircraft dynamics occur at low frequencies concerning the gain from both aileron and rudder deflection to roll rate. Selecting weighting functions to overbound this variation with airspeed will overbound the variation in the other channels of interest as well. Two fourth-order weighting functions were calculated using the Robust Control Toolbox [28] to provide a tight bound for the dynamics variations.

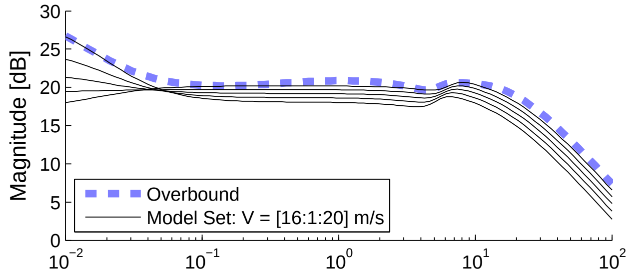
$$W_{u_a} = \frac{0.5338s^4 + 2.427s^3 + 21.7s^2 + 4.234s + .1471}{s^4 + 13.58s^3 + 71.06s^2 + 24.67s + 0.1613} \tag{6.1}$$

$$W_{u_r} = \frac{0.5675s^4 + 3.591s^3 + 12.03s^2 + 1.239s + 0.03823}{s^4 + 3.845s^3 + 63.18s^2 + 6.248s + 0.04674} \tag{6.2}$$

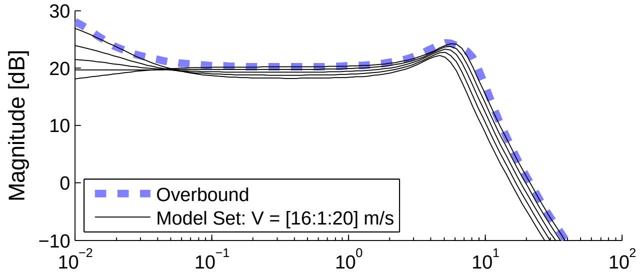
Fig. 6.4(a)-(b) shows the magnitude frequency response of the model set while Fig. 6.4(c) shows the the input uncertainty weights W_{u_a} and W_{u_r}. Note that the weights chosen are indicative of increased levels of model uncertainty due to changing dynamics at very low and high frequencies.

2) Fault, Sensor Noise, and Performance Weighting: Several weighting transfer functions are defined: aileron fault weight W_f, roll rate sensor noise weight W_{n_p}, yaw rate sensor noise weight W_{n_r}, roll angle sensor noise weight W_{n_ϕ}, and performance weight W_P. The frequency response of the aileron fault is modeled by a first order low-pass filter W_f applied to the injected fault signal. As fast faults should not significantly affect the UAV response, the low-pass rolls off for frequencies exceeding the bandwidth of the UAV dynamics. The W_f DC gain corresponds

(a) Frequency Response: δ_a to p



(b) Frequency Response: δ_r to p



(c) Input Uncertainty Weights

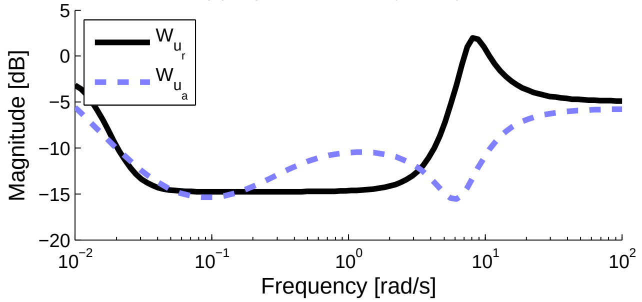


Figure 6.4: Magnitude frequency responses for overbounded lateral model set and corresponding multiplicative input uncertainty weighting functions.

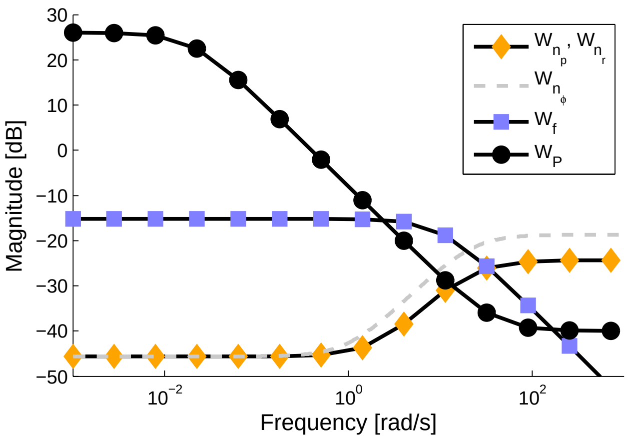


Figure 6.5: Frequency Response: Magnitude of fault, noise, and performance weighting functions.

to a 10 degree expected maximum δ_a fault. The noise weighting functions are determined based on sensor manufacturer specifications. Sensor noise has mostly high-frequency content, and the yaw-rate gyroscope noise model, W_{n_r}, reflects high frequency noise up to 3.5 deg/s. The roll rate gyroscope noise model, W_{n_p}, uses increased gain as a tuning parameter to drive the high frequency poles of the FDI filter to lower frequencies. The performance weighting function, W_P, acts on the aileron fault estimate error. It is selected to shape the closed-loop filter sensitivity function, affecting the steady-state error, bandwidth, and the high frequency roll-off. The performance weight at a given frequency is inversely related to the desired maximum fault estimation error magnitude. W_P is chosen to meet desired performance specifications of less than 3% fault estimation steady-state error (i.e. low frequency gain of W_P is ≤ 33.5) and rise (detection) time of less than 20 seconds. These weights used for FDI filter synthesis are shown in Fig. 6.5.

3) FDI Filter Synthesis: The interconnection shown in Fig. 6.1 is used to generate a weighted generalized plant. The H_∞-synthesis algorithm is applied to this generalized plant to yield a the 20-state filter F with one output, $\hat{f}$, and four inputs, $[p_m \; r_m \; \phi_m \; \delta_a]$. A 10-state FDI filter is derived using balanced realization model reduction. A first order low-pass filter with a 2 rad/s bandwidth is applied to the output of F to further limit high frequency variations.

6.5 Data-driven Anomaly Detection

This section details the factors motivating data-driven detection and the development of the applied methodology.

Model-based FDI techniques can be designed to reliably detect known failure modes, but their performance can degrade for unexpected faults. Data-driven methods, however, do not utilize knowledge of the system dynamics. This approach to system monitoring allows data-driven algorithms to sometimes detect anomalies that may not be properly estimated by a model-based FDI algorithm. Data-driven techniques also have the advantage of avoiding expensive and time-consuming engineering necessary for high-fidelity system identification, model validation, and robust filter design [48, 49]. However, eschewing insight regarding the system behavior can lead to results that are misleading or insufficient for detecting and identifying faults. Thus, model-based and data-driven techniques should be seen as complementary. The proposed data-driven approach for this research should operate using only the flight data gathered during the *Ultrastick* flight tests to allow the performance of the algorithm to be compared and contrasted with that of the H_∞ FDI filter. Moreover, unlike methods that are commonly selected for static systems or which are chosen for prognostics purposes, the desired data-driven technique will provide a rapid indication of anomalous behavior subsequent to the incidence of a fault. In this respect, the data-driven approach would shadow the model-based approach and provide information which could be used for residual evaluation, fault isolation, and control reconfiguration.

