

Control allocation as part of a fault-tolerant control
architecture for UAVs

by

Lionel Basson

*Thesis presented in partial fulfilment of the requirements for the
degree
Master of Science in Engineering
at Stellenbosch University*



Supervisor:

Dr I.K. Peddle

Department Electrical and Electronic Engineering

March 2011

Declaration

By submitting this thesis/dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

March 2011

Abstract

The development of a control allocation system for use as part of a fault-tolerant control (FTC) system in unmanned aerial vehicles (UAVs) is presented. This system plays a vital role in minimising the possibility that a fault will necessitate the reconfiguration of the control, guidance or navigation systems of the aircraft by minimising the difference between the desired and achievable aircraft performance parameters. This is achieved by optimising the allocation of control effort commanded by the virtual actuators to the physical actuators present on the aircraft.

A simple general six degree of freedom aircraft model is presented that contains all of the relevant terms needed to find the trim biases of the aircraft actuators and evaluate the performance of the virtual actuators. This model was used to develop a control allocation formulation that optimises the performance of the virtual actuators of the aircraft while minimising adverse effects and avoiding actuator saturation. The resulting problem formulation was formulated as a multi-objective optimisation problem which was solved using the sequential quadratic programming method.

The control allocation system was practically implemented and tested. A number of failure categories of varying severity were defined and two aircraft with different levels of actuator redundancy were used to test the system. The control allocation algorithm was evaluated for each failure category, aircraft test case and for a number of differing control allocation system configurations. A number of enhancements were then made to the control allocation system which included adding frequency-based allocation and adapting the algorithm for an unconventional ducted-fan UAV.

The control allocation system is shown to be applicable to a number of different conventional aircraft configurations with no alterations as well as being applicable to unconventional aircraft with minor alterations. The control allocation system is shown to be capable of handling both single and multiple actuator failures and the importance of actuator redundancy is highlighted as a factor that influences the effectiveness of control allocation. The control allocation system can be effectively used as part of a FTC system or as a tool that can be used to investigate control allocation and aircraft redundancy.

Opsomming

Die ontwikkeling van 'n beheertoekenning sisteem vir gebruik as deel van 'n fout verdraagsame beheersisteem in onbemande lugvaartuie word voorgelê. Hierdie sisteem speel 'n essensiële rol in die vermindering van die moontlikheid dat 'n fout die herkonfigurasie van die beheer, bestuur of navigasiesisteme van die vaartuig tot gevolg sal hê, deur die verskil te verminder tussen die verlangde en bereikbare werkverrigtingsraamwerk van die vaartuig. Dit word bereik deur die optimisering van die toekenning van beheerpoging aangevoer deur die virtuele aktueerders na die fisiese aktueerders teenwoordig op die vaartuig.

'n Eenvoudige algemene ses grade van vryheid lugvaartuig model word voorgestel wat al die relevante terme bevat wat benodig word om die onewewigtigheid verstelling van die vaartuig se aktueerders te vind en die werksverrigting van die virtuele aktueerders te evalueer. Hierdie model is gebruik om 'n beheer toekenning formulering te ontwikkel wat die werkverrigting van die virtuele aktueerders van die vaartuig optimiseer terwyl nadelige gevolge verminder word asook aktueerder versadiging vermy word. Die gevolglike probleem formulering is omskryf as 'n multi-doel optimiserings probleem wat opgelos is deur gebruik van die sekweniële kwadratiese programmerings metode.

Die beheertoekenning sisteem is prakties geïmplementeer en getoets. 'n Aantal fout kategorieë van verskillende grade van erns is gedefinieer en twee vaartuie met verskillende vlakke van aktueerder oortolligheid is gebruik om die sisteem te toets. Die beheer toekenning algoritme is geëvalueer vir elke fout kategorie, vaartuig toetsgeval, asook vir 'n aantal verskillende beheertoekenning sisteem konfigurasies. 'n Aantal verbeterings is aangebring aan die beheertoekenning sisteem, naamlik die toevoeging van frekwensie gebaseerde toekenning en wysiging van die algoritme vir 'n onkonvensionele onbemande geleide waaier lugvaartuig.

Die beheertoekenning sisteem is van toepassing op 'n aantal verskillende konvensionele vaartuig konfigurasies met geen verstellings asook van toepassing op onkonvensionele vaartuie met geringe verstellings. Die beheertoekenning sisteem kan beide enkel- en veelvoudige aktueerder tekortkominge hanteer en die belangrikheid van aktueerder oor-

tolligheid is beklemtoon as 'n faktor wat die effektiwiteit van beheertoekenning beïnvloed. Die beheertoekenning sisteem kan effektief geïmplementeer word as deel van 'n fout verdraagsame beheersisteem of as 'n werktuig om beheertoekenning en vaartuig oortolligheid te ondersoek.

Contents

Abstract	iii
Opsomming	iv
Acknowledgements	xxi
1 Introduction	1
1.1 The control allocation problem	4
1.1.1 Typical goals of control allocation	5
1.1.2 The general problem formulation	6
1.1.3 Typical solution methods	8
1.2 Objectives of research	13
1.2.1 Control allocation system	13
1.2.2 Evaluation of effectiveness	13
1.3 Overview of the work done	14
1.4 Thesis overview	17
2 Aircraft model	19
2.1 General Aircraft model	19
2.1.1 Axis System definitions	19
2.1.2 Notation	21
2.1.3 Euler angles and axes transformations	22
2.1.4 Forces and Moments	23
2.1.5 Expanded forces and moments	31
2.2 Conclusion	32

3	Problem formulation	33
3.1	Introduction	33
3.2	Virtual actuators and control mixing vectors	34
3.2.1	Virtual Actuators	34
3.2.2	Control mixing and biasing vectors	35
3.3	The Control allocation system	36
3.4	The full 6-DOF problem formulation	37
3.4.1	Trimming	37
3.4.2	Minimising force and moment errors	44
3.4.3	Minimising adverse effects	45
3.4.4	Actuator dynamics	47
3.4.5	Control usage	48
3.4.6	Constraints	48
3.5	Summary of objectives and constraints	50
3.5.1	Objectives	50
3.5.2	Constraints	51
3.6	Conclusion	53
4	Optimisation Detail	54
4.1	Introduction	54
4.2	Multi-objective Optimisation	55
4.2.1	General problem formulation	55
4.2.2	Considerations when solving	56
4.2.3	The choice of a cost function	57
4.2.4	Normalisation	59
4.3	Selecting an optimisation method	60
4.4	The SQP algorithm	62
4.5	Optimisation problem setup	64
4.5.1	Design vector	64
4.5.2	Setting up the cost function and finding its derivatives	65

4.5.3	Setting up the constraints and finding their derivatives	68
4.5.4	Final optimisation problem formulation	69
4.6	SQP algorithm practical implementation	69
4.7	Conclusion	70
5	Base system implementation	71
5.1	Overview	71
5.2	System inputs	72
5.2.1	Offline Inputs	72
5.2.2	Online inputs: Aircraft model data	74
5.3	Cost function and constraints	75
5.3.1	Cost and constraints	75
5.3.2	Weighting and normalisation selection	76
5.4	Optimisation layer	78
5.5	System Outputs	80
5.5.1	Control mixing information	81
5.5.2	Performance information	81
5.5.3	Optimisation histories and other useful data	81
5.6	Conclusion	81
6	Base system testing and results	83
6.1	Testing procedure	83
6.1.1	Control allocation system configurations	83
6.1.2	Tested aircraft	84
6.1.3	Failure categories	85
6.2	Additional notes on the testing procedure followed	86
6.3	Testing	86
6.3.1	Validation of optimisation and problem parameters	87
6.3.2	Single actuator soft failures	91
6.3.3	Single actuator extreme failures	105
6.3.4	Multiple actuator soft failures	108

6.3.5	Multiple actuator extreme failures	117
6.3.6	Other failure cases	124
6.3.7	Control allocation system configuration 3	126
6.3.8	Conclusion	127
7	Control allocation system expansion	129
7.1	Proposed additions	129
7.1.1	Actuator dynamics: Frequency-based allocation	129
7.1.2	Unconventional UAV: A ducted-fan UAV	132
7.1.3	Splitting the optimisation	133
7.2	Implementation	135
7.2.1	Method of extension	135
7.3	Testing	136
7.3.1	Combination SLADe test and frequency allocation test	136
7.3.2	Splitting the optimisation	141
7.4	Conclusion	144
8	Conclusion and recommendations	145
8.1	Conclusion	145
8.2	Recommendations	146
8.2.1	Detailed recommendations	146
8.2.2	Further recommendations	149
A	Aircraft configurations and parameters	151
A.1	Modular UAV	151
A.1.1	Engine Specifications	152
A.1.2	Physical Specifications	152
A.1.3	Aerodynamic Specifications	152
A.2	Variable stability UAV	154
A.2.1	Engine Specifications	154
A.2.2	Physical Specifications	154

A.2.3	Aerodynamic Specifications	154
A.3	SLADE Ducted fan UAV	156
A.3.1	Engine Specifications	156
B	Aircraft test cases	158
B.1	Modular UAV	158
B.1.1	Failure category 1: Single-actuator failures	158
B.1.2	Failure category 2: Single-actuator failures at extreme positions . .	159
B.1.3	Failure category 3: Multiple-actuator failures	159
B.1.4	Failure category 4: Multiple-actuator failures at extreme positions .	161
B.2	VSA UAV	163
B.2.1	Failure category 1: Single-actuator failures	163
B.2.2	Failure category 2: Single-actuator failures at extreme positions . .	164
B.2.3	Failure category 3: Multiple-actuator failures	164
B.2.4	Failure category 4: Multiple-actuator failures at extreme positions .	165
C	Adverse force minimisation graphs	167
C.1	Modular UAV	167
C.1.1	Failure category 2	167
C.1.2	Failure category 3	168
C.1.3	Failure category 4	168
C.2	VSA UAV	169
C.2.1	Failure category 2	169
C.2.2	Failure category 3	169
C.2.3	Failure category 4	170
D	Individual case error plots	171
	List of References	176

List of Figures

- 1.1 Modular UAV fault tolerant test-bed 2
- 1.2 An overview of the FTC system [1] 3
- 1.3 Control Allocation in a typical system 5
- 1.4 Daisy-chain control allocation [2] 10
- 1.5 Semi-static control allocation architecture 15

- 2.1 The inertial reference frame [3] 20
- 2.2 The body axis system [4] 21
- 2.3 The wind axis system [4] 22
- 2.4 The Euler 3-2-1 rotation sequence [3] 23
- 2.5 General aircraft model block diagram [5] 24
- 2.6 Engine position and angle offsets from body axes 27

- 3.1 The control allocation system architecture 36
- 3.2 Asymmetric flight configuration [6] 41
- 3.3 Asymmetric flight with zero bank angle [6] 42
- 3.4 Asymmetric flight with zero side-slip angle [6] 43
- 3.5 Asymmetric flight with a combination of bank angle and side-slip angle [6] 43

- 4.1 Optimisation problem setup overview 54

- 5.1 Control allocation implementation overview 72

- 6.1 Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 1 showing the benefit of taking into account adverse forces for the Modular UAV 94

6.2	Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 1 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV	95
6.3	A plot of the largest primary moment errors for each failure case for failure category 1 (Modular UAV)	95
6.4	A plot of the largest primary moment errors for each failure case for failure category 1 (VSA UAV)	101
6.5	A plot of the largest primary moment errors for each failure case for failure category 2 (Modular UAV)	107
6.6	A plot of the largest primary moment errors for each failure case for failure category 2 (VSA UAV)	108
7.1	Frequency-based allocation with complimentary filters	130
7.2	The SLADe ducted-fan UAV	133
7.3	Trim bias and moment optimisation link	134
7.4	Expanded system input layer	135
A.1	The Meraka Modular UAV	152
A.2	The SLADe ducted-fan UAV technical diagram	156
C.1	Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 2 showing the benefit of taking into account adverse forces for the Modular UAV	167
C.2	Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 3 showing the benefit of taking into account adverse forces for the Modular UAV	168
C.3	Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 4 showing the benefit of taking into account adverse forces for the Modular UAV	168

- C.4 Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 2 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV 169
- C.5 Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 3 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV 169
- C.6 Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 4 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV 170
- D.1 A plot of the largest primary moment errors for each failure case for failure category 3 (Modular UAV) 172
- D.2 A plot of the largest primary moment errors for each failure case for failure category 4 (Modular UAV) 173
- D.3 A plot of the largest primary moment errors for each failure case for failure category 3 (VSA UAV) 174
- D.4 A plot of the largest primary moment errors for each failure case for failure category 4 (VSA UAV) 175

List of Tables

- 4.1 Non-linear Direct and Indirect optimisation methods 61
- 5.1 User input requirements 73
- 5.2 The weights that prioritise the four main goals of the optimisation 77
- 5.3 SQP optimisation algorithm parameters 79
- 5.4 Optimisation algorithm parameter values 80
- 6.1 Trim angles for the Modular UAV test aircraft for the nominal case 88
- 6.2 Comparison between nominal and CA actuator biases for the Modular UAV 88
- 6.3 Comparison between nominal and CA selected mixing vectors for the Modular UAV 90
- 6.4 Force and moment errors for the CA results 91
- 6.5 Control allocation performance overview for failure category 1 93
- 6.6 Trim angles for failure case 31 (Modular UAV) 96
- 6.7 CA-selected actuator biases for failure case 31 (Modular UAV) 97
- 6.8 CA-selected mixing vectors for failure case 31 (Modular UAV) 97
- 6.9 Force and moment errors for failure case 31 (Modular UAV) 98
- 6.10 Trim angles for failure case 46 (Modular UAV) 99
- 6.11 CA-selected actuator biases for failure case 46 (Modular UAV) 99
- 6.12 CA-selected mixing vectors for failure case 46 (Modular UAV) 99
- 6.13 Trim angles for failure case 2 (VSA UAV) 101
- 6.14 CA-selected actuator biases for failure case 2 (VSA UAV) 102
- 6.15 CA-selected mixing vectors for failure case 2 (VSA UAV) 102
- 6.16 Force and moment errors for failure case 2 (VSA UAV) 102

6.17 Force and moment errors for failure case 2, system configuration 1 (VSA UAV)	103
6.18 Trim angles for failure case 21 (VSA UAV)	104
6.19 CA-selected actuator biases for failure case 21 (VSA UAV)	104
6.20 CA-selected mixing vectors for failure case 21 (VSA UAV)	104
6.21 Force and moment errors for failure case 21 (VSA UAV)	104
6.22 Control allocation performance overview for failure category 2	106
6.23 Control allocation performance overview for failure category 3	110
6.24 Trim angles for failure case 1 (Modular UAV)	111
6.25 CA-selected actuator biases for failure case 1 (Modular UAV)	112
6.26 CA-selected mixing vectors for failure case 1 (Modular UAV)	112
6.27 Force and moment errors for failure case 1 (Modular UAV)	112
6.28 Trim angles for failure case 190 (Modular UAV)	113
6.29 CA-selected actuator biases for failure case 190 (Modular UAV)	114
6.30 CA-selected mixing gains for failure case 190 (Modular UAV)	114
6.31 Force and moment errors for failure case 190 (Modular UAV)	114
6.32 Trim angles for failure case 57 (VSA UAV)	116
6.33 CA-selected actuator biases for failure case 57 (VSA UAV)	116
6.34 CA-selected mixing vectors for failure case 57 (VSA UAV)	116
6.35 Force and moment errors for failure case 57 (VSA UAV)	116
6.36 Control allocation performance overview for failure category 4	118
6.37 Trim angles for failure case 452 (Modular UAV)	120
6.38 CA-selected actuator biases for failure case 452 (Modular UAV)	120
6.39 CA-selected mixing vectors for failure case 452 (Modular UAV)	121
6.40 Force and moment errors for failure case 452 (Modular UAV)	121
6.41 Trim angles for failure case 166 (VSA UAV)	122
6.42 CA-selected actuator biases for failure case 57 (VSA UAV)	123
6.43 CA-selected mixing vectors for failure case 57 (VSA UAV)	123
6.44 Force and moment errors for failure case 57 (VSA UAV)	123
6.45 Trim angles for the Modular UAV test aircraft for $\lambda = 0.5$	124

6.46	CA-selected biasing vectors for a left engine failure (Modular UAV)	125
6.47	CA-selected mixing vectors for a left engine failure (Modular UAV)	125
6.48	Trim angles for the Modular UAV test aircraft for $\lambda = 1$	126
6.49	Trim angles for the Modular UAV test aircraft for $\lambda = 0$	126
7.1	Trim angles for the SLADe test aircraft	137
7.2	Comparison between nominal and CA actuator biases for the SLADe UAV	137
7.3	Comparison between nominal and CA selected mixing vectors for the SLADe UAV	138
7.4	Trim angles for the SLADe test aircraft	139
7.5	Comparison between nominal and CA actuator biases for the SLADe UAV	139
7.6	Comparison between nominal and CA selected mixing vectors for the SLADe UAV	140
7.7	Fast virtual yaw actuator (Frequency allocation)	141
7.8	Slow virtual yaw actuator (Frequency allocation)	141
7.9	Trim angles for the Modular UAV test aircraft	142
7.10	Comparison between nominal and CA actuator biases for the Modular UAV	142
7.11	Trim angles for the SLADe UAV test aircraft	142
7.12	Comparison between nominal and CA actuator biases for the SLADe UAV	143
A.1	Modular UAV Engine Parameters	152
A.2	Modular UAV Physical Parameters	153
A.3	Modular UAV Stability Derivatives	153
A.4	Modular UAV Control Derivatives	153
A.5	VSA UAV Engine Parameters	154
A.6	VSA UAV Physical Parameters	155
A.7	VSA UAV Stability Derivatives	155
A.8	VSA UAV Control Derivatives	155
A.9	VSA UAV Engine Parameters	156

Nomenclature

Abbreviations & Acronyms

AMS	Attainable moment space
DCM	Direction Cosine Matrix
DOF	Degrees of freedom
ESL	Electronic Systems Laboratory
FTC	Fault tolerant control
I.D.	Identification
SQP	Sequential Quadratic Programming
UAV	Unmanned Aerial Vehicle
LSA	Lateral specific acceleration

Greek Letters

α	Angle of attack
β	Side-slip angle
δ	Command vector to physical actuators.
δ_A	Virtual roll command
δ_E	Virtual pitch command
δ_R	Virtual yaw command
δ_T	Virtual thrust command
ϕ_e	Engine pitch offset angle from the aircraft body axes

ψ_e	Engine yaw offset angle from the aircraft body axes
ϕ	Bank angle
ψ	Yaw angle
θ	Pitch angle

Lowercase Letters

e	Oswald efficiency factor
x_e	Engine position offset from X_B -axes
y_e	Engine position offset from Y_B -axes
z_e	Engine position offset from Z_B -axes
\bar{c}	Mean aerodynamic chord
b	Wingspan
q	Dynamic pressure

Subscripts

A	Pertaining to a virtual roll command
E	Pertaining to a virtual pitch command
I	Element is in inertial axes
R	Pertaining to a virtual yaw command
T	Pertaining to a virtual thrust command
B	Element is in body axes
W	Element is in wind axes

Superscripts

A	Aerodynamic
G	Gravitational
T	Thrust

Uppercase Letters

$C_{L\delta}$	Lift-force effectiveness vector for aerodynamic actuators.
$C_{l\delta}$	Roll moment effectiveness vector for aerodynamic actuators.
$C_{lr\delta_t}$	Vector of roll moments due to rotor effects
$C_{m\delta}$	Pitch moment effectiveness vector for aerodynamic actuators.
$C_{mr\delta_t}$	Vector of pitch moments due to rotor effects
$C_{n\delta}$	Yaw moment effectiveness vector for aerodynamic actuators.
$C_{nr\delta_t}$	Vector of yaw moments due to rotor effects
$C_{x\delta_t}$	Engine effectiveness vector in X_B -axes
$C_{y\delta}$	Side-force effectiveness vector for aerodynamic actuators.
$C_{y\delta_t}$	Engine effectiveness vector in Y_B -axes
$C_{z\delta_t}$	Engine effectiveness vector in Z_B -axes
C	Effectiveness matrix
T_{AA}	Virtual roll actuator mixing vector (Aerodynamic)
T_{AB}	Actuator biasing vector (Aerodynamic)
T_{AE}	Virtual pitch actuator mixing vector (Aerodynamic)
T_{AR}	Virtual yaw actuator mixing vector (Aerodynamic)
T_{AT}	Virtual thrust actuator mixing vector (Aerodynamic)
T_{TA}	Virtual roll actuator mixing vector (Thrust)
T_{TB}	Actuator biasing vector (Thrust)
T_{TE}	Virtual pitch actuator mixing vector (Thrust)
T_{TR}	Virtual yaw actuator mixing vector (Thrust)
T_{TT}	Virtual thrust actuator mixing vector (Thrust)
V	Moment vector commanded by the control system.
A	Aspect ratio of the wing
F_{δ_t}	Force due to thrust in engine axes

L	Rolling moment
L_{δ_t}	Roll moment due to rotor effects in engine axes
M	Pitching moment
N	Yawing moment
C_l	Aerodynamic roll moment coefficient
C_m	Aerodynamic pitch moment coefficient
C_n	Aerodynamic yaw moment coefficient
C_x	Aerodynamic force coefficient for the X -axis
C_y	Aerodynamic force coefficient for the Y -axis
C_z	Aerodynamic force coefficient for the Z -axis
S	Wing reference area
V	Airspeed
J	Cost function

Acknowledgements

This project would not have been possible without the help and support of many people. I would like to acknowledge the following people:

- The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF.
- Dr. Iain K. Peddle for sharing his knowledge and guiding me in my work.
- Chris Jaquet for his help and advice on many topics.
- My family. My mom and dad for providing me with the opportunity to study this far and for their love and support. My brothers, Nicholas and Riaan, for helping me to relax on weekends and for their support.
- Gannie and Wicus, for reasons I can never express.
- Karla and Gerrit for too many reasons to list here.
- My friends in the lab for keeping things interesting, the many distractions and letting me work when I needed to.