The selected data-driven technique requires a set of unfaulted training data collected during the flight tests. Beginning at the t =1 s time sample, a 1 s trailing time window is defined for each unfaulted flight dataset. A baseline parameter b_{train} is defined as shown in Equation 6.3, where $\mathcal{F}$ is the fast Fourier transform (FFT). FFT algorithms can be implemented on small computers such that they may operate on-line in order to produce computations for the data-driven anomaly detector, and as such, they are well suited to the application of interest here.

$$b_{train} = \left|\mathcal{F}\left(\left|\phi_{cmd_{train}} - \phi_{m_{train}}\right|\right)\right| \tag{6.3}$$

For each successive time sample until the conclusion of the flight, b_{train} is computed. To properly capture the dynamic response of the aircraft in this data, the first five discrete Fourier transform (DFT) components are used while the higher frequency components are discarded. The frequencies corresponding to these components are [0, 4.91, 9.82, 14.73, 19.64] rad/s, encompassing the bandwidth of the UAV lateral dynamics. The mean, μ_{train}, and standard deviation, σ_{train}, are computed from the DFT components over each trailing window in order to establish a statistical representation of the expected nominal aircraft behavior at each point during the flight. The test (i.e. faulted) flight data is analyzed by computing a five-component

b_{test}:

$$b_{test} = \left|\mathcal{F}(\left|\phi_{cmd_{test}} - \phi_{m_{test}}\right|)\right| \tag{6.4}$$

A Z-test is then performed. The null hypothesis is that the test flight DFT data b_{test} comes from a distribution with mean μ_{train} and standard deviation σ_{train}. Confirmation of the null hypothesis is indicative of unfaulted flight; rejecting the null hypothesis is be indicative of an anomaly affecting the test flight data for a certain confidence level. The Z-test generates a P-value, i.e., the probability of observing the given result, or one more extreme, by chance if the null hypothesis is true. Small values of P indicate questionable validity of the null hypothesis. The reciprocal of the P-value is used as an initial anomaly score on a log plot so that more disparate test and training data yields a higher anomaly score. A simple threshold mechanism is constructed by applying further scaling such that three standard deviations of the anomaly scores for the unfaulted training set corresponds to a final scaled anomaly score of 1. Scores exceeding 1 are strongly indicative of disparities between the test data and training data.

With existing computational resources suitable for small, low-cost flight systems and a preloaded training data set, these statistical computations may be performed on-line. This enables real-time implementation of the data-driven anomaly detection algorithm.

6.6 Flight Test Experimental Results

The first section presents model-based and data-driven FDI results in linear simulation. A comparison is made between the algorithms based exclusively on data collected from faulted flight tests as described in Sec. 6.2-6.3.

6.6.1 Linear UAV Simulation Performance

It is useful to examine the performance in simulation with the linear model with which the model-based filter was initially designed. The closed-loop UAV linear model trimmed about the reference flight condition is constructed. A desired fault profile is specified (e.g. unfaulted, 5 degree ramp, etc.), and the autopilot input ϕ_{cmd} is taken from the corresponding flight data and used to drive the model. Similarly, the commanded aileron deflection $\delta_{a_{cmd}}$ is provided from the same flight test. The commanded rudder deflection $\delta_{r_{cmd}}$ was not logged during flight testing, and hence is computed by the simulation roll angle control law. Both surface commands are processed by the actuator and time delay models.

The control surface signals are further modified by including model uncertainty to verify the robustness of the FDI algorithm to unmodeled dynamics. The uncertainty norm bound is adjusted to describe the maximum allowable model dynamics variation across frequency. A constant uncertainty bound of 35% was selected. A random uncertainty in each channel was selected, leading to variation in the gain and phase of the nominal linear model. Significant estimation degradation was observed for larger uncertainty bounds. Additive sensor noise is applied to the linear model outputs associated with the state measurements. The filter F, in conjunction with the low-pass filter described in Sec. 6.4.2, generates the fault estimate.

Fig. 6.6 shows the simulated roll angle tracking–(a) and (c)–and fault detection performance–(b) and (d)–for two particular flight tests: a 5 degree aileron fault and a 5 degree bias fault. The simulated flight trajectory is a reasonable approximation of the experimental trajectory. The inclusion of uncertainty accounts for model errors which may affect the FDI filter. Despite model differences, both faults are rapidly and accurately detected by the model-based H_∞ FDI filter. The small spikes in the fault estimates are the result of the start or conclusion of a roll maneuver by the experimental vehicle. Future iterations on this design could focus on improving the filter robustness and detection accuracy. These results in the presence of actuator dynamics, sensor noise, and significant model uncertainty confirm that the model-based H_∞ FDI approach is viable for this application.

The anomaly detector corresponding to the data-driven approach, in contrast, has difficulty detecting slow-acting ramp faults (Fig. 6.6). The anomaly score does not increase consistently until the ramp fault is nearing its maximum value. The rapid bias fault, however, is quickly detected with a sharp increase in the anomaly score within 1 s of fault injection. Also notable is that the anomaly scores can be high prior to fault injection; the model uncertainty incorporated in the simulation causes the test data set to be somewhat different from the nominal training data used in the detection algorithm. Hence training data plays a significant role in the overall performance of data-driven techniques. This is much like the role that the ”model” plays in the model-based design.

The flight test data used in this research is critical for the development of different FDI modules for real systems. While data is always important for data-driven techniques, it is also useful for the model-based approach described herein.

6.6.2 Data-driven vs Model-based Detector Performance

Viewing the performance of the data-driven detection algorithm simply involves processing the roll angle tracking error as described in Sec. 6.5 and computing the scaled anomaly score for each data sample. This processing can be performed on the existing flight data to yield the equivalent of a real-time anomaly score computed by an onboard detector. Similarly, with the

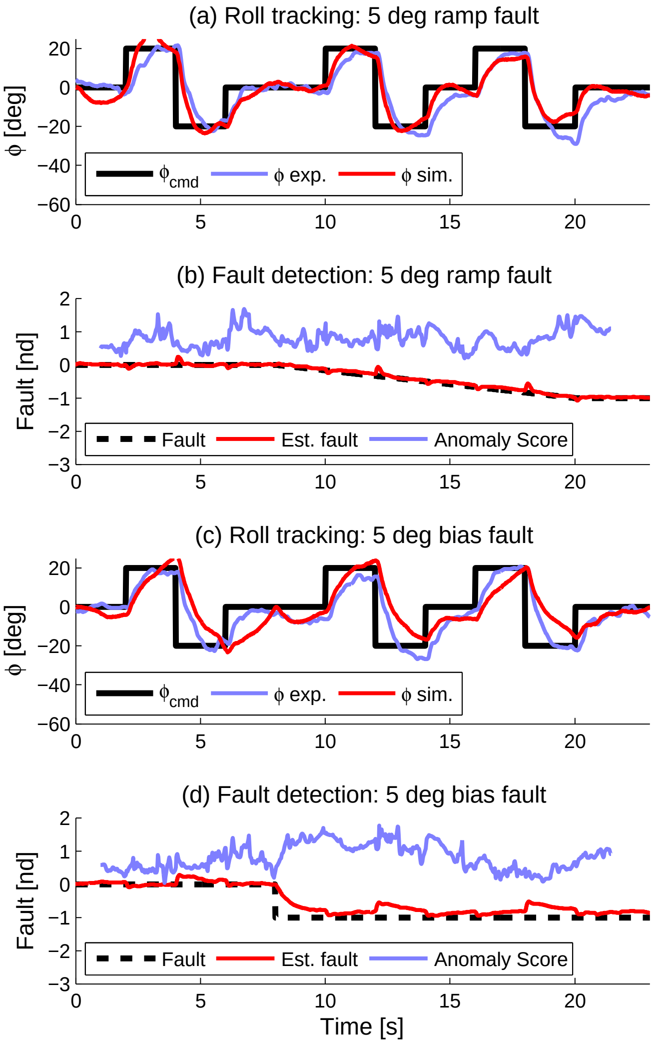


Figure 6.6: Linear UAV Simulations: Roll rate tracking and normalized fault detection performance with constant 35% input uncertainty bound

model-based FDI filter already designed, a real-time equivalent fault estimate can be computed directly from the flight data. The advantage of the data-driven detector is that arriving at this stage of the analysis is significantly faster than with the model-based approach.

Fig. 6.7 shows the anomaly scores for five different flights, each operating with a different fault scenario (including one unfaulted flight). Overlaid is the normalized fault and fault estimate from the H_∞ filter generated using only the respective flight data. Fig. 6.7(a) shows that the anomaly score is consistently below the threshold of 1 for unfaulted flight. Similarly, the FDI filter is near zero during the flight, indicating no aileron fault. The anomaly score nicely mirrors the bias faults (Fig. 6.7(d)-(e)) in the seconds following the fault injection. After approximately 10 seconds, however, the anomaly score regresses to a lesser value, reflecting that the score computation involves a trailing window of data samples for comparison to the training data set. Hence the data-driven filters will have difficulty with steady-state errors that are compensated by the flight control system. This highlights the interaction that can occur between the FDI and control system. A more detailed discussion of this interaction can be found in [33]. In contrast, the FDI filter nicely tracks the bias fault for the length of the flight test and accurately captures the fault magnitude. For ramp faults, the data-driven approach is slow to react. This may be a consequence of the lack of model knowledge embedded in the data-driven algorithm. The model-based FDI filter, however, tracks the ramp fault within a couple of seconds of its insertion and continues to track its increasing magnitude during flight. The flight test data indicates the FDI filter is able to both detect and isolate the bias and ramp faults associated with the aileron.

These qualitative impressions are supported by a simple quantitative metric that can be applied to the results shown in Fig. 6.7. The Theil Inequality Coefficient (TIC) is used as a metric to compare time histories [50]. Normalized such that 0 represents an identical match while 1 suggests a worst-case disparity, the TIC can be used to measure fault estimate performance relative to an actual fault occurring in a flight test. The TIC is defined in Eqn. 6.5 where n is the number of data samples in the data set, f is the true fault signal, and $\hat{f}$ is the fault estimate. The TIC can be applied to the model-based fault estimates to assess the relative tracking performance for each fault mode. For the data-driven approach, measures of the time to reach the simple fault threshold described in Sec. 6.5 and the persistence above that threshold can provide information about the relative ease of detection associated with each fault mode.

$$\text{TIC} = \left(\sqrt{\frac{1}{n}\sum_{i=0}^{n} (\hat{f}_i - f)^2} \right) \Bigg/ \left(\sqrt{\frac{1}{n}\sum_{i=0}^{n} \hat{f}_i^{\,2}} + \sqrt{\frac{1}{n}\sum_{i=0}^{n} f_i^{\,2}} \right) \tag{6.5}$$

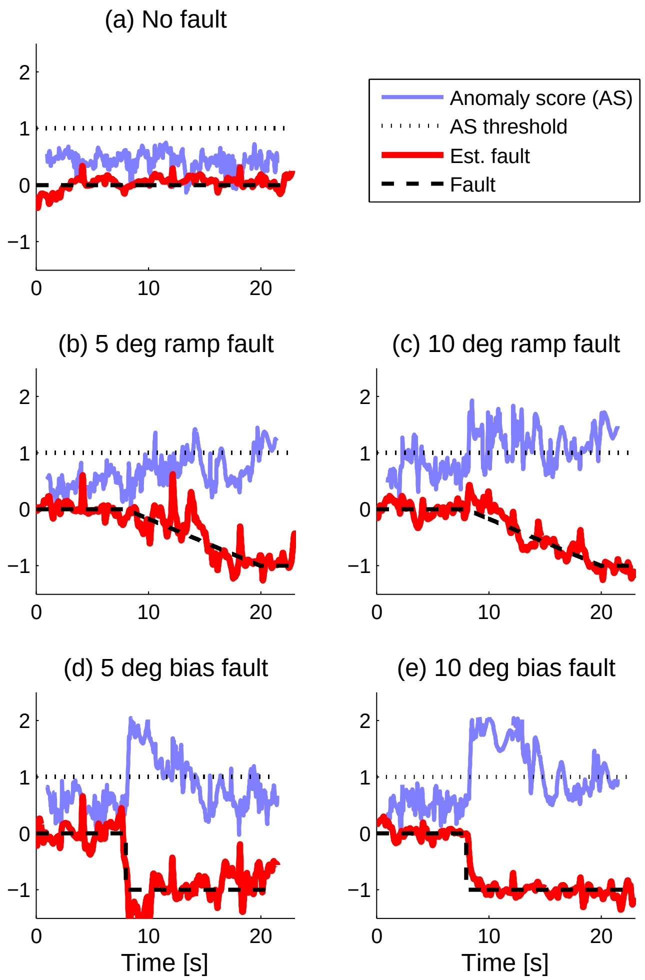


Figure 6.7: Experimental Flight Data: Anomaly scores and normalized fault estimates.

Table 6.1: Experimental FDI Performance Metrics

Fault mode	TIC (MB)	Time to Detection (DD)	Persistence (DD)
5 deg ramp	0.185	2.52 s	0.12 s
10 deg ramp	0.134	0.22 s	1.22 s
5 deg bias	0.159	0.16 s	5.64 s
10 deg bias	0.093	0.30 s	6.50 s

Table 6.1 provides the TIC for the model-based FDI filter (denoted MB) as well as time to detection and persistence metrics for the experimental data-driven results (denoted DD). The TIC confirms that bias faults were more easily tracked and that faults of a larger magnitude proved easier to detect. Since large bias faults induce a more significant, faster-acting aerodynamic force to the body of the vehicle and lead to increased control action, both the model-based and data-driven detectors recognize the anomalous behavior easily.