Chapter 1

Introduction

As the use of unmanned aerial vehicles (UAVs) becomes more popular in both military and civil applications, more stringent performance and safety requirements become necessary. These requirements are necessary in order to reduce the costs associated with lost UAVs and to ensure that the number of operational UAVs can be increased without compromising safety. Conventional feedback control designs may not be capable of providing the necessary performance if an aircraft is damaged and may even result in instability [7]. To address the issues of reliability and performance in the presence of failures, control systems which are capable of automatically tolerating failures are required and these types of control systems are known as fault-tolerant control (FTC) systems [7]. FTC systems will aid in achieving the following industry goals:

- To increase safety and reliability of UAVs, thus facilitating the convergence of manned and unmanned aircraft into the same airspace [1].
- To increase the usefulness of UAVs in harsh environments where there is a high probability of damage to the aircraft [1].

FTC systems research is a broad area of research with many classifications of FTC methods and control structures, as discussed by [7]. [8] also gives a brief overview of the current methods and mentions a number of successful implementations of FTC systems that have been achieved, such as:

- The NASA F15 Self Repairing Flight Control System
- Defense Advanced Research Projects Agency (DARPA) tests on the Boeing T-33 and X-45 unmanned combat aerial vehicles

A brief overview is given of the state of FTC systems research at Stellenbosch University.

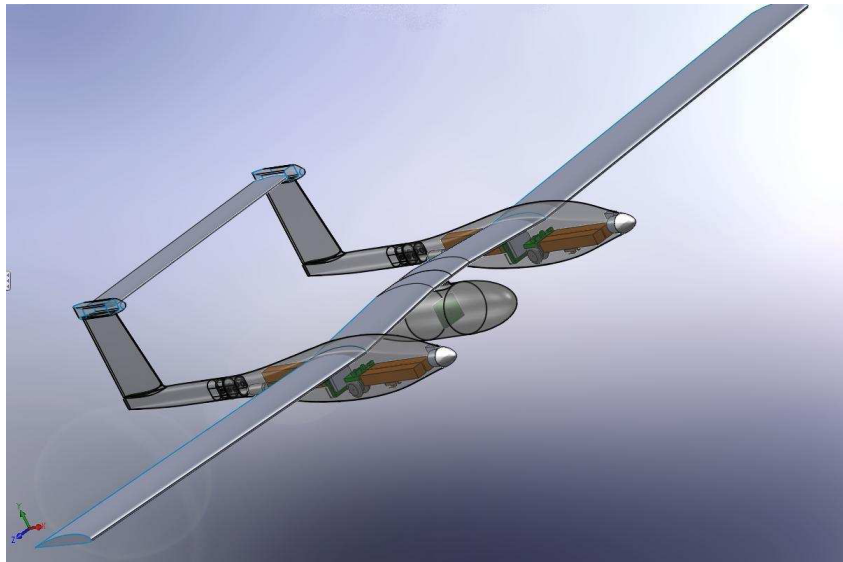


Figure 1.1: Modular UAV fault tolerant test-bed

FTC systems research at Stellenbosch University

The fault tolerant control group within the ESL was started with the focus on developing tools and architectures for a FTC system that is capable of rescuing a UAV from a number of undesirable flight conditions and component failures. The tools and architectures are being developed in such a way as to be applicable to a wide variety of UAVs, not a specific design, although the probable test-bed for these systems is the modular UAV shown in figure 1.1.

There is research being done at the FTC architecture level with a number of supporting projects, some of which are: adaptive control, stall prevention and recovery, system identification (I.D.) and control allocation. Figure 1.2 gives an overview of the FTC system being developed.

The FTC system is controlled by a supervisor system that decides whether reconfiguration is necessary after a failure has occurred. A fault-detection system will be present on-board to detect faults and the supervisor will initiate the recovery sequence. As can be seen in figure 1.2, the system is capable of reconfiguring the inner-loop controllers, guidance controller and navigation controller. Whether or not these systems need to be reconfigured will depend on how the virtual aircraft changes. The control allocation algorithm fits into this system in the virtual aircraft layer, where the virtual control commands are converted to physical actuator commands. The focus of this thesis is on investigating and developing the control allocation system that will be integrated into the FTC architecture being developed.

The presence of a control allocation system is beneficial to the FTC system since the

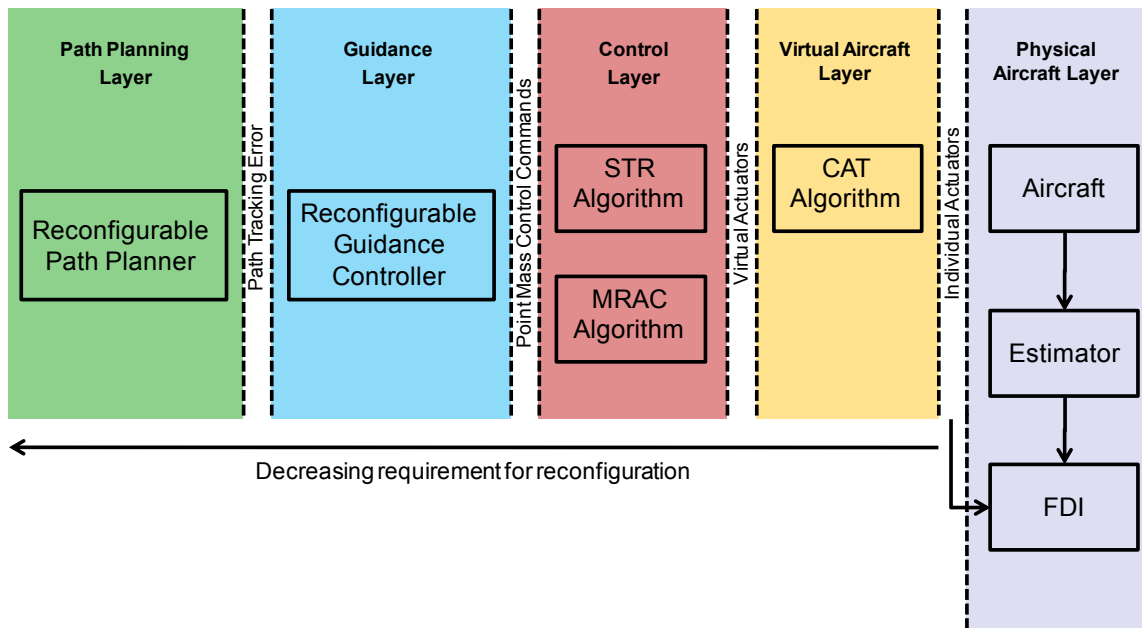


Figure 1.2: An overview of the FTC system [1]

control system for a given aircraft is designed with specific performance parameters in mind and any change in these performance parameters will degrade the performance of the control system. In the event of an actuator failure in the aircraft, the achievable forces and moments using the default actuator sets will be altered. In order for the aircraft to maintain acceptable performance, one of two things must happen: either the performance of the virtual actuators must be restored, or the control system must be reconfigured. The control allocation system is the first sub-system to be reconfigured in the event of a failure and it will attempt to restore the performance of the virtual actuators. Whether or not reconfiguration of the higher-level control systems is necessary will depend on the success of the control allocation system in restoring the virtual actuators. In this application, the job of the control allocation system is therefore the following:

- To provide the control system with actuator sets that perform within a certain range of the desired performance values if possible, thus negating the need to reconfigure the control and guidance systems.
- To inform the control system of the new performance parameters achievable using the new actuator sets so that reconfiguration can be take place if necessary.

The rest of this chapter gives an overview of the control allocation problem in aircraft, followed by a description of the objectives of the work done in this thesis. A brief overview of the work done is then given, including a description of the approach followed in solving

the control allocation problem. The chapter concludes with an overview of the thesis structure.

1.1 The control allocation problem

The general control allocation problem arises when the system to be controlled has more physical actuators than control objectives and the available actuators then need to be allocated to the objectives in some way [9]. In the case of aircraft there are four main control objectives: the control of the roll, pitch, yaw and thrust of the aircraft. These control objectives must be met using the aerodynamic actuators and engines that make up the control suite of the aircraft and this can range from the simple (ailerons, rudder, elevator, single engine) to the complex (canards, thrust vectoring, multiple engines, leading edge flaps, etc).

In order to exercise control over the four main control objectives of an aircraft, conventional controllers generate a single virtual command for each control objective. Specific actuators are then assigned through some mapping to these virtual commands. In the past, control allocation was done by physically attaching the controlling device (joystick, pedals, etc) to the physical actuators, resulting in a simple fixed mechanical mapping between commands and actuator deflections. Typically, roll commands were mapped to differential ailerons, pitch commands were mapped to the elevator and yaw commands were mapped to the rudder. Some additional complexity was often added in the form of an aileron-rudder interconnect to reduce adverse yaw effects due to aileron deflections. Any additional actuators such as flaps were not connected to the usual controlling device but were treated as settings to be adjusted manually by the pilot [10]. This arrangement is simple to implement for aircraft with simple actuator suits but it becomes difficult for aircraft with more complex actuator suits. It also places artificial limits on the manoeuvrability of the aircraft as well as resulting in degraded performance in the event of actuator failures.

As the fixed mechanical link between controller commands and actuators was replaced by fly-by-wire systems, the possibility of more complex allocations as well as control re-allocation was introduced, allowing control allocation schemes that overcome the shortcomings mentioned above. In order to maximise the performance of the aircraft, optimal control allocation can be employed to extract the maximum level of performance allowable by the actuators and control re-allocation can make use of redundant actuators to mitigate the effects of actuator failures and to meet secondary objectives. The control allocation system forms the link between the control system or pilot and the physical

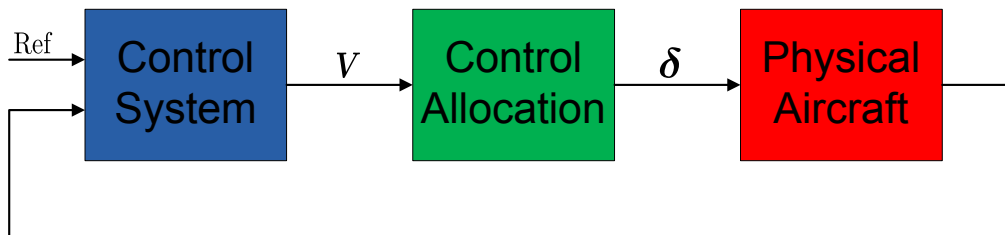


Figure 1.3: Control Allocation in a typical system

aircraft, as shown in fig 1.3.

The input \mathbf{V} is the moment commanded by the control system and the output δ consists of the commands to the individual actuators. Looking at figure 1.3, it is also clear that the control allocation block separates the physical aircraft from the control system commands. This highlights one of the possible benefits of control allocation and the one that this work focuses on, namely the possibility that actuator failures can be compensated for without the need to reconfigure the flight control law. If the physical aircraft changes due to a failure, the control allocation algorithm can potentially hide this change from the control system, avoiding unnecessary control system reconfiguration [8]. In the event that the failure is too severe to completely hide the effects from the control system, the effects can be minimised using control redundancy.

1.1.1 Typical goals of control allocation

Control allocation can be used to achieve a number of primary and secondary goals. Of the many possible goals, the application will determine which are primary and which are secondary. Which of these goals can be considered in a single application is determined both by the method used to solve the problem and by the allowed computational complexity of the problem. The goals that are typically considered are listed below:

- To provide an optimal mapping between control objectives and actuators while avoiding saturation in rate and position limits.
- To provide the maximum attainable moment that the actuators allow given constraints placed on actuator rate and position.
- To compensate for failures: Actuator re-allocation can be used to cancel the effects of a failed actuator and provide optimal performance of the failed system.

- Control redundancy can be used to achieve secondary goals such as drag minimisation, wing-load reduction, minimising actuator usage, radar cross-section reduction, removing adverse effects caused by direct mappings, etc.

Control re-allocation is primarily used when some failure has occurred on the aircraft. The primary goal in control re-allocation is then to redistribute control effort amongst the available physical actuators in order to ensure that the performance penalty due to the failure is minimised. All other possible goals are treated as secondary goals in this application or they are ignored completely. In such an application, secondary goals may initially be considered. If a solution cannot be found taking secondary goals into account, these goals can be removed from the problem in an effort to meet the primary goal.

1.1.2 The general problem formulation

Given the background of control allocation from the previous section, the basic mathematical details of the linear control allocation problem are presented in this section. This formulation is the generally accepted standard formulation used in literature and it is a simplification of the non-linear problem. This model is valid for small actuator deflections on an aircraft which is operating far from stall and is sufficient for the work covered in this thesis.

The basic control allocation problem can be described mathematically by the linear relationship:

$$\mathbf{V} = \mathbf{C}\boldsymbol{\delta} \tag{1.1}$$

subject to the constraints $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$.

In this equation, \mathbf{C} is the matrix that describes the effectiveness of each actuator, $\boldsymbol{\delta}$ is the control vector, constrained by the maximum and minimum values δ_{max} and δ_{min} and \mathbf{V} is the resulting moment, comprising components roll (L), pitch (M) and yaw (N). A fourth virtual actuator for thrust can also be added although this is typically not done. If necessary, this formulation can be expanded in order to include actuator rate limits. This is achieved by changing the basic formulation of equation (1.1) to be a function of time. See [11] for more detail on the time-based formulation. Besides adding the ability to handle rate constraints, a time-based formulation is also more capable of dealing with actuator dynamics. The time-based formulation is not used here since it is assumed that actuator rate limits are taken into account by the control system and using the time-based

formulation places additional constraints on the time allowed to solve for the problem since solutions must be found for each discrete time-step of the control process.

Most control allocation applications are focussed on matching the desired moments with the achieved moments while ignoring adverse forces, so that, when equation 1.1 is expanded, it becomes:

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix} = \begin{bmatrix} c_{l_1} & c_{l_2} & \cdots & c_{l_n} \\ c_{m_1} & c_{m_2} & \cdots & c_{m_n} \\ c_{n_1} & c_{n_2} & \cdots & c_{n_n} \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_n \end{bmatrix} \quad (1.2)$$

where n is the number of physical actuators available to the aircraft. It is possible to expand this equation to include a virtual thrust actuator.

Given this basic formulation, a number of commonly-used goal-based formulations can be described. [12] describes the following four commonly-used mathematical formulations of the control allocation problem:

- Direct allocation problem: The direct allocation method attempts to match the desired moment in both magnitude and direction. If this is not possible, the desired moment direction is maintained and the magnitude is scaled to what is possible. This method can be described mathematically as follows:

Find a real number ρ and a vector $\boldsymbol{\delta}_1$ such that $J = \rho$ is maximised, subject to

$$(\mathbf{C})\boldsymbol{\delta}_1 = \rho\mathbf{V}_d \quad (1.3)$$

and $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$. If $\rho > 1$, let $\boldsymbol{\delta} = \boldsymbol{\delta}_1/\rho$. Otherwise, let $\boldsymbol{\delta} = \boldsymbol{\delta}_1$.

- Error minimisation problem: This formulation takes the form of an optimisation problem where the goal is to minimise the difference between the desired moment, \mathbf{V}_d , and the attained moment, $\mathbf{C}\boldsymbol{\delta}$. This method can be described mathematically as follows:

Find a control vector $\boldsymbol{\delta}$, such that:

$$J = \|\mathbf{C}\boldsymbol{\delta} - \mathbf{V}_d\| \text{ is minimised} \quad (1.4)$$

subject to $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$.

[13] describes the error minimisation problem where an additional disturbance term to account for disturbances due to actuator failures, is included.

- Control minimisation problem: This formulation has the same form as the previous formulation except that the goal is to find the control vector that meets the moment goal while minimising actuator usage. The formulation is described mathematically as follows:

Given a vector of desired actuator positions, $\boldsymbol{\delta}_d$, and a vector which meets the moment goals, $\boldsymbol{\delta}_1$, such that $\delta_{min} \leq \boldsymbol{\delta}_1 \leq \delta_{max}$, find a vector $\boldsymbol{\delta}$ such that:

$$J = \|\boldsymbol{\delta} - \boldsymbol{\delta}_d\| \text{ is minimised} \quad (1.5)$$

subject to

$$(\mathbf{C})\boldsymbol{\delta} = (\mathbf{C})\boldsymbol{\delta}_1 \quad (1.6)$$

and $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$.

This formulation does not work if the desired moment is not attainable since the equality constraint $(\mathbf{C})\boldsymbol{\delta} = (\mathbf{C})\boldsymbol{\delta}_1$ becomes violated. This formulation is therefore usually used as a secondary step in a two-step allocation process once the moment requirements have already been met [12].

- Mixed optimisation problem: This formulation is a combination of the error-minimisation and the control-minimisation problems. A parameter ϵ determines which problem is given priority in the optimisation process. This formulation can be described mathematically as follows:

$$J = \|\mathbf{C}\boldsymbol{\delta} - \mathbf{V}_d\| + \epsilon\|\boldsymbol{\delta} - \boldsymbol{\delta}_p\| \quad (1.7)$$

subject to $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$.

The formulations given above are the general formulations seen in the majority of literature to date. Most of the work done to date on control allocation focuses on the application of various solution methods and the modification of these methods to overcome various shortcomings. While most articles mention the ability of control allocation systems to handle failures, few explicitly investigate this property. In addition to the lack of detailed literature on failure cases, most literature considers only the roll, pitch and yaw moments in the standard formulations and they ignore the adverse forces caused by controller commands.