Revisiting the desirable attributes of a FDI system discussed in Section 6.4, both approaches show sensitivity to the fault modes tested and are mostly insensitive to the commanded doublet maneuvers. The robustness of these detectors to other faults, (e.g. sensor, different control surface) or simultaneous faults was not considered in this research; this is an opportunity for future investigation. As the linear simulation results indicate, the model-based approach demonstrates a superior robustness to anticipated model variation due to aircraft maneuvers and unmodeled dynamics. Additionally, the model-based filter was successfully designed to be insensitive to trim offset at the beginning of each flight run, implementation of a closed-loop feedback system, wind gusts and sensor noise.

The robustness of the model-based design to these factors existing in real systems demonstrates the utility of this approach. Moreover, the accurate fault estimates are valuable information for many system reconfiguration algorithms that could be employed in fault-tolerant, safety-critical systems. In contrast, the anomaly detector exhibits reduced robustness to unmodeled effects and consequently may not clearly detect the injection of a particular fault. The data-driven technique used in this research, however, enjoys the significant advantage of ease of development and may exhibit better FDI performance for unexpected faults. The model-based FDI and data-driven anomaly detection results may additionally be compared against other approaches using the experimental UAV data benchmark.

Chapter 7

Conclusion and Discussion

This dissertation detailed a conventional failure modes and effects analysis for the University of Minnesota flight research platform aircraft *Ibis*, a small low-cost UAV. The FMEA identified several critical fault modes for which the vehicle could experience loss of control or other significant failures. Design modifications were made to *Ibis* to yield a reliability-focused variant, *Baldr*.

The stuck elevator failure mode was selected for further study, and a trim state discovery analysis was performed to identify the achievable steady wings-level flight envelope using control reconfiguration. The resulting information was used to extend the prior FMEA to show more detailed failure effects and system vulnerabilities. The trim state discovery results indicated significant elevator fault vulnerability–especially for hardover and floating surface faults, two of the more common control surface impairments. The shape of the flight envelope indicated where the vehicle was likely able to fly and suggested that much of the available elevator range is not utilized to maintain steady, level flight. In addition, experimental flight-test data was superimposed on the identified flight envelope to validate these results. Candidate flight envelope protection schemes or reconfiguration algorithms could restrict the elevator deflection during certain operational modes to reduce the likelihood of a stuck surface in non-recoverable position.

The linearized model set was analyzed in the frequency domain to understand how control authority of the elevators and other effectors change throughout the flight envelope. The throttle and ailerons were both shown to provide very limited pitch control authority in the event of an elevator failure, raising questions about the viability of adaptive/reconfigured control strategies for there airframes. While these aircraft may not be a good platform for adaptive control research, this is probably common for low-cost vehicles without advanced airframe design. Dynamics suitable for adaptive control solutions may need to be ‘designed-into’ airframes in order to be effective strategies, and this advanced design may not be feasible for low-cost systems.

This system dynamics analysis can provide a rich understanding of how system capabilities and vulnerabilities change throughout an operational envelope. The FMEA conventionally assumes worst-case scenarios, but fundamental linear analysis tools can provide a FMEA feedback loop with the additional insight necessary to make appropriate design decisions. The analysis can be extended to all of the *Ibis* effectors and provides a general framework for any small UAV reliability analysis.

Finally, model-based and data-driven fault detection and isolation modules were designed for the UMN flight research platform aircraft. These modules were validated in simulation and for real flight test data, demonstrating that the approaches are complementary and can be employed together on real UAS in their operating environment. Rapid and accurate fault detection and isolation can trigger control reconfiguration algorithms, emergency path-planning strategies, and provide valuable system health data for maintenance.

This work presents many future avenues for research that will be invaluable to the growth of safe UAS technology. The extended FMEA can be applied to all phases of flight (e.g. takeoff, landing, loiter), and fault-tolerance algorithms can be more finely tuned to each operating condition. These tools can help to extract the maximum performance of fault-tolerance algorithms and mitigate the effects of reduced hardware redundancy. Coupling this enhanced reliability assessment and analytical fault detection approaches provides feedback about system fault tolerance, and it can assist in developing more formalized certification procedures for small UAS. This is a timely, necessary challenge for the aerospace industry, and there will be many opportunities to build upon this work as UAS applications continue to thrive.

Bibliography

[1] Y.C. Yeh. Triple-triple redundant 777 primary flight computer. In *Proc. 1996 IEEE Aerospace Applicat. Conf.*, pages 293–307, 1996.

[2] Y.C. Yeh. Design considerations in Boeing 777 fly-by-wire computers. In *3rd IEEE Int. High-Assurance Systems Engineering Symp.*, pages 64–72, 1998.

[3] Boeing Commercial Airplanes. Statistical Summary of Commercial Jet Airplane Accidents, Worldwide Operations, 1959-2009. Technical report, Boeing Co., July 2010.

[4] Christine M. Belcastro. Aircraft loss of control: Analysis and requirements for future safety-critical systems and their validation. In *Proc. 8th Asian Control Conference*, pages 399–406, Kaohsiung, Taiwan, May 2011.

[5] FAA Modernization and Reform Act of 2012,. H.R. 658, 112th Cong., 2nd Sess., 2012.

[6] E.M. Atkins. Risk identification and management for safe uas operation. In *Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on*, pages 774–779, 2010.

[7] Guoxing Yi and E.M. Atkins. Methods of trim state discovery. In *Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on*, pages 654–659, 2010.

[8] Thomas Lombaerts, Herve Huisman, Ping Chu, Jan A Mulder, and Diederick Joosten. Nonlinear reconfiguring flight control based on online physical model identification. *Journal of Guidance, Control, and Dynamics*, 32(3):727–748, 2009.

[9] T.J.J. Lombaerts, G.H.N. Looye, Q.P. Chu, and J.A. Mulder. Design and simulation of fault tolerant flight control based on a physical approach. *Aerospace Science and Technology*, 23(1):151 – 171, 2012. 35th ERF: Progress in Rotorcraft Research.

[10] Paul Freeman, Rohit Pandita, Nisheeth Srivastava, and Gary J. Balas. Model-based and data-driven fault detection performance for a small UAV. *IEEE Trans. Mech.*, 18(4), August 2013.

[11] Paul Freeman and Gary J. Balas. Analytical fault detection for a small UAV. In *AIAA Infotech@Aerospace Conf.*, Boston, Massachusetts, USA, August 2013.

[12] Paul Freeman and Gary J. Balas. Actuation failure modes and effects analysis for a small UAV. In *American Control Conf.*, Portland, Oregon, USA, June 2014.

[13] A. Dorobantu, W. Johnson, F.A. Lie, B. Taylor, A. Murch, Y.C. Paw, D. Gebre-Egziabher, and G. Balas. An airborne test platform: From theory to flight. In *American Control Conference*, pages 659–673, Washington, DC, USA, June 2013.