1.1.3 Typical solution methods

Given the typical goals of control allocation and some common problem formulations, some of the more common solution methods that have been applied to the control allocation

problem to date are described in this section. [10] divides the solution methods into two groups, non-optimal and optimal methods. This structure is used here as well, with the non-optimal methods discussed first, followed by the optimal methods. In addition to [10], detailed summaries of the various control allocation solution techniques can also be found in [11] and [2].

Non-optimal methods

Non-optimal solution methods are those methods that are not able to find solutions to the control allocation problem that utilise the entire attainable moment space (AMS). While non-optimal solutions generally provide answers very quickly, they are unable to provide full utilisation of the aircraft's capabilities. They may also provide solutions which are not achievable by the aircraft. A number of non-optimal methods used in previous literature are discussed below.

Rule-based and offline methods: A number of offline or rule-based methods have been used in literature. These methods have generally been used to perform control re-allocation in failed systems where time and/or allowed computational complexity is very limited. The general approach with these methods is to have a pre-defined set of control allocations or rules for allocating controls which the control system selects according to some criteria.

[14] used an offline nonlinear constrained optimisation approach where solutions to particular failures were calculated offline and stored in a look-up table. In the event of a failure, the fault detection and diagnosis system then diagnoses the failure and selects the appropriate control mixing from the lookup table.

[8] proposes a rule-based method to solve the control allocation problem. In this case, a number of fault configurations and a number of actuator behaviour modes are defined. Certain actuator behaviour modes are then applied to certain failure configurations. Each actuator behaviour mode is defined by a number of fixed laws that define how the actuators are deflected for that mode.

While rule-based and offline methods are very quick, they offer limited functionality and can only provide solutions for a fixed number of pre-defined failure cases. These methods will not provide optimal solutions for failures that have not been provided for. These methods have the additional disadvantage that the designer is responsible for defining and providing solutions for all of the failure cases that need to be considered, which can become a difficult and time-consuming task.

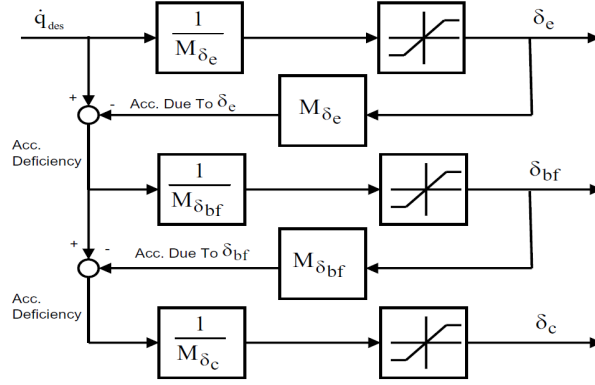


Figure 1.4: Daisy-chain control allocation [2]

Daisy-chaining: The idea behind the daisy-chaining method is that the available actuators are divided into sets of actuators that perform the same task. These sets are then prioritised according to desired actuator usage, actuator effectiveness or a combination of the two. Control effort is then firstly assigned to the set with the highest priority until either the requirements are met or the maximum performance available from that set is reached. If the former condition is met, the remaining sets of actuators are not used while in the latter case the set with the next highest priority is used to try to meet the desired performance criterion. This process is repeated either until the performance requirements are met or until all the sets are saturated. This process is illustrated in figure 1.4.

This method is simple to implement but it has the tendency to saturate actuators as opposed to distributing control effort evenly among all the actuators. Furthermore, it may not find optimal solutions in all cases and it has been shown to also provide solutions that are not feasible [11]. This method also places the burden of selecting the actuator sets on the designer and this may become a very complex and difficult task.

Pseudo-Inverse methods: Pseudo-Inverse methods are based on a quadratic formulation of the control allocation problem. The cost function is defined as follows:

$$J = \frac{1}{2} (\mathbf{C}\boldsymbol{\delta} - \mathbf{V}_d)^T (\mathbf{C}\boldsymbol{\delta} - \mathbf{V}_d) \quad (1.8)$$

subject to $\delta_{min} \leq \boldsymbol{\delta} \leq \delta_{max}$, where \mathbf{V}_d is the desired moment.

In the case where the inequality constraints are not active, the control vector that minimises the cost function, J , is then described by:

$$\boldsymbol{\delta} = \mathbf{C}^T (\mathbf{C}\mathbf{C}^T)^{-1} \mathbf{V}_d \quad (1.9)$$

The major drawback of this method is the fact that equation (1.9) only holds if the constraints on the actuators are not active. The result is that this method will not provide solutions for all attainable moments and it will not provide optimal solutions for problems where the desired moment is unattainable [11]. Methods have been described to increase the possibility of optimal solutions using this method. These methods generally involve calculating an initial control vector using equation 1.9. The saturated elements are then removed from the \mathbf{C} matrix and the equation is calculated again. This process is repeated until an acceptable solution is found or until a specified number of actuators have been saturated. This variation of the pseudo-inverse method is known as the redistributed pseudo inverse [2].

Optimal methods

Solutions to the control allocation problem that are termed optimal are those methods that are capable of attaining moments in the entire attainable moment space. There are various optimal solution methods that have been used in literature and the most common of these are discussed below:

Direct allocation: The direct allocation method as described by [15] is a two-step process that was developed to solve the direct allocation problem defined in section 1.1.2. The first step is to determine the attainable moment set of the aircraft and the second step is to determine the controls that are able to generate moments within the AMS without violating the constraints placed on the controls. For calculating which controls generate the desired moment, two methods have been used: Facet searching and Bisecting edge search [10].

The direct allocation method has the property of maintaining the direction of the desired vector even when its magnitude cannot be matched. Initially, the second step of the direct allocation method was accomplished using the facet searching technique and the method could only handle three objectives [15]. Later implementations provided increased speed through using the Bisecting edge search method and [10] expanded the method to handle four or more objectives.

Optimisation-based solutions: Optimisation-based solutions were initially avoided due to their computational complexity which resulted in fears that they would not be able to find solutions quickly enough [12]. As the computational power available in small on-board computer systems has increased, these methods have become more feasible and subsequently, more popular. The benefit of using optimisation-based solutions over other

optimal methods is that these methods allow for the extension to many objectives with little difficulty. Both linear and non-linear formulations are considered in literature and the properties of both of these are briefly discussed below:

- **Linear programming**

Given the general linear formulation of the control allocation problem defined in section 2.2, with linear position constraints, defining a linear cost function will result in a linear programming problem. Standard linear programming solving methods can then be used to solve the control allocation problem. One commonly-used linear programming method is the Simplex method and this method has been used to solve linear formulations of the control allocation problem [10]. Interior point methods have also been investigated by [16].

[17] reformulated the error minimisation and control minimisation problems as linear programs and [12] reformulated the direct allocation problem and the mixed optimisation problem as linear programs. [12] indicates that the mixed-optimisation problem is the best formulation to use for linear programming methods.

Formulating the control allocation problem as a linear program for more complex control allocation formulations is not always intuitive and the solutions it provides tend not to use evenly distributed control effort, rather using the most effective actuators only [12], [10].

- **Non-linear programming**

Defining a non-linear cost function and/or non-linear constraints results in a non-linear programming problem. The most commonly used non-linear formulation in control allocation is the l_2 -norm, largely due to the extensive use of the pseudo-inverse method [10]. Solving the control allocation problem formulated using the l_2 -norm with constraints and optimisation-based methods has also been done and is investigated by [18] and [19] applies the SQP method to a quadratic formulation of a 3-DOF control allocation problem. When using the l_2 -norm, control effort tends to be more distributed among the available actuators than is the case for the linear programming methods. Another benefit of the l_2 -norm is that for error-based formulations, larger errors are penalised more heavily.

The control allocation problem is now well defined. The basic linear formulation of the control allocation problem is described in section 1.1.2 and the solution methods discussed

in section 1.1.3 cover the whole range of commonly-used methods to solve the control allocation problem. The rest of this chapter deals with the specific details of the work presented in this thesis.

1.2 Objectives of research

This body of work has two primary objectives. The first is to develop a control allocation system that forms part of the FTC system mentioned at the start of this chapter. The second is to investigate the effectiveness of this control allocation system for a variety of conditions. The sub-objectives for each of these primary objectives are described below.

1.2.1 Control allocation system

The control allocation system is expected to interface with other sub-systems of the FTC system, most notably the system I.D. subsystem and the control reconfiguration sub-system, via the supervisor. The focus of the control allocation system itself is on dealing with actuator failures. The control allocation system should have the following properties:

- Be as general as possible so that it can be applied to different aircraft with minimal alteration.
- Be capable of dealing with the linear control allocation problem.
- Be capable of handling multiple failures.
- Be usable as a tool to aid in the understanding and evaluation of control allocation formulations.

1.2.2 Evaluation of effectiveness

There are a number of factors that influence the effectiveness of the control allocation system. Some of the factors that are investigated in this thesis are:

- Investigating different control allocation configurations.
- Investigating different categories of failures.
- Investigating the effects of varying levels of control redundancy on the effectiveness of control allocation.

The knowledge gained from this work will be beneficial in making decisions on FTC system design as well as aircraft design by revealing the strengths and weaknesses of control allocation for a variety of conditions and by revealing what types of actuator failures are more difficult to deal with. This will reveal what types of aircraft configurations are likely to benefit the most from FTC systems as well as providing valuable information on what actuators should be given special attention when it comes to improving actuator redundancy for aircraft, thus influencing the designs of future redundant aircraft.

1.3 Overview of the work done

This section provides an overview of the work described in this thesis and where it fits into the control allocation literature.

The control allocation strategy

Most control allocation strategies used in literature make use of real-time or dynamic control allocation where control allocation is done in every discrete time-step of the control process, as illustrated by figure 1.3. This is done so that the control allocation system can take rate-constraints and actuator dynamics into account and also so that the control allocation process can quickly adapt the aircraft to failures. Using this strategy places hard limits on the amount of time available to perform control allocation, resulting in the necessity for methods that can calculate solutions very quickly. The non-optimal solution methods proposed in previous literature and discussed in section 1.1.3 were used due to their quick solution times.

In this FTC system, the first line of defence after a failure is an adaptive control system that quickly reacts to stabilise the aircraft. An online system identification process is then used to find the new aircraft parameters and once this information is available, the control allocation system is activated to restore the aircraft to its optimal performance. Given that the control allocation system proposed in this thesis is not the first line of defence in the event of a failure, it is not as time-critical a process. The solution proposed uses control allocation to restore the aircraft to the desired level of performance after a failure has occurred by updating the control mixing vectors used by the aircraft. This update is performed only after a failure has occurred and finding a solution for each time-step is therefore not necessary. This opens up the possibilities for additional goals that would not be considered in a time-critical application as well as making optimisation-based methods more feasible since the most important goal is minimising the error between the desired

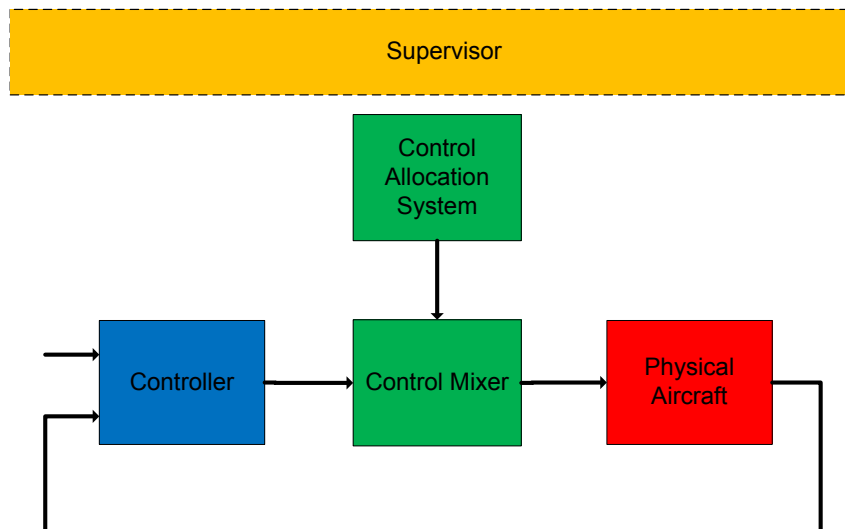


Figure 1.5: Semi-static control allocation architecture

and the achievable performance of the system. The architecture of the system developed here can be illustrated as shown in figure 1.5.

This figure shows that the control allocation block lies outside of the main control loop and it only updates the control mixing vectors when necessary and requested by the supervisor system. A control allocation architecture that fits into an FTC system as described above will therefore have the following properties:

- It must be capable of minimising the errors between desired and achievable virtual actuator performance for both primary and adverse effects for the full 6-DOF problem.
- It must take into account the trim biases of all of the available actuators so that the resulting combination of trim biases and mixing gains do not saturate the physical actuators.
- It must be able to handle a wide variety of failure cases involving both single- and multi-actuator failures.
- It must be applicable to a wide variety of aircraft with little or no alteration.
- The issue of time constraints should be kept in mind but is secondary to the above properties.

Taking these properties into account, the solution methods discussed in section 1.1.3 can be evaluated for applicability. Firstly, consider the non-optimal solution methods.

The principle advantages of these methods is that they provide solutions very quickly while the major drawbacks are limited capacity to handle failures and limited use of the attainable moment space. The advantages of these methods are mitigated by the fact that the problem in this particular case of control allocation is not as time critical while the drawbacks limit the usefulness of these methods when considering many possible failure scenarios. Rule-based, offline, daisy-chain and pseudo-Inverse methods are therefore not suitable.

Optimal methods are more capable of successfully minimising the errors between desired and achievable moments. Particularly, optimisation-based methods are attractive options since they allow for multiple objectives to be added with minimal difficulty. Additionally, the fact that these methods typically take longer to find solutions than the non-optimal methods discussed is inconsequential due to the differing time constraints placed on this system. The direct allocation method, while it has been modified to handle more than four objectives, needs to calculate the AMS for each failure case and defining the problem for many objectives is a complex task. The methods based on linear or non-linear optimisation therefore appear to be the most suitable.

Approach to the problem

As seen in section 1.1, much work has been done on the control allocation problem to date. This work encompasses a wide variety of problem formulations and solution methods to the control allocation problem. Very little literature, however, was found that discusses the process of investigating the control allocation problem in aircraft from the level of the aircraft model. Additionally, the nature of the control allocation system developed here differs as follows:

- The control allocation system is semi-static and only updates the mixing vectors when required.
- Adverse forces due to actuator deflections are taken into account.

With these important differences in mind, the control allocation problem was addressed by starting at the aircraft model and investigating what the requirements and effects of control allocation are on the physical aircraft. The aircraft model was kept general in order to satisfy the need for the FTC system as a whole to be applicable to a large variety of aircraft.

From the aircraft model, an intuitive control allocation formulation was developed that can be directly related to the force and moment equations that make up the aircraft model.

This formulation was matched to the broad optimisation category of multi-objective optimisation and an existing SQP optimisation engine was applied to the problem. Once this system was completed and tested, a number of problem areas and interesting applications were identified. The control allocation formulation was subsequently enhanced in order to address these issues. The final system's capabilities and limits were then extensively tested using a number of aircraft test cases.

The resulting control allocation system

The resulting control allocation system is a versatile tool that can be used for a variety of conventional aircraft configurations with no alteration to the underlying system. Additionally, the system can be adapted for use with unconventional aircraft with little alteration to the system. The resulting system takes as inputs the following data:

- Aircraft configuration and actuator effectiveness
- Desired aircraft performance data

Once the control allocation procedure is complete, the system provides the following information:

- Aerodynamic and thrust control mixing vectors for each virtual command
- The new achievable moments of the aircraft
- The new adverse effects of each virtual command

The resulting system is then an algorithm that is run only when control allocation is required. The aircraft is then provided with new actuator mappings to allow for the best possible performance in the event of a failure. Some additional experimental features are also available to investigate additional functionality and uses of the system aside from the primary goal of recovering the aircraft performance after a failure has occurred.

1.4 Thesis overview

In chapter 2, a general aircraft model is developed so that the control allocation system can be built from this base. In chapter 3, the control allocation formulation is developed using the aircraft model from chapter 2 and the goals and constraints on the system are identified. Chapter 4 discusses optimisation and how the control allocation formulation

is handled by the chosen optimisation engine. Chapter 5 describes the practical implementation of the system and chapter 6 gives the test results for the base system. Chapter 7 deals with the expansion of the formulation to handle a number of additional tasks and describes the final practical implementation and testing of the expanded system. In Chapter 8 the conclusion and some recommendations are given.

Chapter 2

Aircraft model

2.1 General Aircraft model

The previous chapter introduced the control allocation problem and section 1.1.2 provided the generally accepted basic formulation used in literature. This formulation is however a high-level description of the control allocation problem which provides no insight into the practical requirements of control allocation. In order to fully understand the control allocation problem in aircraft, a model of the physical aircraft and how it reacts to control inputs is required. To this end, a general 6-DOF model is developed in order to describe how the aircraft is affected by actuator positions and thrust settings. The aircraft model developed assumes that the aircraft is a rigid body with six degrees of freedom moving in inertial space.

2.1.1 Axis System definitions

The various axis systems used to describe the aircraft model are defined. These include: inertial axes, in which Newton's equations of motion apply, body axes, which are used to describe the aircraft motion within inertial axes and wind axes, which are a type of body axes in which the aerodynamic model is more easily formulated.

Inertial axes

In order to use Newton's laws, an inertial axis system needs to be defined [20]. An inertial axis system is approximated using an earth-fixed axis system with its origin fixed to a point on the earth's surface, as shown in figure 2.1. This right-handed axis system has a positive Z_I -axis pointing toward the centre of the earth, an X_I -axis perpendicular to this,

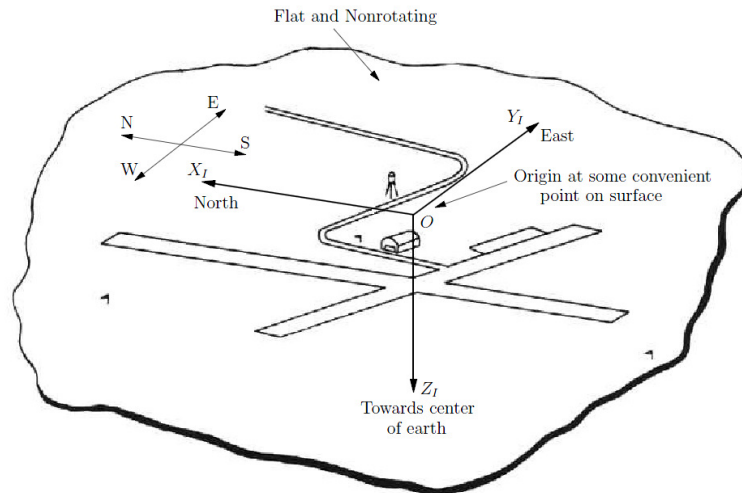


Figure 2.1: The inertial reference frame [3]

pointing North and the Y_I -axis completing the right-handed axis system. In addition to this, it is assumed that gravity is uniform throughout the inertial reference frame so that the center of gravity of the aircraft and mass center of the aircraft coincide [21]. Finally, the localised motion of the aircraft allows us to assume a flat and non-rotating earth.

Body axes

The Body axes used here are defined as a set of axes fixed to the aircraft with its origin at the centre of gravity of the aircraft. The Y_B -axis points along the right wing and the X_B -axis is perpendicular to the Y_B -axis, pointing forward along the aircraft's longitudinal reference line. The Z_B -axis completes the right-hand axis system, pointing downward in relation to the aircraft. These axes are illustrated in figure 2.2.