[14] A. Lie, A. Dorobantu, B. Taylor, D. Gebre-Egziabher, P. Seiler, and G. Balas. An airborne experimental test platform: From theory to flight (part 1). *Inside GNSS*, pages 10–21, March/April 2014.

[15] A. Lie, H. Mokhtarzadeh, P. Freeman, J. Larson, T. Layh, B. Hu, B. Taylor, D. Gebre-Egziabher, P. Seiler, and G. Balas. An airborne experimental test platform: From theory to flight (part 2). *Inside GNSS*, pages 40–47, May/June 2014.

[16] D. B. Owens, D. E. Cox, and E. A. Morelli. Development of a low-cost sub-scale aircraft for flight research: The FASER project. In *25th AIAA Aerodynamic Measurement Technology and Ground Testing Conference*, number 2006-3306, 2006.

[17] University of Minnesota. UAV Research Group, 2013. http://www.uav.aem.umn.edu/.

[18] M. Stamatelatos, W. Vesely, J. Dugan, J. Fragola, J. Minarick, and J. Railsback. *Fault Tree Handbook with Aerospace Applications*. NASA Office of Safety and Mission Assurance, August 2002.

[19] NASA Lewis Research Center. *Tools of Reliability Analysis – Introduction and FMEAs*, 2009.

[20] NASA Goddard Space Flight Center, Greenbelt, MD, USA. *Flight Assurance Procedure P-302-720: Performing a Failure Modes and Effects Analysis*.

[21] S. Reimann, J. Amos, E. Bergquist, J. Cole, J. Phillips, and S. Shuster. UAV for reliability build. Technical report, Department of Aerospace Engineering and Mechanics, University of Minnesota, May 2014.

[22] E.A. Morelli and R. DeLoach. Wind tunnel database development using modern experiment design and multivariate orthogonal functions. In *41st AIAA Aerospace Sciences Meeting and Exhibit*, volume 653, Reno, Nevada, USA, January 2003.

[23] G. Hoe, D.B. Owens, and C. Denham. Forced oscillation wind tunnel testing for FASER flight research aircraft. In *AIAA Atmospheric Flight Mechanics Conference*, number 2012-4645, Minneapolis, Minnesota, USA, August 2012.

[24] V. Klein and E.A. Morelli. *Aircraft system identification: theory and practice*. American Institute of Aeronautics and Astronautics Reston, VA, USA, 2006.

[25] B.L. Stevens and F.L. Lewis. *Aircraft Control and Simulaion*. John Wiley & Sons, 1992.

[26] George Zames and AK El-Sakkary. Unstable systems and feedback: The gap metric. In *Proc. Allerton Conf*, pages 380–385, 1980.

[27] Ahmed El-Sakkary. The gap metric: Robustness of stabilization of feedback systems. *Automatic Control, IEEE Transactions on*, 30(3):240–247, 1985.

[28] G. Balas, R. Chiang, A. Packard, and M. Safonov. *Robust Control Toolbox*. The MathWorks, Inc., 2010.

[29] Steven X. Ding. *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer, 1st edition, 2008.

[30] Janos J. Gertler. *Fault detection and diagnosis in engineering systems*. Marcel Dekker, 1st edition, 1998.

[31] Rolf Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer, 1st edition, 2005.

[32] D.S. Bodden, W. Hadden, B.E. Grube, and N.S. Clements. PHM as a design variable in air vehicle conceptual design. In *Aerospace Conference, 2005 IEEE*, pages 1 –11, march 2005.

[33] R. Pandita, J. Bokor, and G. Balas. Closed-loop performance metrics for fault detection and isolation filter and controller interaction. *International Journal of Robust and Nonlinear Control*, 2011.

[34] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State-space solutions to standard H_2 and H_∞ control problems. *IEEE Trans. Automat. Contr.*, 34(8):pp. 831–847, 1989.

[35] J. Stoustrup and H. Niemann. Fault estimation–a standard problem approach. *International Journal of Robust and Nonlinear Control*, 12(8), 2002.

[36] Austin M. Murch, Yew Chai Paw, Rohit Pandita, Zhefeng Li, and Gary J. Balas. A low cost small uav flight research facility. In Florian Holzapfel and Stephan Theil, editors, *Advances in Aerospace Guidance, Navigation and Control*, pages 29–40. Springer Berlin Heidelberg, 2011.

[37] Rohit Pandita. *Dynamic Flight Envelope Assessment with Flight Safety Applications*. PhD thesis, University of Minnesota-Twin Cities, 2010.

[38] Paul Freeman. Model-based and data-driven fault detection performance for a small UAV: Linear models used in research, August 2012.

[39] A. Falcoz, D. Henry, and A. Zolghadri. Robust fault diagnosis for atmospheric reentry vehicles: A case study. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 40(5):886 –899, sept. 2010.

[40] E. Balaban and J.J. Alonso. An approach to prognostic decision making in aerospace domain. In *Annual Conf. of the Prognostics and Health Management Society*, 2012.

[41] P. Freeman, P. Seiler, and G. Balas. Robust fault detection for commercial transport air data probes. In *IFAC 18th World Congr.*, pages 13723–13728, Milan, Italy, August 2011.

[42] P. Freeman. Robust fault detection for commercial transport air data probes. Master's thesis, University of Minnesota, November 2011.

[43] Paul Freeman, Peter Seiler, and Gary J. Balas. Air data system fault modeling and detection. *Control Engineering Practice*, 2013. Accepted for publication.

[44] J. Bokor, A. Edelmayer, and L. Keviczky. An H_∞ filtering approach to robust detection of failures in dynamical systems. In *Proc. 33rd IEEE Conf. on Decision and Control*, 1994.

[45] Kunpeng Sun. *Robust Linear Filter Design via LMIs and Controller Design with Actuator Saturation via SOS Programming*. PhD thesis, University of California-Berkeley, 2004.

[46] K. Sun and A.K. Packard. Robust H_2 and H_∞ filters for uncertain LFT systems. *IEEE Trans. Automat. Contr.*, 50(5):715–720, 2005.

[47] P. Gahinet and P. Apkarian. A linear matrix inequality approach to H_∞ control. *Int. J. Robust Nonlinear Control*, 1994.

[48] Nisheeth Srivastava and Jaideep Srivastava. A hybrid-logic approach towards fault detection in complex cyber-physical systems. In *Annu. Conf. Prognostics and Health Management Society*, Portland, Oregon, USA, October 2010.

[49] Aleksandar Lazarevic, Nisheeth Srivastava, Ashutosh Tiwari, Josh Isom, Nikunj Oza, and Jaideep Srivastava. Theoretically optimal distributed anomaly detection. In *IEEE Int. Conf. Data Mining Workshops*, pages 515–520, Miami, Florida, USA, December 2009.

[50] H. Theil. *Economic forecasts and policy*. Contributions to economic analysis. North-Holland Pub. Co., 1961.

Appendix A

Failure Modes and Effects Analysis Summary

This appendix includes a summary of the *Ibis* UAV failure modes and effects analysis, presented in a conventional spreadsheet format. The individual components are organized by subsystem as shown in Table 3.1.

Table A.1: Failure Modes, Effects, and Criticality Summary

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
A. AIRFRAME SUBSYSTEM							
A.1 Fuselage: Aircraft main body section. Contains payload. Positions control and stabilization surfaces in relationship to lifting surfaces.							
A.1-a	open avionics door	• unsecured door	M	• avionics pallet loss • battery loss • loss of thrust • LOM, LOC, LOV	1	H	• install locking door • affix payload to fuselage
A.1-b	structural	• in-air collision • hard landing	L	• altered aerodynamics • vibration • LOM, LOC, LOV	1	M	• pre-/post-flight inspections • fly only in safe environments
A.2 Wing: Aircraft lifting surface.							
A.2-a	structural	• in-air collision • hard landing	L	• altered aerodynamics • vibration • LOM, LOC, LOV	1	M	• pre-/post-flight inspections • fly only in safe environments
A.3 Landing Gear: Undercarriage and wheels to support vehicle takeoff and landing.							
A.3-a	structural	• in-air collision • hard landing	L	• crash during takeoff roll • inability to safely land • LOM, LOC, LOV	1	M	• pre-/post-flight inspections • fly only in safe environments

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
A.3-b	loss of wheel	• loose wheel nut	M	• crash during takeoff roll • inability to safely land • LOV	1	H	• locking wheel nuts • pre-/post-flight inspections • fly only in safe environments
B. POWERPLANT SUBSYSTEM							
B.1 Motor Batteries: Powers the aircraft motor.							
B.1-a	internal	• internal short circuit • overheating	M	• loss of thrust • vehicle becomes glider • LOM, LOC, LOV	1R	H	• pre-/post-flight inspection
B.1-b	external	• overcharging • overdischarging • mechanical damage • extreme temperatures	M	• loss of thrust • vehicle becomes glider • LOM, LOC, LOV	1R	H	• charge and store according to specifications • ensure battery is secured
B.2 Electronic Speed Controller: Sends signals for motor rotation. Varies motors speed and may act as a dynamic brake.							
B.2-a	incorrect voltage output	• internal circuitry short or damage	M	• incorrect motor speed • LOC, LOM	3	M	• create livestream functionality for ESC data • override default cutoff setting to ‘soft’ cutoff with safer setting

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
B.2-b	interruption of power to unit	• damage to battery connections	H	• intermittent loss of thrust • incorrect motor speed • LOC, LOM	2	H	• create livestream functionality for ESC data • override default cutoff setting to ’soft’ cutoff with safer setting
B.2-c	cutoff switch tripped	• damage to battery connections	H	• intermittent loss of thrust • incorrect motor speed • LOC, LOM	1	H	• create livestream functionality for ESC data • override default cutoff setting to ‘soft’ cutoff with safer setting
B.2-d	voltage output is outside thresholds for motor	• internal circuitry short or damage	M	• intermittent loss of thrust • vehicle becomes glider • LOM, LOC, LOV	1	H	• create livestream functionality for ESC data • override default cutoff setting to ’soft’ cutoff with safer setting
B.2-e	disconnection from motor battery	• damage to battery connections	H	• total loss of thrust • LOM, LOC, LOV	1	H	• securely affix connections • add redundant wiring
B.3 Motor: Turns hub, spinner, and propeller to generate thrust							
B.3-a	Bearing failure	• insufficient lubrication	M	• motor spins with difficulty or not at all • loss of thrust LOM, LOC, LOV	1	H	• scheduled maintenance • pre-/post-flight inspection

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
B.3-b	Short circuit at coil	• unbalanced propeller vibration	L	• stuck motor • loss of thrust • LOM, LOC, LOV	1	M	• scheduled maintenance • pre-/post-flight inspection
B.3-c	Loose windings	• unbalanced propeller vibration	L	• incorrect motor speed • LOC, LOM	3	L	• scheduled maintenance • pre-/post-flight inspection
B.4 Propeller hub: Connects motor driveshaft to propeller blades							
B.4-a	disconnection from motor driveshaft	• in-air collision • hard landing • loose bearings	L	• loss of propeller • loss of thrust • LOM, LOC, LOV	1	M	• pre-/post-flight inspections • fly only in safe environments • use locking bearings
B.5 Spinner: Aerodynamic fairing for propeller hub; provides more laminar airflow and less turbulence entering propeller cross-section.							
B.5-a	structural failure or cracking	• in-air collision • hard landing	L	• possible fouled airflow • slight reduction of maximum power	4	L	• pre-/post-flight inspections • fly only in safe environments

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
B.6 Propeller blades: Provide thrust							
B.6-a	structural failure or cracking	• in-air collision • hard landing	L	• moderate to severe reduction in maximum power • total loss of thrust • LOM, LOC, LOV	1	M	• pre-/post-flight inspections • fly only in safe environments
B.6-b	loss of blade from hub	• vibration due to propeller imbalance	L	• total loss of thrust • LOM, LOC, LOV	1	M	• pre-/post-flight inspections
C. EFFECTORS SUBSYSTEM							
C.1 Servo Battery: Powers avionics and effectors subsystems							
C.1-a	internal	• internal short circuit • overheating	M	• loss of power to servos • vehicle is uncontrollable • LOM, LOC, LOV	1	H	• pre-/post-flight charge retention inspection
C.1-b	external	• overcharging • overdischarging • mechanical damage • extreme temperatures	M	• loss of power to servos • vehicle is uncontrollable • LOM, LOC, LOV	1	H	• charge and store according to specifications • ensure battery is secured

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.2 Rx Mux Servo Multiplexer: uses R/C servo signals to switch servo output between manual pilot on ground or autopilot							
C.2-a	inconsistent servo signal	• noise from other wiring	L	• undesired operation mode switching between auto and manual pilot	1	M	• proper grounding • shielded wiring
C.2-b	circuit overload	• power surge • electrostatic discharge	M	• no actuation signals sent to servos, control surfaces	1	H	• proper grounding • anti-ESD handling and maintenance
C.2-c	disconnection from servo battery	• excessive vibration • severed wires	H	• no actuation signals sent to servos, control surfaces	1	H	• redundant wiring • use locking connectors
C.3 Electronic Speed Controller: Controls servo command signals							
C.3-a-e				—– *see* B.2-a-e —–			
C.4 Battery Eliminator Circuit (BEC): Splits and regulates power to receiver, failsafe switch, servos							
C.4-a	circuit overload	• power surge • electrostatic discharge	L	• loss of power to receiver, failsafe switch, servos • vehicle is uncontrollable • LOM, LOC, LOV	1	M	• proper grounding • anti-ESD handling and maintenance