Wind axes

Wind axes are defined as axes that are attached to the aircraft, like body axes, with the difference that the axes are rotated so that the X_W -axis is pointing along the velocity vector, as shown in figure 2.3. The axes are rotated about the Y_B -axis by the angle of attack α and about the resulting new Z -axis by the angle of sideslip β . The Z_W -axis lies in the plane of symmetry of the aircraft. This axis system has the advantage that the aerodynamic model is reduced to its simplest form [20].

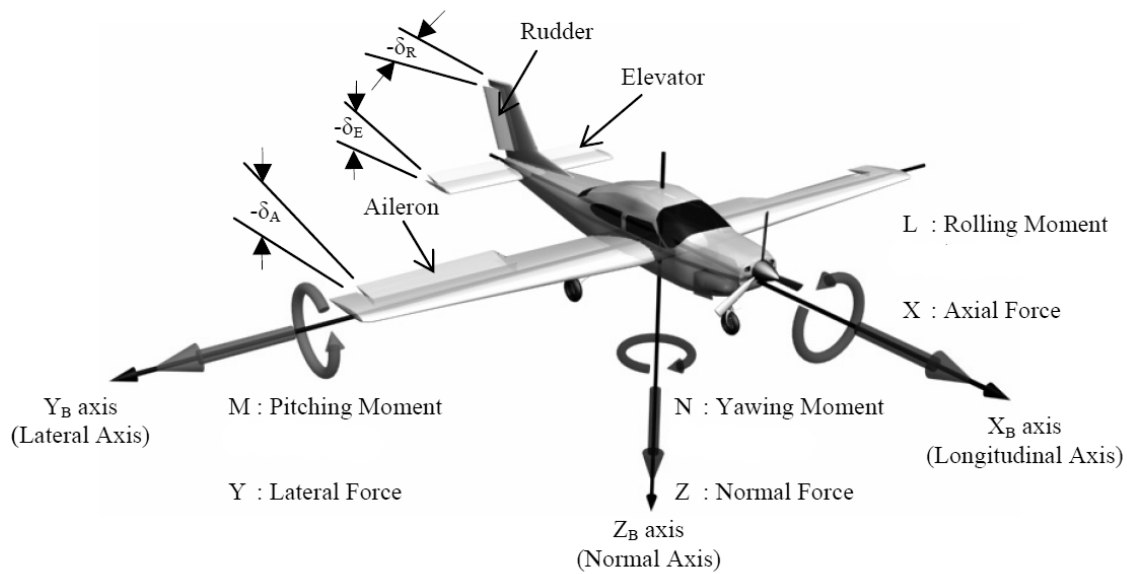


Figure 2.2: The body axis system [4]

2.1.2 Notation

Figure 2.2 also shows the notation used for the forces and moments acting on the aircraft. The Forces are:

- X - Axial force
- Y - Lateral force
- Z - Normal force

and the moments are:

- L - Roll moment
- M - Pitch moment
- N - Yaw moment

Positive deflections of the ailerons, elevators and rudders are defined such that they produce negative moments while positive deflections of flaps are defined such that they produce negative Z -axis forces.

The axes labelling convention used in equations is as follows: Forces and moments in wind axes are written without subscripts while forces and moments in other are indicated by subscripts, for example the X -force in body axes is X_B .

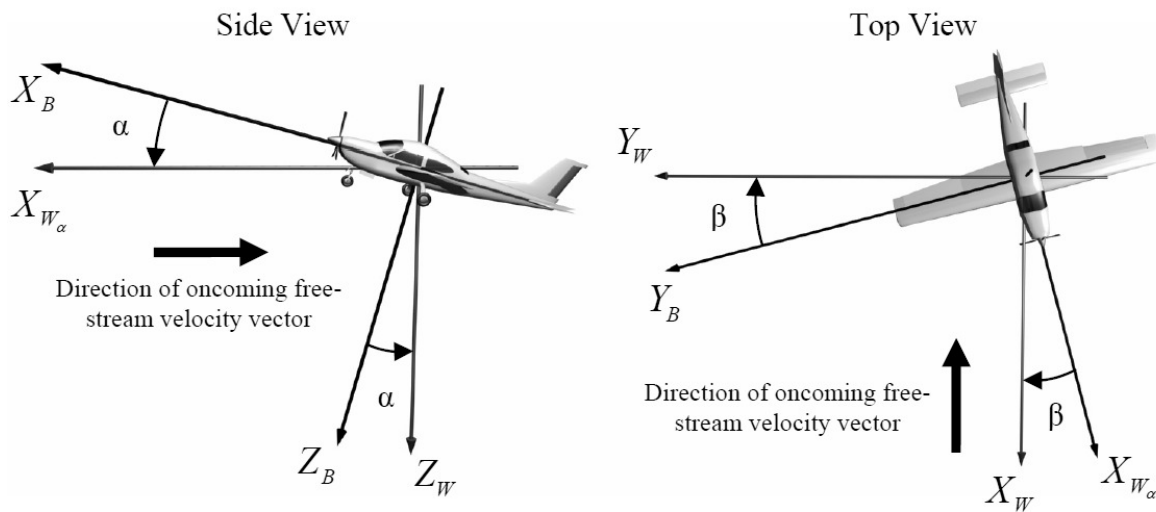


Figure 2.3: The wind axis system [4]

2.1.3 Euler angles and axes transformations

Since a number of axis systems are used in the aircraft modeling process some method of describing the orientation of one set of axes with respect to another is needed, as well as some method of transforming co-ordinates from one set of axes to another. To this end, Euler angles and the direction cosine matrix (DCM) are described.

Euler angles

Euler angles provide a simple way of characterising the attitude of one system of axes relative to another. This is done by three successive non-commutable rotations, meaning that the rotation order is of importance. The Euler method used is the 3-2-1 method which starts with a rotation about the Z -axis through the yaw angle ψ , followed by a rotation about the Y -axis through the pitch angle θ and finally completed by a rotation about the X -axis through the roll angle ϕ . These three rotations are illustrated in figure 2.4.

Using Euler angles to describe the attitude of a set of axes presents some limitations. These limitations will not however be a problem in the context of the control allocation system since the attitudes between the systems of axes being used are expected to remain within these limitations. The limitations relate to a possible ambiguity at $\theta = \pm\frac{\pi}{2}$.

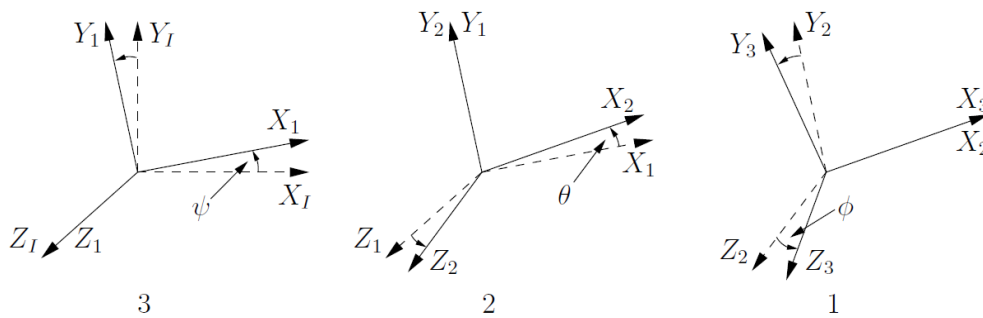


Figure 2.4: The Euler 3-2-1 rotation sequence [3]

Axis transformations

Using the Euler rotation angles ψ , θ and ϕ from the previous section, it can be shown that a transformation matrix exists to transform vectors from one axis system to another. This matrix is known as the Direction Cosine Matrix (DCM), for a derivation of the DCM, see [22]. The DCM is defined as follows:

$$DCM^{3-0} = \begin{bmatrix} C_\psi C_\theta & S_\psi C_\theta & -S_\theta \\ C_\psi S_\theta S_\phi - S_\psi C_\phi & S_\psi S_\theta S_\phi + C_\psi C_\phi & C_\theta S_\phi \\ C_\psi S_\theta C_\phi + S_\psi S_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi & C_\theta C_\phi \end{bmatrix} \quad (2.1)$$

where the superscript 3-0 describes the direction of the transformation, in this case from some arbitrary axes 0 to some arbitrary axes 3. $C_{(\cdot)}$ and $S_{(\cdot)}$ represent the *Cos* and *Sin* of the angles indicated in the subscripts.

It will also prove useful to be able to transform vectors in the reverse direction. In order to transform the axes in the opposite direction, the inverse of the DCM given in equation 2.1 is required. It can be shown that the DCM is orthogonal and the inverse is simply its transpose [20].

2.1.4 Forces and Moments

The sections above have described the mathematical tools required to formulate a mathematical model of an aircraft. These tools are now applied in order to develop a model that can be used to describe the control allocation problem from first principals. Figure 2.5 shows a block diagram of a general aircraft model. The control allocation problem deals with the forces and moments generated by controller commands and as such only the force and moment model from the figure is applicable to this problem.

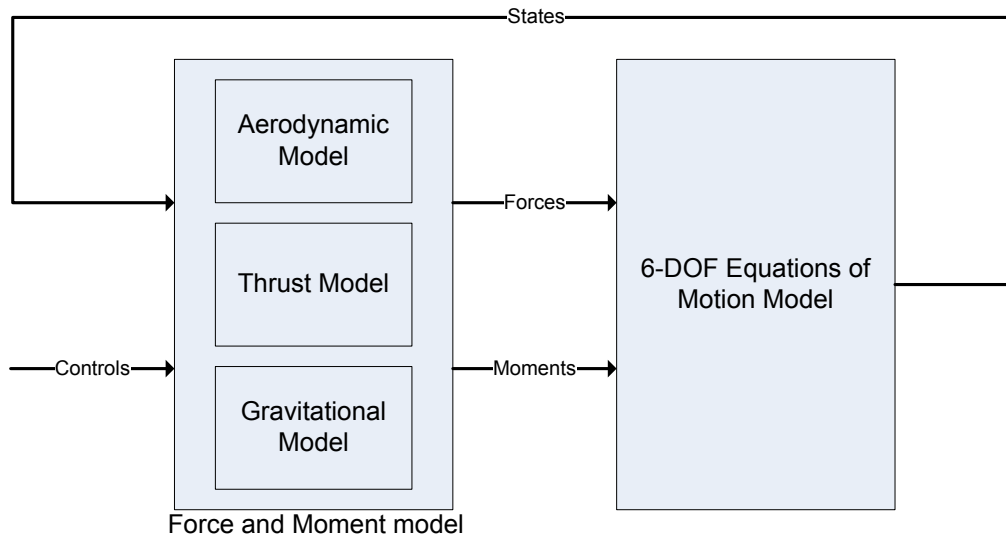


Figure 2.5: General aircraft model block diagram [5]

The forces and moments acting on the aircraft can be described in terms of the aerodynamic, thrust and gravitational models [22]. The force and moment equations can therefore be expanded as follows:

$$X = X^A + X^T + X^G \quad (2.2)$$

$$Y = Y^A + Y^T + Y^G \quad (2.3)$$

$$Z = Z^A + Z^T + Z^G \quad (2.4)$$

$$L = L^A + L^T + L^G \quad (2.5)$$

$$M = M^A + M^T + M^G \quad (2.6)$$

$$N = N^A + N^T + N^G \quad (2.7)$$

with all of the forces and moments above coordinated in wind axes. The contributions of the aerodynamic, thrust and gravitational effects on the aircraft are discussed in the following sections.

Aerodynamic

Describing the aerodynamic model of an aircraft is considered to be one of the more complex parts of the aircraft modelling process. The aerodynamic effects on the aircraft are complex and difficult to model and as a result, aerodynamic models that represent simplified approximations of the true aircraft while maintaining acceptable accuracy are

usually used. This results in aerodynamic models that are only valid for small operating regions [22]. These limitations must be kept in mind and it is important that the model is not used outside of its valid envelope. The aerodynamic components of the forces and moments acting on the aircraft are defined below in wind axes:

$$X^A = -qSC_D \quad (2.8)$$

$$Y^A = qSC_y \quad (2.9)$$

$$Z^A = -qSC_L \quad (2.10)$$

$$L^A = qSbC_l \quad (2.11)$$

$$M^A = qS\bar{c}C_m \quad (2.12)$$

$$N^A = qSbC_n \quad (2.13)$$

with $q = \frac{1}{2}\rho V^2$ the dynamic pressure as a function of ρ , the air density and V , the airspeed. The dynamic pressure term together with S , the wing reference area, gives the aerodynamic force parameter used to normalise the force coefficients. The dynamic pressure term together with b , the wingspan, gives the aerodynamic moment parameter used to normalise the lateral moment coefficients. Finally, the dynamic pressure term together with \bar{c} , the mean aerodynamic chord, gives the aerodynamic moment parameter used to normalise the longitudinal moment coefficients.

The coefficients C_D , C_y , C_L , C_l , C_m and C_n are the non-dimensional aerodynamic coefficients of the aircraft and they are expanded into their component forms below. The equations shown are valid for a standard small incidence angle model and the lateral and longitudinal equations are decoupled, the coefficients are stability-derivative based and apply for a linear aerodynamic coefficient model [23]:

$$C_D = C_{D_0} + \frac{C_L^2}{\pi A e} \quad (2.14)$$

$$C_y = C_{y_0} + C_{y_\beta}\beta + \mathbf{C}_{y_\delta}\boldsymbol{\delta} \quad (2.15)$$

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + \mathbf{C}_{L_\delta}\boldsymbol{\delta} \quad (2.16)$$

$$C_l = C_{l_0} + C_{l_\beta}\beta + \mathbf{C}_{l_\delta}\boldsymbol{\delta} \quad (2.17)$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + \mathbf{C}_{m_\delta}\boldsymbol{\delta} \quad (2.18)$$

$$C_n = C_{n_0} + C_{n_\beta}\beta + \mathbf{C}_{n_\delta}\boldsymbol{\delta} \quad (2.19)$$

Where C_{D_0} is the parasitic drag coefficient, A is the aspect ratio of the wing, e is the Oswald efficiency factor and C_{L_0} , C_{l_0} , C_{m_0} , C_{n_0} and C_{y_0} are the static lift, roll, pitch, yaw and side-force coefficients, respectively. The C_{l_0} , C_{n_0} and C_{y_0} terms are usually not included in these equations since they are typically zero due to the symmetry of the aircraft. They are included in this application since they are used to capture the force and moment biases that may be present, for example due to an actuator failure. The effects of aerodynamic actuators are captured by the vectors $\boldsymbol{\delta}$ and $\mathbf{C}_{(\cdot)_\delta}$. These are the actuator command vector and control effectiveness vectors respectively.

The control effectiveness of the actuators is assumed to be linear. While this will negatively affect the accuracy of the calculated mixing and biasing vectors, it greatly simplifies the control allocation problem.

Thrust

The general aircraft model is developed so that the aircraft can have any number of engines offset from the X_B -, Y_B - and Z_B -axes as well as offset by the angle θ_e from the body axes. Figure 2.6 illustrates the engine position and angle offset relative to the aircraft body axes. The engine reference frame is defined such that the force vector produced by an engine acts through the origin of the reference frame along the X_E -axis:

$$\mathbf{F}_E^T = \begin{bmatrix} F_{\delta_t} \\ 0 \\ 0 \end{bmatrix} \delta_t \quad (2.20)$$

and the moment caused by the spinning rotor is about the X_E -axis only:

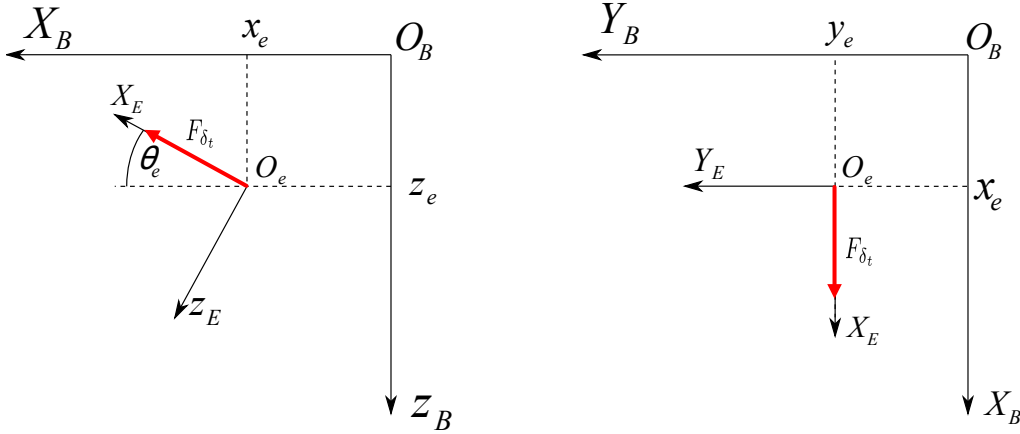


Figure 2.6: Engine position and angle offsets from body axes

$$M_E^T = \begin{bmatrix} L_{\delta_t} \\ 0 \\ 0 \end{bmatrix} \delta_t \quad (2.21)$$

The dynamics of the engines are not taken into account by the control allocation system and the forces and moments caused by the rotors are assumed to be the steady-state forces and moments caused by a particular thrust setting. F_{δ_t} and L_{δ_t} therefore represent scaling terms that relate a specific thrust setting, δ_t , to a force and moment that would be generated at that thrust setting. This limitation means that these dynamics will need to be taken into account elsewhere in the aircraft control systems. Additional effects could be added in a higher fidelity model of the thrust effects. Effects such as propeller swirl and changes of dynamic pressure on control surfaces that are within the propeller slipstream could be modelled. These effects are not included in this model as the focus of the thesis is on the control allocation problem and not a high fidelity aircraft model. These effects should however be kept in mind for practical applications.

Transforming the thrust vector from the engine reference to the aircraft body reference frame, the DCM is used:

$$\begin{bmatrix} X_B^T \\ Y_B^T \\ Z_B^T \end{bmatrix} = (DCM^{BE}) \begin{bmatrix} F_{\delta_t} \\ 0 \\ 0 \end{bmatrix} \delta_t \quad (2.22)$$

So that the forces caused by the thrust in body axes for each engine are:

$$\begin{bmatrix} X_B^T \\ Y_B^T \\ Z_B^T \end{bmatrix} = \begin{bmatrix} C_{\theta_e} \\ 0 \\ -S_{\theta_e} \end{bmatrix} F_{\delta_t} \delta_t \quad (2.23)$$

Expanding this to include the effects of multiple engines, the result is:

$$X_B^T = \mathbf{C}_{x_{\delta_t}} \boldsymbol{\delta}_t \quad (2.24)$$

$$Y_B^T = 0 \quad (2.25)$$

$$Z_B^T = \mathbf{C}_{z_{\delta_t}} \boldsymbol{\delta}_t \quad (2.26)$$

with $\boldsymbol{\delta}_t = [\delta_{t_1} \ \delta_{t_2} \ \dots \ \delta_{t_p}]^T$ the vector of p thrust settings and

$$\mathbf{C}_{x_{\delta_t}} = \begin{bmatrix} C_{\theta_{e_1}} F_{\delta_{t_1}} & C_{\theta_{e_2}} F_{\delta_{t_2}} & \dots & C_{\theta_{e_p}} F_{\delta_{t_p}} \end{bmatrix} \quad (2.27)$$

$$\mathbf{C}_{z_{\delta_t}} = \begin{bmatrix} -S_{\theta_{e_1}} F_{\delta_{t_1}} & -S_{\theta_{e_2}} F_{\delta_{t_2}} & \dots & -S_{\theta_{e_p}} F_{\delta_{t_p}} \end{bmatrix} \quad (2.28)$$

the vectors of transformations for the engines.