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.4-b	disconnection from servo battery	• excessive vibration • severed wires	H	• loss of power to receiver, failsafe switch, servos • vehicle is uncontrollable • LOM, LOC, LOV	1	H	• redundant wiring • use locking connectors
C.4-c	loss of voltage feed to receiver	• overheating due to excessive current drawn from connected component(s)	L	• vehicle switched to autopilot • no manual control • potential LOM, LOV	1	M	• improve cooling • ventilation
C.4-d	loss of voltage feed to RxMux	• overheating due to excessive current drawn from connected component(s)	L	• control surfaces unpowered • motor unpowered • LOM, LOC, LOV	1	M	• improve cooling • ventilation
C.4-e	loss of voltage feed to servos	• overheating due to excessive current drawn from connected component(s)	M	• surfaces fixed at current positions • vehicle is uncontrollable • LOM, LOC, LOV	1	H	• improve cooling • ventilation

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C-4-f	incorrect voltage to servos	• short circuit • mechanical damage	L	• incorrect surface deflections commanded	1	M	• surge protection
C.4-g	incorrect voltage to RxMux	• short circuit • mechanical damage	L	• undesired operation mode switching between auto and manual pilot	2	M	• surge protection
C.5 Servo: Actuates control surfaces according to pilot or autopilot commands.							
C.5-a-1	elevator: bias	• poor rigging • slippage of gears • broken linkages	M	• reduced pitch control effectiveness • LOM	2	M	• inspections • redundant surface • redundant servo
C.5-a-2	elevator: stuck-at position	• broken linkage • broken servo driveshaft • unbalanced surface	M	• loss of pitch control effectiveness • dive/stall • LOM, LOC, LOV	1	H	• inspections • redundant surface • redundant servo
C.4-a-3	elevator: hardover (maximum deflection)	• broken linkage • broken servo driveshaft • unbalanced surface	L	• loss of pitch control • induced dive/stall • LOM, LOC, LOV	1	M	• inspections • redundant surface • redundant servo

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.5-a-4	elevator: floating surface	• broken linkage • broken servo driveshaft	M	• loss of pitch control • LOM, LOC, LOV	H	H	• inspections • redundant surface • redundant servo
C.5-a-5	elevator: oscillatory	• electrical damage • electrical interference	L	• uncommanded pitching	1	M	• inspections • redundant surface • redundant servo • shielding
C.5-a-6	elevator: increased deadband	• damaged gears	L	• reduced pitch control effectiveness • LOM	2	M	• inspections • redundant surface • redundant servo
C.5-a-7	elevator: increased stiction	• damaged gears	L	• slower actuator dynamics • reduced or total loss of pitch control effectiveness	2	M	• inspections • redundant surface • redundant servo
C.5-b-1	rudder: bias	• poor rigging • slippage of gears • broken linkages	M	• reduced yaw control effectiveness • LOM	2	M	• inspections • redundant surface • redundant servo

No.	**Failure Mode**	**Cause**	**Likelihood**	**Effects**	**Criticality**	**Risk**	**Controls**
C.5-b-2	rudder: stuck-at position	• broken linkage • broken servo driveshaft • unbalanced surface	M	• loss of yaw control effectiveness • LOM, LOC, LOV	1	H	• inspections • redundant surface • redundant servo
C.5-b-3	rudder: hardover (maximum deflection)	• broken linkage • broken servo driveshaft • unbalanced surface	L	• loss of yaw control • induced hard yaw • LOM, LOC, LOV	1	M	• inspections • redundant surface • redundant servo
C.5-b-4	rudder: floating surface	• broken linkage • broken servo driveshaft	M	• loss of yaw control • LOM, LOC, LOV	1	H	• inspections • redundant surface • redundant servo
C.5-b-5	rudder: oscillatory	• electrical damage • electrical interference	L	• uncommanded yawing	1	M	• inspections • redundant surface • redundant servo • shielding
C.5-b-6	rudder: increased deadband	• damaged gears	L	• reduced yaw control effectiveness • LOM	2	M	• inspections • redundant surface • redundant servo

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.5-b-7	rudder: increased stiction	• damaged gears	L	• slower actuator dynamics • reduced or total loss of yaw control effectiveness	2	M	• inspections • redundant surface • redundant servo
C.5-c-1	left/right aileron: bias	• poor rigging • slippage of gears • broken linkages	M	• reduced roll control effectiveness • LOM	2R	M	• inspections • redundant surface • redundant servo
C.5-c-2	left/right aileron: stuck-at position	• broken linkage • broken servo driveshaft • unbalanced surface	M	• loss of roll control effectiveness • LOM, LOC, LOV	1R	H	• inspections • redundant surface • redundant servo
C.5-c-3	left/right aileron: hardover (maximum deflection)	• broken linkage • broken servo driveshaft • unbalanced surface	L	• loss of roll control • induced hard yaw • LOM, LOC, LOV	1R	M	• inspections • redundant surface • redundant servo
C.5-c-4	left/right aileron: floating surface	• broken linkage • broken servo driveshaft	M	• loss of roll control • LOM, LOC, LOV	1R	H	• inspections • redundant surface • redundant servo

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.5-c-5	left/right aileron: oscillatory	• electrical damage • electrical interference	L	• uncommanded rolling	1R	M	• inspections • redundant surface • redundant servo • shielding
C.5-c-6	left/right aileron: increased deadband	• damaged gears	L	• reduced roll control effectiveness • LOM	2R	M	• inspections • redundant surface • redundant servo
C.5-c-7	left/right aileron: increased stiction	• damaged gears	L	• slower actuator dynamics • reduced or total loss of roll control effectiveness	2R	M	• inspections • redundant surface • redundant servo
C.5-d-1	left/right flap: bias	• poor rigging • slippage of gears • broken linkages	M	• altered lift/drag properties	4	L	• inspections • redundant surface • redundant servo
C.5-d-2	left/right flap: stuck-at position	• broken linkage • broken servo driveshaft • unbalanced surface	M	• altered lift/drag properties	2R	M	• inspections • redundant surface • redundant servo

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
C.5-d-3	left/right flaps: hardover (maximum deflection)	• broken linkage • broken servo driveshaft • unbalanced surface	L	• altered lift/drag properties • induced maneuvering	2R	M	• inspections • redundant surface • redundant servo
C.5-d-4	left/right flaps: floating surface	• broken linkage • broken servo driveshaft	M	• loss of flap control • altered lift/drag properties	4	L	• inspections • redundant surface • redundant servo
C.5-d-5	left/right flaps: oscillatory	• electrical damage • electrical interference	L	• uncommanded maneuvering	2R	M	• inspections • redundant surface • redundant servo • shielding
C.5-d-6	left/right flaps: increased deadband	• damaged gears	L	• reduced flap effectiveness	4	L	• inspections • redundant surface • redundant servo
C.5-c-7	left/right flaps: increased stiction	• damaged gears	L	• slower actuator dynamics • reduced or total loss of flap control	4	L	• inspections • redundant surface • redundant servo

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.1 Flight Computer: Provides guidance, navigation, control, fault detection algorithm implementation. Provides data acquisition.							
D.1-a	Disconnection from BEC	• vibrations • damaged wiring	M	• loss of data and automatic control • manual control only • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring
D.1-b	Disconnection from IMU	• vibrations • damaged wiring	M	• loss of IMU data • incorrect effector commands applied • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring
D.1-c	Disconnection from ADC	• vibrations • damaged wiring	M	• loss of surface position measurements • incorrect effector commands could be applied • LOM	2	M	• locking connectors • redundant wiring

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.1-d	Disconnection from pressure transducer	• vibrations • damaged wiring	M	• loss of pressure data, airspeed, altitude measurements • vehicle may fly erratically • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring
D.2 Interface Board: Power and signal interface between flight computer, avionics, effectors, powerplant							
D.2-a	Internal short circuit	• manufacturing defect	M	• Loss of communication between computer and other components • LOM, LOC, LOV	1	H	• thorough testing and inspections
D.3 Modem: Broadcasts telemetry data to ground station							
D.3-a	Disconnection from battery	• vibrations • damaged wiring	M	• no flight data received • no telemetry data sent to ground station • LOM	3	M	• locking connectors • redundant wiring

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.3-b	Disconnection from BEC	• vibrations • damaged wiring	M	• no flight data received • no telemetry data sent to ground station • LOM	3	M	• locking connectors • redundant wiring
D.4 Radio Receiver: Receives commands from ground pilot radio controller							
D.4-a	Antenna fade	• signal noise in flight test environment	L	• partial or total loss of manual control • possible LOM, LOC, LOV	1	M	• N/A
D.4-b	Disconnection from BEC	• vibrations •damaged wiring	M	•loss of manual control •autopilot only •possible LOM, LOC, LOV	1	H	• locking connectors • redundant wiring
D.5 PWM Reader: Transmits received radio commands to interface board and flight computer							
D.5-a	loss of pilot control	• vibrations • damaged wiring	L	• loss of manual control • LOM, LOC, LOV	1	M	• emergency landing

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.5-b	Loss of autopilot control	• vibrations • damaged wiring	L	• loss of autopilot control • LOM, LOC, LOV	1	M	• manual emergency landing
D.5-c	Loss of servo control	• vibrations • damaged wiring	L	• loss of surface control • LOM, LOC, LOV	1	M	• emergency landing
D.5-d	Loss of speed control	• vibrations • damaged wiring	L	• loss of speed control • LOM, LOC, LOV	1	M	• emergency landing
D.6 IMU: Provides velocity, orientation, and gravitational force measurements							
D.6-a	Circuitry overload and failure	• power surge • electrostatic discharge	L	• loss of attitude, rate data • autopilot ineffective; manual control only • LOM, LOC, LOV	1	M	• manual emergency landing • surge protection, grounding
D.6-b	Decalibration	• reverts to factory setting • physical damage	M	• erroneous attitude, rate data, flight data • autopilot may have reduced efficacy • LOM	2	M	• inspections • recalibration

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.6-c	Disconnection from BEC	• vibrations • damaged wiring	M	• loss of attitude, rate data • autopilot ineffective; manual control only • LOM, LOC, LOV	1	M	• locking connectors • redundant wiring
D.7 GPS: Provides location, velocity, altitude, time data							
D.7-a	Circuitry overload and failure	• power surge • electrostatic discharge	M	• navigation states lost • autopilot ineffective • manual control only • LOM, LOC, LOV	1	H	• manual emergency landing • surge protection, grounding
D.7-b	Disconnection from antenna	• vibrations • damaged wiring	M	• Navigation states lost • autopilot ineffective; manual control only • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.7-c	Disconnection from BEC	• vibrations • damaged wiring	M	• Navigation states lost • autopilot ineffective; manual control only • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring
D.7-d	Antenna fade	• signal noise in flight test environment	M	• Navigation states lost intermittently • autopilot ineffective; manual control only • LOM, LOC, LOV	1	H	• N/A
D.8 Air Data Probe: Provides static and total pressure measurements (does it also do AOA/AOS?)							
D.8-a	Partial or full pitot port blockage	• debris • water/condensation	H	• incorrect airspeed measurement • uncommanded maneuvering • LOM, LOC, LOV	1	H	• keep probe covered when not in use • clean probe before use with pressurized air • fly in appropriate environments

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.8-b	Static port blockage	• debris • water/condensation	H	• incorrect altitude or airspeed measurement • uncommanded maneuvering • LOM, LOC, LOV	1	H	• keep probe covered when not in use • clean probe before use with pressurized air fly in appropriate environments
D.8-c	Misalignment	• hard landing, bumps	M	• Incorrect measurements	2	M	• inspections
D.9 Pressure Transducer: Converts pneumatic pressure signal to digital format for data acquisition							
D.9-a	Disconnection from BEC	• vibrations • damaged wiring	M	• Loss of pressure measurements • Loss of primary altitude, airspeed measurements • Uncommanded maneuvering • LOM, LOC, LOV	1	H	• locking connectors • redundant wiring

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.10 AOA/AOS Vane: Measures vehicle angle of attack and angle of sideslip							
D.10-a	Stuck/sticky vane	• insufficient lubrication	L	• LOM if collecting AOA/AOS deflection data; otherwise the criticality is minor and risk is low	3	L	• inspections • lubrication
D.11 Rotary Potentiometers: Measure deflection of AOA/AOS vanes and control surfaces							
D.11-a	Total failure	• short-circuit due to wiper and track wear	L	• exhibits high noise as wiper/track wear and approach failure • LOM if collecting surface, AOA/AOS deflection data; otherwise the criticality is minor and risk is low	3	L	• inspections

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.11-b	Incorrect measurement	• loose strap	M	• LOM if collecting surface, AOA/AOS deflection data; otherwise the criticality is minor and risk is low	3	M	• inspections
D.12 ADC: Convert rotary potentiometer signals to digital format for data acquisition							
D.12-a	Loss of ADC chip functionality	• short-circuit	L	• Exhibits high noise as wiper/track wear and approach failure • LOM if collecting surface, AOA/AOS deflection data; otherwise the criticality is minor and risk is low	3	L	• inspections

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.13 Camera Battery: Powers camera							
D.13-a	*see* B.1		L	• LOM if using camera to film • LOM, LOC, LOV if using vision-based GNC • otherwise the criticality is minor and risk is low	3	🟩 L	*see* B.1
D.14 Camera Voltage Regulator: Regulates voltage for camera system							
D.14-a	*see* B.2-a-e		L	• LOM if using camera to film • LOM, LOC, LOV if using vision-based GNC • otherwise the criticality is minor and risk is low	3	🟩 L	*see* B.2-a-e

No.	Failure Mode	Cause	Likelihood	Effects	Criticality	Risk	Controls
D.15 Camera: Provides image data for guidance, navigation, control research							
D.15-a	Disconnection from power supply	• vibrations • damaged wiring	L	• LOM if using camera to film • LOM, LOC, LOV if using vision-based GNC • otherwise the criticality is minor and risk is low	2	M	• locking connectors • redundant wiring