The moments due to the thrust of the engines are then calculated using the forces for each engine and the offsets of the engine along the X_B -, Y_B - and Z_B -axes. The result is:

$$L_B^T = \mathbf{C}_{z_{\delta_t}} \mathbf{Y}_e \boldsymbol{\delta}_t \quad (2.29)$$

$$M_B^T = \mathbf{C}_{z_{\delta_t}} \mathbf{X}_e \boldsymbol{\delta}_t + \mathbf{C}_{x_{\delta_t}} \mathbf{Z}_e \boldsymbol{\delta}_t \quad (2.30)$$

$$N_B^T = \mathbf{C}_{x_{\delta_t}} \mathbf{Y}_e \boldsymbol{\delta}_t \quad (2.31)$$

where the vectors \mathbf{X}_e , \mathbf{Y}_e and \mathbf{Z}_e are p -length vectors containing the x_e -, y_e - and z_e -offsets of the engines.

Next, the moments caused by the spinning rotors are also added. Since the thrust model is defined with engines offset at an angle θ_e from body axes, the moments generated by the rotor need to be coordinated into body axes. The resulting moments due to rotor effects for each engine are therefore calculated using the DCM:

$$\begin{bmatrix} L_B^R \\ M_B^R \\ N_B^R \end{bmatrix} = (DCM^{BE}) \begin{bmatrix} L_{\delta_t} \\ 0 \\ 0 \end{bmatrix} \delta_t \quad (2.32)$$

so that, for each engine:

$$\begin{bmatrix} L_B^R \\ M_B^R \\ N_B^R \end{bmatrix} = \begin{bmatrix} C_{\theta_e} \\ 0 \\ -S_{\theta_e} \end{bmatrix} L_{\delta_t} \delta_t \quad (2.33)$$

Expanding this in order to handle the effects of multiple engines:

$$L_B^R = \mathbf{C}_{lr_{\delta_t}} \boldsymbol{\delta}_t \quad (2.34)$$

$$M_B^R = 0 \quad (2.35)$$

$$N_B^R = \mathbf{C}_{nr_{\delta_t}} \boldsymbol{\delta}_t \quad (2.36)$$

with $\boldsymbol{\delta}_t = [\delta_{t_1} \ \delta_{t_2} \ \dots \ \delta_{t_p}]^T$ the vector of p thrust settings and

$$\mathbf{C}_{lr_{\delta_t}} = [C_{\theta_{e_1}} L_{\delta_{t_1}} \quad C_{\theta_{e_2}} L_{\delta_{t_2}} \quad \dots \quad C_{\theta_{e_p}} L_{\delta_{t_p}}] \quad (2.37)$$

$$\mathbf{C}_{nr_{\delta_t}} = [-S_{\theta_{e_1}} L_{\delta_{t_1}} \quad -S_{\theta_{e_2}} L_{\delta_{t_2}} \quad \dots \quad -S_{\theta_{e_p}} L_{\delta_{t_p}}] \quad (2.38)$$

the vectors of transformations for the engines.

Writing the total moments acting on the aircraft due to the engines, the following equations result:

$$L_B^T = \mathbf{C}_{l_{\delta_t}} \boldsymbol{\delta}_t \quad (2.39)$$

$$M_B^T = \mathbf{C}_{m_{\delta_t}} \boldsymbol{\delta}_t \quad (2.40)$$

$$N_B^T = \mathbf{C}_{n_{\delta_t}} \boldsymbol{\delta}_t \quad (2.41)$$

with

$$\mathbf{C}_{l_{\delta_t}} = \mathbf{C}_{z_{\delta_t}} \mathbf{Y}_e + \mathbf{C}_{lr_{\delta_t}} \quad (2.42)$$

$$\mathbf{C}_{m_{\delta_t}} = \mathbf{C}_{z_{\delta_t}} \mathbf{X}_e + \mathbf{C}_{x_{\delta_t}} \mathbf{Z}_e \quad (2.43)$$

$$\mathbf{C}_{n_{\delta_t}} = \mathbf{C}_{x_{\delta_t}} \mathbf{Y}_e + \mathbf{C}_{nr_{\delta_t}} \quad (2.44)$$

$$(2.45)$$

The complete forces and moments caused by the engines and their rotor effects are now described above in body axes. These are now transformed into wind axes in order to maintain consistency. The wind axes are rotated about the body axes by the angle of attack, α , and the angle of side-slip, β , so that the transformation from body axes to wind axes is:

$$\begin{bmatrix} X^T \\ Y^T \\ Z^T \end{bmatrix} = DCM^{WB} \begin{bmatrix} X_B^T \\ Y_B^T \\ Z_B^T \end{bmatrix} \quad (2.46)$$

If it is assumed that the aircraft will be operating with a small angle of attack, bank angle and side-slip angle, a small-angle assumption can be made and the linearised equations describing the forces are then:

$$X^T = X_B^T + Y_B^T \beta + Z_B^T \alpha \quad (2.47)$$

$$Y^T = -X_B^T \beta + Y_B^T - Z_B^T \beta \alpha \quad (2.48)$$

$$Z^T = -X_B^T \alpha + Z_B^T \quad (2.49)$$

The transformation of moments from body axes into wind axes is then given by:

$$\begin{bmatrix} L^T \\ M^T \\ N^T \end{bmatrix} = DCM^{WB} \begin{bmatrix} L_B^T \\ M_B^T \\ N_B^T \end{bmatrix} \quad (2.50)$$

The small angle assumption is applied here as well so that the linear equations describing the total moments due to the engines are:

$$L^T = L_B^T + M_B^T \alpha + N_B^T \alpha \quad (2.51)$$

$$M^T = -L_B^T \beta + M_B^T - N_B^T \beta \alpha \quad (2.52)$$

$$N^T = -L_B^T \alpha + N_B^T \quad (2.53)$$

Gravity

Finally, the effects of gravity on the forces and moments acting on the aircraft are considered. Gravity forms part of the force and moment equations used to trim the aircraft and the control allocation system needs to know the trim biases of the actuators in order

to generate mixing vectors that do not saturate the actuators. The effect of gravity in inertial coordinates is given by:

$$F_I^G = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.54)$$

with gravity acting in the inertial Z_I -axis only.

The effects of gravity on the forces and moments acting on the aircraft can now be described by transforming the gravity vector from inertial to wind axes. Since the wind-axes being used have their origin at the centre of gravity of the aircraft, gravity has no effect on the moments of the aircraft so that:

$$L^G = M^G = N^G = 0 \quad (2.55)$$

This leaves only the forces acting on the aircraft due to gravity. The gravity vector can now be transformed into wind axes using the DCM transformation matrix. As the aircraft is trimmed in wind axes and it is assumed that the aircraft is flying parallel to the inertial $X_I - Y_I$ plane, the gravity vector in wind axes will merely act along the Z_W axis, transformed through the bank angle ϕ and along the Y_W axis, also transformed through the bank angle ϕ . Using the DCM:

$$\begin{bmatrix} X^G \\ Y^G \\ Z^G \end{bmatrix} = DCM^{WI} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \quad (2.56)$$

results in the following equations describing the forces acting on the aircraft due to gravity:

$$X^G = 0 \quad (2.57)$$

$$Y^G = \sin \phi mg \quad (2.58)$$

$$Z^G = \cos \phi mg \quad (2.59)$$

2.1.5 Expanded forces and moments

Now that the forces and moments produced by aerodynamic, power and gravitational influences on the aircraft have been defined, equations (2.2) to (2.7) can be re-written in their fully expanded forms in wind axes:

$$X = qS \left(-C_{D_0} - \frac{(C_{l_0} + C_{l_\beta}\beta + C_{l_\delta}\delta)^2}{\pi Ae} \right) + C_{x_{\delta_T}} \delta_t + C_{z_{\delta_T}} \delta_t \alpha \quad (2.60)$$

$$Y = qS (C_{y_0} + C_{y_\beta}\beta + C_{y_\delta}\delta) - C_{x_{\delta_T}} \delta_t \beta - C_{z_{\delta_T}} \delta_t \beta \alpha + \phi mg \quad (2.61)$$

$$Z = qS (-C_{L_0} - C_{L_\alpha}\alpha - C_{L_\delta}\delta) - C_{x_{\delta_T}} \delta_t \alpha + C_{z_{\delta_T}} \delta_t + mg \quad (2.62)$$

$$L = qSb (C_{l_0} + C_{l_\beta}\beta + C_{l_\delta}\delta) + C_{l_{\delta_t}} \delta_t + C_{m_{\delta_t}} \delta_t \beta + C_{n_{\delta_t}} \delta_t \alpha \quad (2.63)$$

$$M = qS\bar{c} (C_{m_0} + C_{m_\alpha}\alpha + C_{m_\delta}\delta) - C_{l_{\delta_t}} \delta_t \beta + C_{m_{\delta_t}} \delta_t - C_{n_{\delta_t}} \delta_t \beta \alpha \quad (2.64)$$

$$N = qSb (C_{n_0} + C_{n_\beta}\beta + C_{n_\delta}\delta) - C_{l_{\delta_t}} \delta_t \alpha + C_{n_{\delta_t}} \delta_t \quad (2.65)$$

2.2 Conclusion

The linearised equations describing the forces and moments acting on the aircraft are now available using equations (2.60) to (2.65). These equations are suitable for trimming the aircraft for horizontal flight in inertial space and for describing the static forces and moments that result from aerodynamic actuator deflections as well as engine thrust and rotor moments. These equations form the basis from which the linear control allocation formulation is developed in the next chapter.

Chapter 3

Problem formulation

This chapter describes the process of formulating the control allocation problem using the force and moment equations developed in chapter 2. The introduction below describes the differences between the proposed problem formulation and those mentioned in chapter 1. Next, some definitions crucial to the control allocation process that is to be developed are given and thereafter the full 6-DOF problem formulation is developed. Finally, the resulting goals and constraints are briefly summarised.

3.1 Introduction

Given the desired properties for this control allocation formulation as discussed in section 1.3, the problem formulation developed here has a number of distinguishing features. These differences from standard control allocation formulations are largely due to the architecture of the FTC system with which the system has to interface.

As mentioned in section 1.3, the large majority of control allocation systems are implemented within the control loop of the aircraft. This means that these control allocation systems find solutions to the control allocation problem for each discrete time-step of the control process. The main drawbacks of this method of control allocation are as follows:

- Firstly, the control allocation problem must be solved very quickly. A number of methods that can achieve this have been developed and are discussed in section 1.1.3 on solution methods. A side-effect of this time constraint is that in order to find solutions quickly enough, the control allocation problem is often limited to the very basic formulation where adverse forces and secondary goals are ignored.
- Secondly, the control allocation system and the control system interact with one another since both are constantly in flux. This interaction can have negative ef-

fects on performance and stability. For example, the control system does not know before-hand whether the moments it demands are achievable since the control allocation to meet those demands is generated after the command is given. The interactions between the control system and the control allocation system typically become problematic where actuator saturation occurs.

Another method of implementing the control allocation system is the semi-static approach used in this thesis. In this approach, a number of mixing vectors are defined that mix virtual commands from the controller to physical actuators on the aircraft. A fixed mapping is then used to generate the desired forces and moments until the FTC supervisor determines that this mapping needs to change, typically due to some failure. The supervisor system then requests a new set of mixing vectors to restore the aircraft to a more optimal state. It should be noted that optimal in this case can be defined as the state that minimises the difference in performance between the healthy aircraft and the damaged aircraft. This method relaxes the time constraints and minimises the interactions between the control system and the control allocation system. This method can also provide the FTC system with performance parameters when allocation is completed so that the control systems can be reconfigured if necessary.

3.2 Virtual actuators and control mixing vectors

As has been mentioned, the control allocation system fits between the aircraft control system and the physical aircraft. The control system makes use of virtual actuator commands in order to control the aircraft and these virtual commands are mapped into physical actuator deflections via the control allocation system. The various virtual commands and control mixing vectors required are described here.

3.2.1 Virtual Actuators

The control system makes use of four virtual actuators in order to control the aircraft. These virtual actuators are listed below along with their virtual commands:

- Virtual roll actuator, with its command δ_A
- Virtual pitch actuator, with its command δ_E
- Virtual yaw actuator, with its command δ_R

- Virtual thrust actuator, with its command δ_T

The control system then issues commands to the virtual actuator by issuing a command: δ_A , δ_E , δ_R or δ_T . In the case of a fault or failure, the nominal mappings from virtual actuators to physical actuators will need to change. Control mixing and biasing vectors are defined in the next section for this purpose.

3.2.2 Control mixing and biasing vectors

The proposed control allocation procedure is based around mixing vectors. Given a virtual actuator control signal δ_A , δ_E , δ_R or δ_T from the inner-loop controller, the aircraft actuators are controlled by mixing these signals to the appropriate actuators through these mixing vectors. The command vector sent to the physical actuators is then constructed as follows:

$$\delta = \begin{bmatrix} \mathbf{T}_{AA} & \mathbf{T}_{AE} & \mathbf{T}_{AR} & \mathbf{T}_{AT} \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \\ \delta_T \end{bmatrix} + \mathbf{T}_{AB} \quad (3.1)$$

and

$$\delta_t = \begin{bmatrix} \mathbf{T}_{TA} & \mathbf{T}_{TE} & \mathbf{T}_{TR} & \mathbf{T}_{TT} \end{bmatrix} \begin{bmatrix} \delta_A \\ \delta_E \\ \delta_R \\ \delta_T \end{bmatrix} + \mathbf{T}_{TB} \quad (3.2)$$

where $\mathbf{T}_{A(.)}$ are n -length column vectors containing the gains for all of the individual aerodynamic actuators and $\mathbf{T}_{T(.)}$ are p -length column vectors containing the gains for all of the engines. \mathbf{T}_{AB} and \mathbf{T}_{TB} are n - and p -length column vectors used internally by the control allocation system to bias the actuators and are used to ensure that position constraints are not violated. These mixing vectors allow a linear mixing from virtual actuator commands to physical actuator commands.

It should be kept in mind that not all aircraft will use all of the mixing vectors defined above. The mixing vectors for the engines are especially dependent on the moment generating abilities provided by the physical placement of the engines. A thrust mixing vector for each virtual command is however described to keep the problem formulation general.

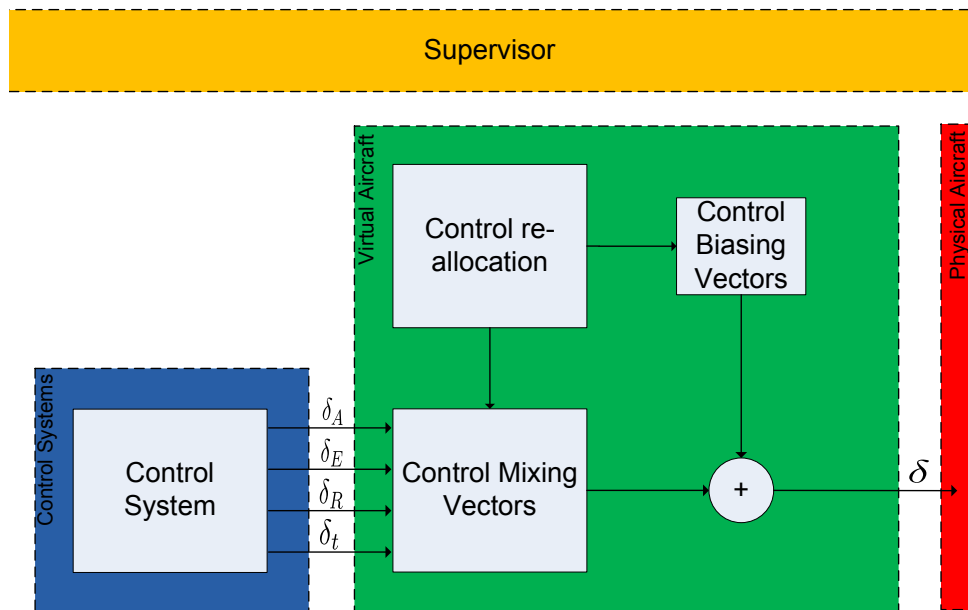


Figure 3.1: The control allocation system architecture

3.3 The Control allocation system

Having defined the control mixing and biasing vectors as well as the virtual actuators of the aircraft, the control allocation architecture can be defined. The system will work as shown in figure 3.1. The control system generates the four virtual commands defined above and these commands are fed into the control mixing block where the control mixing vectors mix each command to individual actuators. These commands are then added to the biasing vector to form the final control vector sent to the physical actuators.

The mixing and biasing vectors are updated by the control allocation system when triggered by the supervisor. The required aircraft parameters are provided to the control allocation system by the system I.D. system.

The control allocation system must then generate the control mixing vectors that allow the aircraft to follow some minimum magnitude of virtual actuator command without the command vector, δ , saturating the physical actuators. The biasing vector determines the point from which the actuators are deflected and as such, the maximum amount of deflection allowed to an actuator depends on its initial bias.

The control allocation system must perform the following tasks:

- Find the bias vectors for the actuators.

- Attempt to match the requested virtual actuator performance while minimising adverse effects.
- Ensure that some minimum level of virtual actuator deflection is available without saturating the physical actuators.

In the next section, each of these tasks are described in more detail and the full 6-DOF problem formulation is developed.

3.4 The full 6-DOF problem formulation

The full 6-DOF problem formulation is now developed taking into account the effects of the aerodynamic actuators as well as the engines. The equations necessary to trim a damaged aircraft for asymmetric flight are developed, followed by a description of the force and moment error minimisation. The minimisation of the adverse effects for each of the virtual commands is discussed thereafter. Some considerations with respect to actuator dynamics are mentioned and minimising control usage is briefly discussed. Finally, the constraints on the actuators are discussed in more detail.

3.4.1 Trimming

The occurrence of a failure may result in biases being introduced on the forces and moments acting on the aircraft. In order to maintain a steady flight condition, these biases need to be removed so that the net forces and moments acting on the aircraft are once again zero. The act of removing the biases is essentially the calculation of the new trim conditions of the aircraft after a failure has occurred.

The process of trimming the aircraft is the process of finding the control settings that reduce the total forces and moments acting on the aircraft to zero for a specific flight condition. The result is an aircraft with some angle of attack, α , angle of side-slip, β , and bank angle, ϕ , as well as specific actuator position and throttle settings. In most conventional cases with no external disturbances, a healthy aircraft is trimmed for straight and level flight with zero bank angle and zero side-slip angle. This simplifies the trimming process to calculating the elevator and throttle settings that bring the net longitudinal forces and moments to zero. In the case where failures have occurred, the angle of attack, bank angle and side-slip angle will all need to be taken into account. Additionally, the aircraft may need to use actuators other than the elevator and engines to trim the aircraft. Therefore, the full set of actuators must be considered.

The aircraft model developed in chapter 2 was for a general aircraft. In order to develop the trim equations, some assumptions are made about the configuration of the aircraft being considered. These assumptions and their impact on the resulting equations are listed here:

- It is assumed that the thrust produced by the engines acts primarily along the X_B -axis, possibly with some small component acting in the Z_B -axis due to an engine angle offset θ_E . Consequently, the forces due to thrust acting in the Z_B -axes will be negligible in most cases.
- It is assumed that the engine configuration is such that engines in the healthy or failed aircraft only affect the pitch and yaw moments of the aircraft, not the roll moment. Additionally, the roll moment caused by the rotation of the props is assumed to be negligible.

These assumptions limit the configuration of aircraft to a conventional aircraft with engines that have only small offsets in θ_E angles. Positional offsets along all three axes are allowed. This configuration is still general enough to allow the control allocation system to work with most conventional aircraft and will only need to be adjusted if unconventional aircraft need to be accommodated, as will be demonstrated in chapter 7 where a ducted-fan UAV is considered.

The equations used to trim the aircraft are discussed below. When trimming an aircraft, these equations are normally used to directly find the trim actuator positions that will result in zero forces and moments acting on the aircraft. In this case, the trim equations are developed so that the trim angles of attack, side-slip and bank angle are given in terms of actuator positions. Similarly, the remaining equations are left as equality constraints in terms of actuator positions. This is due to the fact that these equations are added as goals and constraints to the optimisation engine which calculates the actuator trim positions that will satisfy these equations as well as various other constraints that are placed on actuator usage.

Longitudinal trim equations

The equations describing the longitudinal forces and moments acting on the aircraft are given by equations (2.60), (2.62) and (2.64) in the modelling section. These equations are re-written here with the forces and moments set to zero:

$$(-q_T S C_{D_T}) + \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TB} + \mathbf{C}_{z_{\delta_T}} \mathbf{T}_{TB} \alpha_T = 0 \quad (3.3)$$

$$-q_T S (C_{L_0} + C_{L_\alpha} \alpha_T + \mathbf{C}_{L_\delta} \mathbf{T}_{AB}) - \mathbf{C}_{x_{\delta_t}} \mathbf{T}_{TB} \alpha_T + \mathbf{C}_{z_{\delta_t}} \mathbf{T}_{TB} + mg = 0 \quad (3.4)$$

$$q_T S \bar{c} (C_{m_0} + C_{m_\alpha} \alpha_T + \mathbf{C}_{m_\delta} \mathbf{T}_{AB}) - \mathbf{C}_{l_{\delta_t}} \mathbf{T}_{TB} \beta_T + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} - \mathbf{C}_{n_{\delta_t}} \mathbf{T}_{TB} \beta_T \alpha_T = 0 \quad (3.5)$$

Given the aforementioned assumptions with respect to the thrust effects on the aircraft, these equations can be simplified by removing terms that will have a negligible effect. Equation (3.3) becomes:

$$(-q_T S C_{D_T}) + \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (3.6)$$

With the small thrust term in the Z_B -axis multiplied by the small angle α_T so that this term can be ignored.

Equation (3.4) stays the same and equation (3.5) becomes:

$$q_T S \bar{c} (C_{m_0} + C_{m_\alpha} \alpha_T + \mathbf{C}_{m_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} = 0 \quad (3.7)$$

Equation (3.6) is added as a constraint to the optimisation, equation (3.4) is used to write the trim angle of attack, α_T , as a function of the actuator positions and equation (3.7) is added as a constraint to ensure zero pitching moment.

Writing the trim angle of attack α_T as a function of the aerodynamic and thrust actuator positions is now accomplished:

$$\alpha_T = \frac{q_T S C_{L_0} + q_T S \mathbf{C}_{L_\delta} \mathbf{T}_{AB} - \mathbf{C}_{z_{\delta_t}} \mathbf{T}_{TB} - mg}{-q_T S C_{L_\alpha} - \mathbf{C}_{x_{\delta_t}} \mathbf{T}_{TB}} \quad (3.8)$$

Lateral trim equations

Similarly to the longitudinal case above, the equations that describe the lateral forces and moments acting on the aircraft are given by equations (2.61), (2.63) and (2.65) in the modelling section. These equations are re-written here with the forces and moments set to zero:

$$q_T S (C_{y_0} + C_{y_\beta} \beta_T + \mathbf{C}_{y_\delta} \mathbf{T}_{AB}) - \mathbf{C}_{x_{\delta_t}} \mathbf{T}_{TB} \beta_T - \mathbf{C}_{z_{\delta_t}} \mathbf{T}_{TB} \beta_T \alpha_T + mg \phi_T = 0 \quad (3.9)$$

$$q_T S b (C_{l_0} + C_{l_\beta} \beta_T + \mathbf{C}_{l_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{l_{\delta_t}} \mathbf{T}_{TB} + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} \beta_T + \mathbf{C}_{n_{\delta_t}} \mathbf{T}_{TB} \alpha_T = 0 \quad (3.10)$$

$$q_T S b (C_{n_0} + C_{n_\beta} \beta_T + \mathbf{C}_{n_\delta} \mathbf{T}_{AB}) - \mathbf{C}_{l_{\delta_T}} \mathbf{T}_{TB} \alpha_T + \mathbf{C}_{n_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (3.11)$$

Once again, these equations are simplified by removing negligible terms due to thrust. Equation (3.9) becomes:

$$q_T S (C_{y_0} + C_{y_\beta} \beta_T + C_{y_\delta} \mathbf{T}_{AB}) - C_{x_{\delta_t}} \mathbf{T}_{TB} \beta_T + mg \phi_T = 0 \quad (3.12)$$

Equation (3.10) becomes:

$$q_T S b (C_{l_0} + C_{l_\beta} \beta_T + C_{l_\delta} \mathbf{T}_{AB}) + C_{m_{\delta_t}} \mathbf{T}_{TB} \beta_T + C_{n_{\delta_t}} \mathbf{T}_{TB} \alpha_T = 0 \quad (3.13)$$

and equation (3.11) becomes:

$$q_T S b (C_{n_0} + C_{n_\beta} \beta_T + C_{n_\delta} \mathbf{T}_{AB}) + C_{n_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (3.14)$$

Now, from (3.13), β_T is:

$$\beta_T = \frac{-q_T S b C_{l_0} - q_T S b C_{l_\delta} \mathbf{T}_{AB} - C_{n_{\delta_t}} \mathbf{T}_{TB} \alpha_T}{q_T S b C_{l_\beta} + C_{m_{\delta_t}} \mathbf{T}_{TB}} \quad (3.15)$$

and, from (3.12), ϕ_T is:

$$\phi_T = \frac{-q_T S (C_{y_0} + C_{y_\beta} \beta_T + C_{y_\delta} \mathbf{T}_{AB}) + C_{x_{\delta_t}} \mathbf{T}_{TB} \beta_T}{mg} \quad (3.16)$$

and equation (3.14) is added as a constraint to ensure zero yawing moment.

The equations for α_T , β_T and ϕ_T are now available as functions of the bias vectors \mathbf{T}_{AB} and \mathbf{T}_{TB} and there are three equations that act as constraints on the system. There are two equations that form goals, minimising β_T and ϕ_T . The equations required to trim the aircraft for both a healthy and damaged configuration are now available. These equations permit the trimming of the aircraft to be accomplished as a function of the bias vectors so that numerical optimisation methods can be used.

Equations (3.15) and (3.16) describe the angle of side-slip and bank angle of the aircraft. For a healthy aircraft, these angles will typically be zero if external disturbances are not present. In the case of damage to the aircraft, it may not be possible to trim the aircraft while keeping these angles zero. In this case, it is still desired that these angles be kept small. Additionally, however, [24] shows that in the case of asymmetric flight, it is possible to trim the aircraft for either a non-zero bank angle or non-zero side-slip angle. [6] extends this to include any valid combination of bank angle and side-slip angle. To illustrate this, an example of asymmetric flight given by [6] is considered below.

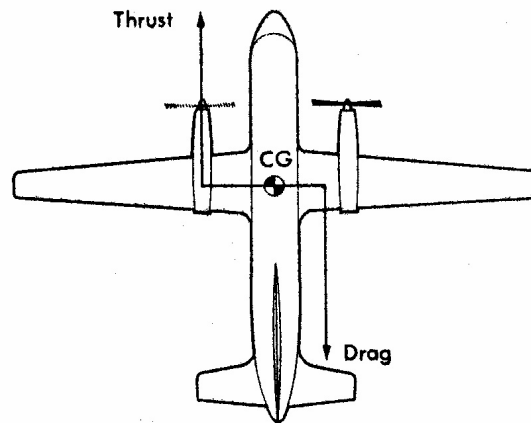


Figure 3.2: Asymmetric flight configuration [6]

Asymmetric flight

Consider the case shown in figure 3.2 below where a twin-engined aircraft has experienced a failure in one engine.

The thrust produced by the single available engine will produce a significant yaw moment bias acting on the aircraft and trimming the aircraft will now result in an asymmetric flight condition. There are three flight conditions that can be used to trim the aircraft in a situation such as the one being considered. These are: zero bank angle, zero side-slip and a combination of bank angle and side-slip. Each of these configurations are discussed below.

Zero bank angle

If a large adverse yaw moment is acting on the aircraft and it is desired that the aircraft be trimmed with $\phi = 0$, then the aircraft has to be trimmed with some non-zero side-slip angle β , as shown in figure 3.3.

In this configuration, the rudder is used to cancel out the yaw moment caused by the engine and the side-slip angle cancels the side-force caused by the rudder. The ailerons are used to counter the roll moment cause by side-slip. The benefit of this configuration is that the aircraft remains level with zero lateral specific acceleration (LSA) and the full range of bank angles remain available to the aircraft.

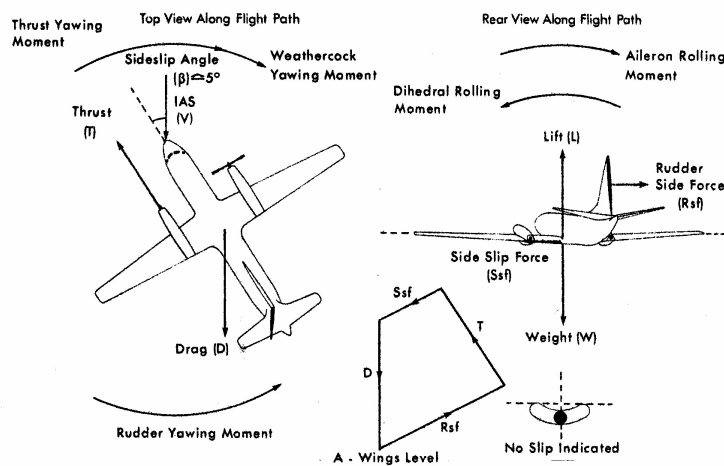


Figure 3.3: Asymmetric flight with zero bank angle [6]

Zero side-slip

If it is desired that the aircraft is trimmed with a zero side-slip angle, $\beta = 0$, then the aircraft may be trimmed at some non-zero bank angle, ϕ , as shown in figure 3.4.

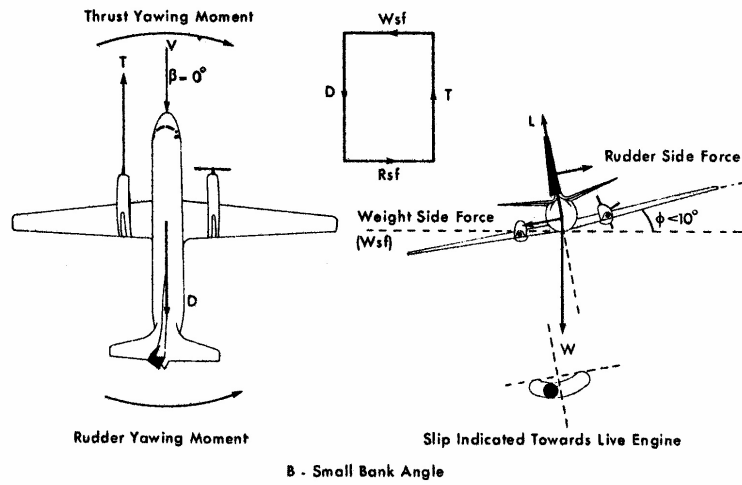
In this configuration, the rudder is once again used to cancel the adverse yaw moment caused by differential thrust but instead of using side-slip to cancel the side-force induced by the rudder, the aircraft is banked and the gravity vector is used to counter this force, resulting in a non-zero LSA.

Combined bank and side-slip

Either of the two configurations given above may result in a bank angle or side-slip angle that is larger than desired. In such a case, it may be beneficial to use a combination of bank angle and side-slip, as shown in figure 3.5.

In this case, it is possible to trim the aircraft for asymmetric flight without using the rudder as the yawing moment caused by the thrust can be countered with the weathercock yawing moment and the bank angle is used to counter the side-force due to side-slip. The ailerons are also used to counter the roll-moment due to side-slip. This configuration typically results in a large bank angle [6] and therefore a significant LSA.

It is clear from the above example that the side-slip angle and bank angle of the aircraft can be traded off when the aircraft is trimmed for an asymmetric flight condition. This trade-off will be handled when setting up the final goals for the control allocation system.



Figs 2A and 2B Forces Acting in Asymmetric Flight

Figure 3.4: Asymmetric flight with zero side-slip angle [6]

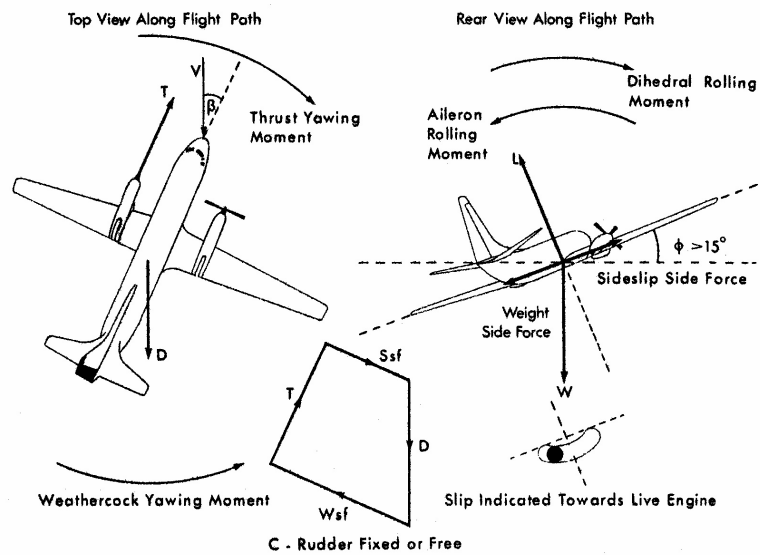


Fig 2C Forces Acting in Asymmetric Flight

Figure 3.5: Asymmetric flight with a combination of bank angle and side-slip angle [6]

3.4.2 Minimising force and moment errors

When a controller generates a virtual roll, pitch, yaw or thrust command, this command is mixed to the appropriate actuators to generate the desired force or moment. For a healthy aircraft, a command in any of these virtual actuators will correctly generate the desired force or moment. The goal of force and moment optimisation is generate the mixing vectors that will allow the virtual actuators to generate forces or moments that are as close as possible to the desired forces or moments. The desired forces and moments for the four virtual actuators are represented by L^d , M^d , N^d and T^d . These are scalar values that can be selected by the user and could, as an example, be based on the known performance of some aircraft configuration although this is not necessary.

The force and moment equations from chapter 2 are now used to describe the force and moment error minimisation procedure. In the equations below, it is assumed that the aircraft has been trimmed so that the bias terms can be ignored, leaving only the forces and moments generated by the aerodynamic actuators and the engines. The assumptions made with respect to forces and moments generated by the engines in section 3.4.1 are slightly different from the assumptions applied here. In this section, the forces and moments generated by the engines that would form a useful and reliable part of the virtual actuator are included. Some of the terms included will be zero for the aircraft configuration described in section 3.4.1. These terms are included here so that this part of the control allocation system can be used without alteration for different aircraft configurations so that only the trim equations need to be updated if a different aircraft type is considered. The primary forces and moments generated by the engines in each axis are therefore included regardless of whether they apply to the aircraft configuration stipulated in section 3.4.1. The terms resulting from small angle deviations between body and wind axes are not considered reliable actuators and are therefore ignored. The results are the following equations describing the effectiveness of the virtual actuators:

$$L^r = qSb(C_{l_\delta} \mathbf{T}_{AA}) + C_{l_{\delta_t}} \mathbf{T}_{TA} \quad (3.17)$$

$$M^r = qS\bar{c}(C_{m_\delta} \mathbf{T}_{AE}) + C_{m_{\delta_t}} \mathbf{T}_{TE} \quad (3.18)$$

$$N^r = qSb(C_{n_\delta} \mathbf{T}_{AR}) + C_{n_{\delta_t}} \mathbf{T}_{TR} \quad (3.19)$$

$$T^r = qS(C_{t_\delta} \mathbf{T}_{AT}) + C_{t_{\delta_t}} \mathbf{T}_{TT} \quad (3.20)$$

where \mathbf{T}_{AA} , \mathbf{T}_{TA} , \mathbf{T}_{AE} , \mathbf{T}_{TE} , \mathbf{T}_{AR} , \mathbf{T}_{TR} , \mathbf{T}_{AT} and \mathbf{T}_{TT} are the mixing vectors that must be solved for according to the following goals:

$$\min \{L^r - L^d\} \quad (3.21)$$

$$\min \{M^r - M^d\} \quad (3.22)$$

$$\min \{N^r - N^d\} \quad (3.23)$$

$$\min \{T^r - T^d\} \quad (3.24)$$

It should be noted that the virtual thrust actuator is represented by T with the actuator effectiveness vectors \mathbf{C}_{t_δ} and $\mathbf{C}_{t_{\delta_T}}$ describing the effectiveness of the actuators for a general thrust axis. For a specific aircraft, these terms will be replaced by the terms that represent the primary thrust axis of the aircraft. For example, for a normal fixed-wing aircraft with the primary thrust axis being the X -axis, the following equation will be used for the virtual thrust actuator:

$$X = \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TT} \quad (3.25)$$

where \mathbf{C}_{t_δ} has been replaced by \mathbf{C}_{x_δ} , which is a zero vector when actuator drag is ignored, and $\mathbf{C}_{t_{\delta_t}}$ has been replaced by $\mathbf{C}_{x_{\delta_t}}$. This generalisation of the equations will make it possible to substitute a different primary thrust axis if an unconventional aircraft is to be considered.

These are the goals of the control allocation system with respect to the primary forces and moments produced by the virtual actuators. In terms of the application of these goals within the FTC system, depending on the success of minimising the differences between desired and achieved performance, the FTC system may or may not have to reconfigure the control systems of the aircraft.

3.4.3 Minimising adverse effects

Each actuator on an aircraft is designed to perform a specific task and its placement on the aircraft gives it the ability to perform that task. Unfortunately, while deflecting an actuator will generate the desired force or moment, there are also other unwanted effects due to the deflection. These are called adverse effects and aircraft control systems are usually designed taking these into account. These adverse effects are normally quite small when compared to the forces and moments produced in the desired axes. However, when the mixing from virtual actuators to physical actuators is changed by the control allocation system, these adverse effects will be changed as well and the use of unconventional actuators may result in larger than normal adverse effects. Minimising these adverse effects is

desirable in order to minimise the effect on the control system performance. The adverse effects most commonly considered are adverse yaw caused by aileron deflection, adverse roll caused by rudder deflection and adverse lift. During the process of minimising the adverse effects, the full adverse effects of the aircraft will be taken into account, not just those that normally dominate. The simplifying assumptions made for force and moment error minimisation with respect to engine effects will be used here as well and the effects of thrust and rotor moments that are multiplied by the small angles α_T and β_T will be ignored.

Adverse effects due to a roll command

For a roll command, the adverse forces and moments to consider are: adverse pitch, adverse yaw, adverse side-force, adverse lift and adverse axial force. Each of these effects can be described by substituting the aerodynamic and thrust command vectors for a roll command into the respective equations describing each degree of freedom:

$$M_A^r = qS\bar{c}(C_{m_\delta}T_{AA}) + C_{m_{\delta_t}}T_{TA} \quad (3.26)$$

$$N_A^r = qSb(C_{n_\delta}T_{AA}) + C_{n_{\delta_t}}T_{TA} \quad (3.27)$$

$$X_A^r = qS(C_{x_\delta}T_{AA}) + C_{x_{\delta_T}}T_{TA} \quad (3.28)$$

$$Y_A^r = qS(C_{y_\delta}T_{AA}) + C_{y_{\delta_T}}T_{TA} \quad (3.29)$$

$$Z_A^r = qS(-C_{L_\delta}T_{AA}) + C_{z_{\delta_T}}T_{TA} \quad (3.30)$$

These adverse effects must be minimised, resulting in the following goals:

$$\min \{M_A^r\} \quad (3.31)$$

$$\min \{N_A^r\} \quad (3.32)$$

$$\min \{Y_A^r\} \quad (3.33)$$

$$\min \{Z_A^r\} \quad (3.34)$$

$$\min \{X_A^r\} \quad (3.35)$$

The adverse effects due to virtual pitch, yaw and thrust commands can be described similarly by substituting the T_{AE} and T_{TE} vectors, the T_{AR} and T_{TR} vectors and T_{AT} and T_{TT} vectors into the appropriate equations so that each virtual actuator command has five equations describing the adverse effects of that actuator.

3.4.4 Actuator dynamics

Although actuator dynamics are not being considered in the control allocation system at this point, the following should be kept in mind. The aerodynamic actuators and engines are used together in the above equations to generate the desired forces and moments and the dynamics of these actuators are not being taken into account by the control allocation system. The dynamics of the engines will typically mean that the aerodynamic actuators and engines have differing bandwidths. The difference in actuator bandwidth must be taken into account somewhere to ensure correct control system performance. However, for the initial formulation, it is assumed that these dynamics do not affect the results.

The effects of this assumption are considered to be minimal for the majority of the failure cases that will be considered. Nevertheless, two examples of cases where this may become problematic are given here.

Example 1 Consider an aircraft with twin engines, one on each wing, that has lost the majority of its yawing ability through the loss of its rudders. In order to generate the desired yaw moment, the control allocation algorithm is likely to resort to using differential thrust in order to generate the yaw moment required. The resulting virtual actuator will generate the desired yaw moment but the response of the aircraft to a yaw command will be delayed by the dynamics of the engines. If this delay is not taken into account by the control system, the response of the aircraft will not be as desired.

Example 2 Consider the aircraft described above in a condition where one engine has failed. Issuing a virtual thrust command to the nominal actuators will now result in an adverse yaw moment. In order to minimise this adverse error, the control allocation system may use aerodynamic actuators to remove the adverse yaw caused by a thrust command. The response of the aerodynamic actuators is however much faster than the response of the engines. If the difference in actuator response-time is not taken into account by the control system, the aerodynamic actuators will respond before the engines have produced the adverse yaw moment.

If the basic control allocation formulation being described in this chapter is to be used in practice, these shortcomings will need to be addressed by the control system. To help address this issue, an extension to the control allocation problem formulation is described in chapter 7 that will distinguish between fast and slow actuators by generating two sets of mixing vectors. For more details on this extension and its practical results, refer chapter 7.

3.4.5 Control usage

As seen in section 1.1.2, the mixed optimisation problem presented has a control minimisation term. This term can be included if desired although it is expected that there will be a trade-off between control usage and error minimisation. The idea is to minimise the values of the mixing and biasing vectors:

$$\min \{ \boldsymbol{x} \} \tag{3.36}$$

$$\tag{3.37}$$

where \boldsymbol{x} is a vector made up of all of the mixing and biasing vectors defined in section 3.2.2. This term will compete with the primary goal of force and moment error minimisation, resulting in virtual actuators that use smaller actuator deflections but also produce smaller forces and moments.

This may be advantageous if the forces and moments achievable are secondary to any of the following:

- Control effort and thus power requirements are reduced.
- Induced drag will be reduced as actuator deflections will be smaller.
- Control-load is expected to be reduced.

As mentioned, these advantages need to be weighed against the primary goal of the control re-allocation system: Reducing the errors between desired and achievable performance of the virtual actuators. Minimising control usage is therefore a secondary goal.

3.4.6 Constraints

All of the above goals must be met while ensuring that the control allocation algorithm does not violate any constraints placed on the aircraft. This includes the actuator position constraints. The aerodynamic actuators each have a maximum and minimum deflection angle that cannot be exceeded while the engines have a maximum and minimum thrust level that cannot be exceeded. Equations 3.1 and 3.2 show how the actuator command vector is constructed from the actuator mixing and biasing vectors. For a virtual roll command, the result would be:

$$\boldsymbol{\delta} = \boldsymbol{T}_{AA}\boldsymbol{\delta}_A + \boldsymbol{T}_{AB} \tag{3.38}$$

The control allocation system should ensure that the position limits are not exceeded while ensuring that a certain level of actuation, δ_S , remains available. This results in the following set of inequality constraint equations:

$$|\mathbf{T}_{AA}^r \delta_S + \mathbf{T}_{AB}^r| \leq |\boldsymbol{\delta}_{max}| \quad (3.39)$$

$$|\mathbf{T}_{AE}^r \delta_S + \mathbf{T}_{AB}^r| \leq |\boldsymbol{\delta}_{max}| \quad (3.40)$$

$$|\mathbf{T}_{AR}^r \delta_S + \mathbf{T}_{AB}^r| \leq |\boldsymbol{\delta}_{max}| \quad (3.41)$$

$$|\mathbf{T}_{AT}^r \delta_S + \mathbf{T}_{AB}^r| \leq |\boldsymbol{\delta}_{max}| \quad (3.42)$$

where $\boldsymbol{\delta}_{max}$ is a vector containing the maximum deflections for each individual actuator. The pre-defined minimum level of actuation for the virtual actuators, δ_S , is included to ensure that the resulting combination of mixing and biasing vector allows the control system to maintain an acceptable level of control authority. If this term is excluded, it is conceivable that solutions will result where the aircraft is successfully trimmed and the virtual actuators perform within range of their nominal values but only for a very small virtual actuator command. Similarly for the throttle settings:

$$0 \leq \mathbf{T}_{TA}^r \delta_S + \mathbf{T}_{TB}^r \leq \boldsymbol{\delta}_{tmax} \quad (3.43)$$

$$0 \leq \mathbf{T}_{TE}^r \delta_S + \mathbf{T}_{TB}^r \leq \boldsymbol{\delta}_{tmax} \quad (3.44)$$

$$0 \leq \mathbf{T}_{TR}^r \delta_S + \mathbf{T}_{TB}^r \leq \boldsymbol{\delta}_{tmax} \quad (3.45)$$

$$0 \leq \mathbf{T}_{TT}^r \delta_S + \mathbf{T}_{TB}^r \leq \boldsymbol{\delta}_{tmax} \quad (3.46)$$

where $\boldsymbol{\delta}_t$ is once again a vector containing maximum thrust settings for all of the engines. These equations can be further generalised if necessary to take into account actuators with different minimum and maximum deflection angles as well as different minimum amounts of virtual actuator free-play so that equations 3.39 to 3.42 become:

$$\boldsymbol{\delta}_{min} \leq \mathbf{T}_{AA}^r \delta_{AS} + \mathbf{T}_{AB}^r \leq \boldsymbol{\delta}_{max} \quad (3.47)$$

$$\boldsymbol{\delta}_{min} \leq \mathbf{T}_{AE}^r \delta_{ES} + \mathbf{T}_{AB}^r \leq \boldsymbol{\delta}_{max} \quad (3.48)$$

$$\boldsymbol{\delta}_{min} \leq \mathbf{T}_{AR}^r \delta_{RS} + \mathbf{T}_{AB}^r \leq \boldsymbol{\delta}_{max} \quad (3.49)$$

$$\boldsymbol{\delta}_{min} \leq \mathbf{T}_{AT}^r \delta_{TS} + \mathbf{T}_{AB}^r \leq \boldsymbol{\delta}_{max} \quad (3.50)$$

and similarly, δ_S in equations 3.43 to 3.46 is replaced with δ_{AS} , δ_{ES} , δ_{RS} and δ_{TS} .

3.5 Summary of objectives and constraints

A brief summary of the equations that have been developed in this chapter to describe the goals and constraints for the control allocation system is given here.

3.5.1 Objectives

The objectives that have been identified will be used to formulate the cost function of the optimisation problem that will be used to solve the control allocation problem. There are objectives related to finding the actuator biases, minimising the force and moment errors, minimising adverse effects and minimising control usage.

Trim

The two trim goals are to minimise the trim side-slip and bank angles to ensure that the control allocation system does not calculate the mixing vectors based on large angles:

$$\min \{\beta_T\} \quad (3.51)$$

$$\min \{\phi_T\} \quad (3.52)$$

The ϕ_T term also ensures that the net side-force for trim is zero. The equation for ϕ_T is defined when the side-force equation is set to zero.

Forces and moments

The forces and moments generated by the virtual actuators after control allocation has taken place must be as close as possible to the desired forces and moments:

$$\min \{L^r - L^d\} \quad (3.53)$$

$$\min \{M^r - M^d\} \quad (3.54)$$

$$\min \{N^r - N^d\} \quad (3.55)$$

$$\min \{X^r - X^d\} \quad (3.56)$$

Adverse forces and moments

The adverse forces and moments produced by the virtual actuators must be minimised. The adverse forces and moments for the virtual roll actuator are once again used as an example:

$$\min \{M_A^r\} \quad (3.57)$$

$$\min \{N_A^r\} \quad (3.58)$$

$$\min \{Y_A^r\} \quad (3.59)$$

$$\min \{Z_A^r\} \quad (3.60)$$

$$\min \{X_A^r\} \quad (3.61)$$

There are another 15 such equations describing the minimisation of the adverse forces and moments for the virtual pitch, yaw and thrust actuators.

Control usage

An optional control usage minimisation goal is added:

$$\min \{\mathbf{x}\} \quad (3.62)$$

$$(3.63)$$

where \mathbf{x} is a vector made up of all of the mixing and biasing vectors defined in section 3.2.2.

3.5.2 Constraints

There are both equality and inequality constraints for the control allocation formulation. The equality constraints are used to find the actuator biases used to trim the aircraft. These biases are required so that the control allocation system can generate the required mixing vectors while ensuring that position constraints are not violated. The position constraints form the inequality constraints.

Equality constraints

The equality constraints ensure that the biasing vector reduces the net forces and moment acting on the aircraft to zero for aircraft trim. The six equations describing the 6-DOF force and moment equations are handled as follows: The net axial forces, net pitch moment and net yaw moment equations are added as constraints, shown below. The net force in the z-axis is encapsulated in the α_T term that forms part of the pitch moment equation and the net roll moment is encapsulated in the β_T term that forms part of the yaw moment equation.

$$(-q_T S C_{D_T}) + \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (3.64)$$

$$q_T S \bar{c} (C_{m_0} + C_{m_\alpha} \alpha_T + \mathbf{C}_{m_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} = 0 \quad (3.65)$$

$$q_T S b (C_{n_0} + C_{n_\beta} \beta_T + \mathbf{C}_{n_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{n_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (3.66)$$

Inequality constraints

The biasing vectors that are used to meet the trim goals and the equality constraints are then used to ensure that the resulting mixing vector and bias vector combinations do not violate the position constraints placed on the actuators, forming the following inequality constraints for the aerodynamic actuators:

$$\delta_{min} \leq \mathbf{T}_{AA}^r \delta_{AS} + \mathbf{T}_{AB}^r \leq \delta_{max} \quad (3.67)$$

$$\delta_{min} \leq \mathbf{T}_{AE}^r \delta_{ES} + \mathbf{T}_{AB}^r \leq \delta_{max} \quad (3.68)$$

$$\delta_{min} \leq \mathbf{T}_{AR}^r \delta_{RS} + \mathbf{T}_{AB}^r \leq \delta_{max} \quad (3.69)$$

$$\delta_{min} \leq \mathbf{T}_{AT}^r \delta_{TS} + \mathbf{T}_{AB}^r \leq \delta_{max} \quad (3.70)$$

and the following inequality constraints for the engines:

$$0 \leq \mathbf{T}_{TA}^r \delta_{AS} + \mathbf{T}_{TB}^r \leq \delta_{t_{max}} \quad (3.71)$$

$$0 \leq \mathbf{T}_{TE}^r \delta_{ES} + \mathbf{T}_{TB}^r \leq \delta_{t_{max}} \quad (3.72)$$

$$0 \leq \mathbf{T}_{TR}^r \delta_{RS} + \mathbf{T}_{TB}^r \leq \delta_{t_{max}} \quad (3.73)$$

$$0 \leq \mathbf{T}_{TT}^r \delta_{TS} + \mathbf{T}_{TB}^r \leq \delta_{t_{max}} \quad (3.74)$$

3.6 Conclusion

The control allocation problem is now described mathematically by equations 3.51 to 3.74. The resulting equations can be used to solve for the mixing vectors that will minimise the difference between the desired and achievable virtual actuator performance while minimising the adverse effects associated with these virtual actuators. The constraints ensure that this is achieved without violating the position constraints placed on the actuators. These equations must now be used to set up the optimisation problem so that the mixing vectors can be solved for. This is the subject of the next chapter.

Chapter 4

Optimisation Detail

4.1 Introduction

Figure 4.1 shows the typical process of correctly setting up an optimisation problem. The chapters up to this point have completed these steps so that the final stage of conditioning the problem formulation to fit into an existing optimisation method can be completed here.

The problem formulation provided in the previous chapter has multiple objectives and both equality and inequality constraints. The rest of this chapter will therefore focus on the following class of optimisation problem: multi-objective constrained optimisation. First, the general multi-objective optimisation problem is defined, followed by some definitions of optimality for multi-objective optimisation problems. Some considerations for solving such problems are presented and the choice of cost function as well as issues of normalisation are presented next. After this, a brief description is given of the optimisation methods that could apply to this problem with a brief discussion of which would be most

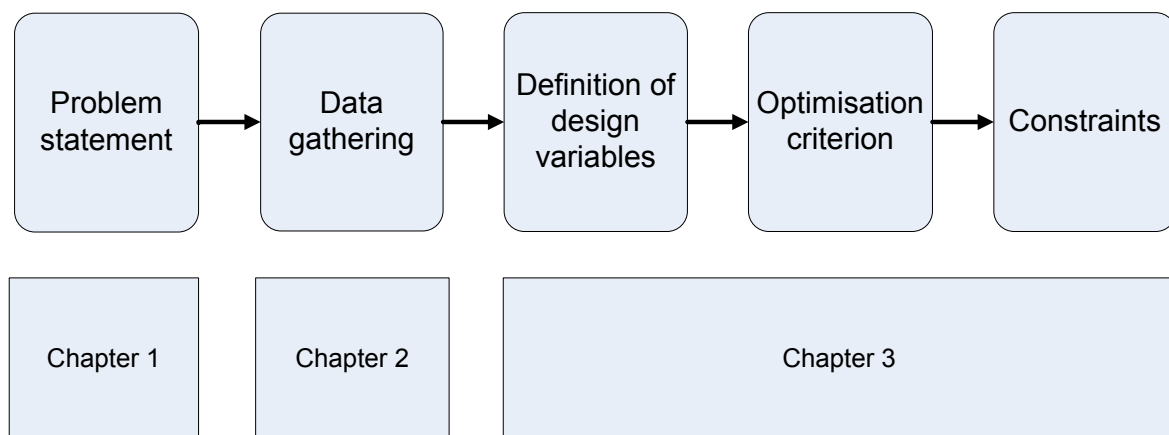


Figure 4.1: Optimisation problem setup overview

suitable. The chosen SQP optimisation method is then described and the mathematical development of the required cost function and constraints is given.

4.2 Multi-objective Optimisation

The general multi-objective optimisation problem formulation is given below. Some considerations for solving such problems are also discussed in this section and the choice of a cost function and normalisation methods are described.

4.2.1 General problem formulation

Optimisation involves maximising or minimising some objective function which is a function of the design variables of the system, often subject to constraints on the design variables. In the case of multi-objective optimisation, this objective function is made up of a number of sub-objectives. The general multi-objective optimisation problem can be described mathematically as follows, from [25]:

$$\text{minimise } f(x) = (f_1(x), f_2(x), \dots, f_k(x)) \quad (4.1)$$

subject to a set of equality constraints:

$$h_i(x) = 0; i = 1 \text{ to } p \quad (4.2)$$

and the set of inequality constraints:

$$g_j(x) \leq 0; j = 1 \text{ to } m \quad (4.3)$$

When solving the above problem, there is generally not a single global optimum solution and some definition of an optimal point is required. [26] gives two concepts that can be used to define optimal design points for a multi-objective optimisation problem. The first is the concept of Pareto optimality and it is defined as follows:

Pareto Optimality - *"A point \mathbf{x}^* in the feasible design space S is called Pareto optimal if there is no other point \mathbf{x} in the set S that reduces at least one objective function without increasing another one."* [25]

The second is the concept of a compromise solution. A compromise solution is defined as a solution that minimises the difference between a possible optimal design point and

a utopia point. The utopia point is not usually a feasible point and so a compromise is made by finding a point as close to the utopia point as possible. A utopia point is defined as follows:

Utopia point - $f(x)$ is a utopia point, f_i^0 , if and only if every objective function at the design point defined by x is at its minimum value.

Calculating or even approximating the utopia point can often be computationally expensive [26]. For this reason, utopia points are sometimes replaced by unattainable aspiration points, z , which are selected by the designer to approximate the utopia point.

4.2.2 Considerations when solving

Certain properties of the cost function and the constraints have an influence on what optimisation methods can be used to solve the problem. Additionally, some designer choices also affect what optimisation methods can be used. These properties and choices determine the class of optimisation problem at hand. The properties of the problem that are important are:

- Whether or not the problem is constrained.
- Whether or not the cost function and the constraints are linear or non-linear.
- Whether or not the cost function and constraints are continuous and differentiable.

The choices made by the designer that influence the class of optimisation problem are:

- Whether to convert the multi-objective problem to a scalar form or treat it as a vector optimisation.
- Whether to employ methods with a priori, posteriori or no articulation of preferences.

It is possible to manipulate the problem to change the class of optimisation problem. Converting the multi-objective problem to a scalar problem has some advantages over vector optimisation. Firstly there are a larger number of optimisation methods to choose from and secondly, it is typically computationally less expensive to calculate a solution for a scalarized problem than for a vector problem [26]. With respect to the articulation of preferences, [26] notes that methods with posteriori articulation of preferences are not as efficient as methods that require a priori articulation of preferences.

As mentioned, the problem at hand is a multi-objective problem with both equality and inequality constraints. It was decided to use a cost function formulation that converts the multi-objective problem into a scalar problem that can be solved using single-objective optimisation methods. The choice of cost function is described below, along with some additional considerations with respect to cost and constraint function normalisation.

4.2.3 The choice of a cost function

There are a large number of cost-function formulations available that are suitable for various designer choices with respect to preferences and scalar/vector optimisation methods. The details of these methods are discussed in detail in [26] and will not be repeated here. From the discussion by [26] on the methods available for multi-objective optimisation, the goal-programming method and the weighted global criterion method are identified as being the two options that fit the problem the best. These formulations are both scalar formulations and they are also both methods that employ a priori articulation of preferences. These two methods are discussed here.

Goal-Programming

In the goal-programming formulation, the optimisation attempts to minimise the deviation of some function from the goal set by the designer. [25] defines the goal-programming method mathematically as follows:

$$\min \sum_{i=1}^k (d_i^+ + d_i^-) \quad (4.4)$$

subject to $f_j(x) + d_j^+ - d_j^- = b_j$

$d_j^+, d_j^- \geq 0$

$d_j^+ d_j^- = 0$ and $j = 1$ to k .

Where b_j is the goal set by the designer for the j^{th} objective function and d^+ and d^- represent under-achievement and over-achievement respectively.

While this method fits the problem very well, it does have some drawbacks. The primary drawback for this application is the fact that two additional design variables, d_j^+ and d_j^- , are defined for each goal. Since the problem being considered has a large number of goals, this formulation will dramatically increase the number of design variables needed to solve the problem, conceivably more than doubling the number of design variables for a typical aircraft application.

Weighted Global Criterion

The weighted global criterion method has some similarities to the goal programming method. As [26] notes, replacing the goal b_j from the previous formulation with the utopia point results in a formulation that is a type of global criterion method. For the weighted global criterion method, the cost function can be defined as:

$$J = \sum_{i=1}^k [w_i (f_i(x) - f_i^o)^p] \quad (4.5)$$

where w_i represents the relative weights between the objective functions and f_i^o is the utopia point.

As it stands in equation (4.5), this method requires the utopia point to be known. However, it can be replaced by the aspiration point, z , if the utopia point is too computationally expensive to calculate [26]. An attractive feature is the property that the power p adds to the cost function. This term is useful for penalising the objective functions with larger errors more than those with smaller errors if the value of p is selected such that $p > 1$.

With reference to the optimality criterion defined earlier in this chapter, the ability of this formulation to find optimal points is discussed here. As seen in equation (4.5), this method minimises the error between the solution point and the utopia point. Global criterion methods are therefore often called compromise programming methods [26] and these methods will find compromise solutions. Additionally, if the weights are fixed and the utopia points are such that they do not fall in the feasible criterion space then minimising (4.5) is necessary and sufficient for Pareto optimality [26]. The definitions for necessary and sufficient conditions are given below.

Necessary - *"If a formulation provides a necessary condition for Pareto optimality, then for a point to be Pareto optimal, it must be a solution to that formulation."* [26]

Sufficient - *"...if a formulation provides a sufficient condition, then its solution is always Pareto optimal, although certain Pareto optimal points may be unattainable."* [26]

4.2.4 Normalisation

Cost function normalisation

In scalarised multi-objective optimisation, multiple objectives are placed in one objective function where they are related by weightings. Since these objective functions may have different units, it is necessary to normalise the objectives in the objective function in order to make comparisons meaningful [25]. There are a number of normalisation approaches available, as discussed by [25] but only the most robust approach available is provided here:

$$f_i^{norm} = \frac{f_i(x) - f_i^{\circ}}{f_i^{max} - f_i^{\circ}} \quad (4.6)$$

where $f_i(x)$ represents the objective function and f_i^{max} represents the maximum value that the objective function is able to or likely to reach.

This approach usually normalises each objective function to a value between zero and one [25]. The normalisation ensures that each goal that makes up the cost function is of the same order of magnitude so that the effect on the cost of each goal is of comparable size.

Constraint normalisation

Similarly to the objective functions, constraints may also have different units. This could become problematic if the optimisation method used defines a maximum constraint violation to be used for all constraints. If the constraints are not normalised then this constraint violation will not have consistent meaning between constraints. For this reason, constraints also need to be normalised. The normalisation approach for constraints is typically to divide the constraint by its allowable value, δ_p [25]. For example, the constraint:

$$\delta - \delta_p \leq 0 \quad (4.7)$$

would be transformed into:

$$\frac{\delta}{\delta_p} - 1 \leq 0 \quad (4.8)$$

However, not all constraints can be normalised in this way. If, for example, the allowable value is $\delta_p = 0$, another method is required. Such constraints can either be left as they are or they can be divided by 100 to provide a percentage value [25].

4.3 Selecting an optimisation method

The problem so far has been developed to the point where some choice must be made about what optimisation method will be used to solve it. The properties of the problem formulation that determine what optimisation methods are suitable are listed below:

- The problem is constrained.
- The problem is continuous and differentiable.
- A scalarised cost function has been selected.
- The cost function selected takes a priori articulation of preferences.

The optimisation method selected must therefore be capable of handling scalar constrained optimisation problems. Either gradient- or non-gradient-based methods can be used. As discussed in section 1.1.3, both linear as well as non-linear optimisation formulations have been considered when solving the control allocation problem. For the simpler control allocation problem where only the roll, pitch and yaw moments are considered, ignoring adverse forces and the trim problem, the linear programming approach has been used to great effect. Methods such as the simplex method and interior point methods have been shown to be effective and while developing the formulation given in chapter 3, the simplex method was applied to the simple one dimensional roll problem in order to investigate the properties of the control allocation problem. However, when the full 6-DOF problem is formulated including the trimming problem, some of the resulting equations can become non-linear, depending on the simplifying assumptions made for these equations. Additionally, the cost function formulation selected has some properties that make the choice of $p > 1$ attractive. Given the additional complexity of the full 6-DOF problem as well as the choice of cost function, non-linear optimisation methods are deemed to be more suitable.

In the category of non-linear constrained optimisation, there are a number of methods that can be considered. These methods fall into two classes: direct and indirect methods [27]. For direct methods, constraints are handled explicitly while for indirect methods, a sequence of unconstrained problems are minimised in order to solve the constrained problem [27]. The methods that fall under each class are shown in table 4.1. [27] gives thorough explanations for each of the methods given in table 4.1 as well as providing advantages and disadvantages for many of these methods. From these considerations, these methods are evaluated for applicability to this problem.

Table 4.1: Non-linear Direct and Indirect optimisation methods

Direct Methods	Indirect methods
<ul style="list-style-type: none"> • Random search methods • Heuristic search methods <ul style="list-style-type: none"> • Complex method • Objective and constraint approximation <ul style="list-style-type: none"> • Sequential linear programming method • Sequential quadratic programming method • Methods of feasible directions <ul style="list-style-type: none"> • Zoutendijk's method • Rosen's gradient projection method • Generalised reduced gradient method 	<ul style="list-style-type: none"> • Transformation of variables technique • Sequential unconstrained minimisation techniques <ul style="list-style-type: none"> • Interior penalty function method • Exterior penalty function method • Augmented Lagrange multiplier method

Direct methods - Firstly, [27] immediately notes that random search methods are not as efficient as the other methods given in table 4.1 and these methods are therefore not considered to be suitable. As for the complex method, it is not capable of handling problems that have equality constraints [27]. It will not be suitable for this application since there are equality constraints present. The sequential linear programming (SLP) method is efficient for solving convex programming problems. The methods of feasible directions and the generalized reduced gradient method could be used although it should be kept in mind that Rosen's gradient projection method is primarily effective only on problems with linear constraints [27]. [27] considers the sequential quadratic programming (SQP) method one of the best methods of optimisation available.

Indirect methods - The transformation of variables technique is limited to solving problems where the constraints are fairly simple functions of the design variables. The sequential unconstrained minimisation techniques do not appear to present any problems and could therefore be considered as possible choices.

For the direct methods, the SLP and SQP methods as well as the gradient projection and generalized reduced gradient (GRG) methods are suitable. From the available indirect methods, the sequential unconstrained minimisation techniques appear to be possible

choices. It is expected that any of these methods will be acceptable choices. As an additional aid in deciding on an optimisation method, [27] gives the following guidelines for a general constrained optimisation problem: *"The sequential quadratic programming approach can be used for solving a variety of problems efficiently. The GRG method and Zoutendijk's method of feasible directions, although slightly less efficient, can also be used for the efficient solution of constrained problems. The ALM and penalty function methods are less efficient but are robust and reliable in finding the solution of constrained problems."*

Given the above information the SQP method was chosen to solve the problem. The SQP method is very powerful and will also allow for the future extension of the problem to include non-linear constraints, which some of the other methods are not capable of. In the next section, the SQP algorithm is discussed in more detail.

4.4 The SQP algorithm

This section provides a brief description of the sequential quadratic programming algorithm used to solve the control allocation problem. This is not an exhaustive discussion on sequential quadratic programming but rather a short introduction into the functioning of the algorithm and what its benefits and shortcomings are. For a more comprehensive discussion, [25] or [27] can be consulted.

The sequential quadratic programming algorithm used to solve the control allocation problem is based on the one developed in [25] and it has its origins in the constrained steepest descent method [28]. This method forms part of the group of numerical methods used to directly find solutions for constrained problems [25], and it is an iterative method that calculates a step size and search direction to iteratively move toward an optimal solution. The design variable is changed for each iteration as follows:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k)} \quad (4.9)$$

where the change in design variables is calculated as follows:

$$\Delta\mathbf{x}^{(k)} = \alpha_k d^{(k)} \quad (4.10)$$

with α_k the step size and $d^{(k)}$ the search direction. The algorithm solves the following two problems: finding the search direction and finding the step size. In order to find the search direction, the non-linear optimisation problem is approximated using a standard quadratic

form with linear approximations of the constraints [28]. One of the side-effects of using an approximation of the original problem is that, for each iteration, the problem is solved for a small region around the current design point and consequently, global solutions cannot be guaranteed. Another consequence of the approximation used when calculating the search direction is that the step size must be limited so that the approximated quadratic sub-problem remains valid. The quadratic approximation sub-problem is defined below:

$$\min q(\mathbf{x}) = \mathbf{c}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \mathbf{H} \mathbf{x} \quad (4.11)$$

subject to:

$$\mathbf{N}^T \mathbf{x} = \mathbf{e} \quad (4.12)$$

$$\mathbf{A}^T \mathbf{x} \leq \mathbf{b} \quad (4.13)$$

with non-negative variables $\mathbf{x} \geq 0$. This quadratic sub-problem is related to the original optimisation problem as follows: \mathbf{c} is the derivative of the original cost function and \mathbf{H} is the hessian matrix for the original cost function, \mathbf{N} contains the derivatives of the equality constraints and \mathbf{e} contains the equality constraints themselves, \mathbf{A} contains the derivatives of the inequality constraints and \mathbf{b} contains the inequality constraints themselves. This QP sub-problem can be solved using any suitable method.

Once the search direction has been calculated, the step size needs to be found. To find the step size α_k , a line search method can be used, for example the inaccurate line search method. The inaccurate line search method follows the following basic rule for selecting a trial step size:

$$\alpha_j = \left(\frac{1}{2}\right)^j ; j = 0, 1, 2, 3, 4... \quad (4.14)$$

The initial trial step size is therefore α_j for $j = 0$ so that $\alpha_0 = 1$. The descent conditions are then evaluated to see whether this is a valid step size. If the descent conditions are not satisfied, j is incremented and the descent conditions are evaluated again. This process continues until valid step size is found or the step size becomes smaller than some predefined value. The descent conditions are determined by the descent function:

$$\Phi_k = f_k + RV_k \quad (4.15)$$

so that it is a function of the cost of the objective function f_k as well as constraint violations V_k through the penalty parameter R . The descent conditions that must be satisfied in selecting the step size are then

$$\Phi_{(k+1),j} + \alpha_j \beta_k \leq \Phi_k \quad (4.16)$$

with $\beta_k = \gamma \|d\|^2$ where γ is a user-selected constant. An initial value for R in equation (4.15) is also selected by the user and the penalty parameter is updated internally as necessary for subsequent iterations.

The above SQP algorithm now requires the problem cost function and constraints as well as their derivatives. The following section describes how the goals and constraints defined in chapter 3 are conditioned to fit the chosen cost function and the normalisation procedures described in section 4.2. The derivatives of the cost and constraints are also given.

4.5 Optimisation problem setup

The sections of this chapter up to this point described the definitions and tools required to set up a multi-objective optimisation problem. Using these tools and the control allocation problem formulation developed in chapter 3, the factors involved in setting up the optimisation problem so that it can be solved practically are discussed here.

4.5.1 Design vector

The design vector is the vector of design variables that describes a specific solution and in this case it is made up of the variables used to describe the problem formulation described in chapter 3, as follows:

$$\mathbf{x} = \left[\mathbf{T}_{AB} \quad \mathbf{T}_{AA} \quad \mathbf{T}_{AE} \quad \mathbf{T}_{AR} \quad \mathbf{T}_{AT} \quad \mathbf{T}_{TB} \quad \mathbf{T}_{TA} \quad \mathbf{T}_{TE} \quad \mathbf{T}_{TR} \quad \mathbf{T}_{TT} \right]^T \quad (4.17)$$

Since each of the aerodynamic mixing and biasing vectors has n elements and each of the thrust mixing and biasing vectors has p elements, the design vector for the control allocation problem has $5n + 5p$ elements. The problem size will therefore scale upwards with five design variables for each additional physical actuator taken into account for this formulation. For each additional virtual actuator that is added to the problem formulation, $n + p$ additional design variables will be needed. The equations that follow are written in terms of the design vector \mathbf{x} .

4.5.2 Setting up the cost function and finding its derivatives

The cost function of the problem consists of four main goals: trimming goals, minimising moment errors, minimising adverse effects and minimising control usage. Within each main goal of the optimisation, there are a number of sub-goals, each with its own weighting. These sub-goals will be defined in this section and the resulting cost function will be shown at the end. In addition to the cost function, the SQP algorithm requires the first order partial derivatives of the cost function with respect to the design variables.

The weighted global criterion method defined in section 4.2.2 is used. Since each objective being considered is an error minimisation objective, the components of the utopia point for each of the objective functions are $f_i^0 = 0$. Additionally, $p = 2$ is chosen to place a larger emphasis on objectives with larger errors. The cost function format is therefore as follows:

$$J = \sum_{i=1}^k [w_i (f_i(\mathbf{x}))^2] \quad (4.18)$$

essentially reducing this to the weighted exponential sum from [26]. Each objective function is normalised using the robust normalisation approach defined in section 4.2.3, once again with the utopia point f_i^o set to zero so that:

$$f_i^{norm} = \frac{f_i(\mathbf{x})}{f_i^{max}} \quad (4.19)$$

The equations that form the goals, sub-goals and constraints of the optimisation problem are now written in the form defined by equations 4.18 and 4.19.

Trimming

The trim goals developed in chapter 3 are given by equations 3.51 and 3.52. These terms are added to the cost function to ensure that the side-slip and bank angles are minimised. This sub-goal also allows the trade-off between bank-angle and side-slip angle mentioned in section 3.4.1 to be implemented. The trimming sub-goals are normalised and summed in the trimming goal so that:

$$f_t(\mathbf{x}) = w_{t_1} \left(\frac{\beta_T}{\beta_T^{max}} \right)^2 + w_{t_2} \left(\frac{\phi_T}{\phi_T^{max}} \right)^2 \quad (4.20)$$

where w_{t_1} and w_{t_2} implement the relative weighting between bank angle and side-slip angle and $w_{t_1} + w_{t_2} = 1$. The terms β_T^{max} and ϕ_T^{max} are used to normalise the side-slip and bank-angles.

The partial derivatives for the trimming goal are calculated analytically and are given by:

$$\frac{\partial f_t(\mathbf{x})}{\partial \mathbf{x}} = 2w_{t_1} \left(\frac{\beta_T}{\beta_T^{max}} \right) \frac{\partial \beta_T}{\partial \mathbf{x}} + 2w_{t_2} \left(\frac{\phi_T}{\phi_T^{max}} \right) \frac{\partial \phi_T}{\partial \mathbf{x}} \quad (4.21)$$

The expanded partial derivatives for $\frac{\partial \beta_T}{\partial \mathbf{x}}$ and $\frac{\partial \phi_T}{\partial \mathbf{x}}$ were calculated analytically although these equations are not shown as they are not discussed in more detail and they will result in unnecessary clutter.

Force and moment errors

In chapter 3, the four sub-goals for the force and moment error minimisation problem were defined. These goals are described by equations 3.53 to 3.56. Each of the four sub-goals are normalised and weighted so that the force and moment error minimisation goal is:

$$f_m(\mathbf{x}) = w_{m_1} \left(\frac{L^r - L^d}{L^{max}} \right)^2 + w_{m_2} \left(\frac{M^r - M^d}{M^{max}} \right)^2 + w_{m_3} \left(\frac{N^r - N^d}{N^{max}} \right)^2 + w_{m_4} \left(\frac{T^r - T^d}{T^{max}} \right)^2 \quad (4.22)$$

where w_{m_1} , w_{m_2} , w_{m_3} and w_{m_4} are the weights for each sub-goal and are selected so that $\sum_{i=1}^4 w_{m_i} = 1$. The terms L^d , M^d , N^d , T^d as well as L^{max} , M^{max} , N^{max} and T^{max} are all scalar so that the partial derivatives for the moment errors are given by:

$$\begin{aligned} \frac{\partial f_m(\mathbf{x})}{\partial \mathbf{x}} = & 2w_{m_1} \left(\frac{L^r - L^d}{L^{max}} \right) \frac{\partial L^r}{\partial \mathbf{x}} + 2w_{m_2} \left(\frac{M^r - M^d}{M^{max}} \right) \frac{\partial M^r}{\partial \mathbf{x}} + 2w_{m_3} \left(\frac{N^r - N^d}{N^{max}} \right) \frac{\partial N^r}{\partial \mathbf{x}} + \\ & 2w_{m_4} \left(\frac{T^r - T^d}{T^{max}} \right) \frac{\partial T^r}{\partial \mathbf{x}} \end{aligned} \quad (4.23)$$

The expanded partial derivatives for $\frac{\partial L^r}{\partial \mathbf{x}}$, $\frac{\partial M^r}{\partial \mathbf{x}}$, $\frac{\partial N^r}{\partial \mathbf{x}}$ and $\frac{\partial T^r}{\partial \mathbf{x}}$ are once again not shown for the reasons mentioned previously.

Adverse effects

Similarly, the adverse effects goal is made up of a number of sub-goals that follows the trend described by equations 3.57 to 3.61. This one broad goal is made up of the following four segments: adverse effects for a roll command, adverse effects for a pitch command, adverse effects for a yaw command and adverse effects for a thrust command. Within each of these segments, there are five sub-goals. Expanding $f_a(\mathbf{x})$ to show all of these sub-goals is cumbersome and consequently only the first few, which describe the adverse effects for a roll command, are given:

$$\begin{aligned}
f_a(\mathbf{x}) = & w_{a_1} \left(\frac{M_A^r}{M_A^{max}} \right)^2 + w_{a_2} \left(\frac{N_A^r}{N_A^{max}} \right)^2 + w_{a_3} \left(\frac{X_A^r}{X_A^{max}} \right)^2 + w_{a_4} \left(\frac{Y_A^r}{Y_A^{max}} \right)^2 + \\
& w_{a_5} \left(\frac{Z_A^r}{Z_A^{max}} \right)^2 + \dots
\end{aligned} \tag{4.24}$$

where the terms w_{a_k} are the weights assigned to each sub-goal and $\sum_{i=k}^{20} w_{a_k} = 1$. The partial derivatives for the adverse effects were also calculated although they are not shown here for the same reasons as those mentioned previously.

Actuator usage

The final part of the cost function is the actuator usage minimisation goal, which was given by equation 3.62 in chapter 3. This goal is now added to the cost function as follows:

$$f_c(\mathbf{x}) = \mathbf{w}_c \langle \mathbf{x} \rangle^2 \tag{4.25}$$

where \mathbf{w}_c is a row vector of weights whose sum is equal to one and \mathbf{x} is the design vector of the optimisation problem. The operator $\langle \rangle^2$ represents the square of each component of the column vectors contained within the brackets. The derivatives were calculated but are not shown for the same reason as mentioned previously.

Resulting cost function

Combining all of the goals above into a single cost function with weightings applied to each goal, the final cost scalar cost function is:

$$J = w_{tm}f_t(\mathbf{x}) + w_{mm}f_m(\mathbf{x}) + w_{am}f_a(\mathbf{x}) + w_{cm}f_c(\mathbf{x}) \tag{4.26}$$

where w_{tm} , w_{mm} , w_{am} and w_{cm} are the weights assigned to each goal. These weights are used to specify the relative importance of each goal and they are selected so that their sum is equal to one. These weightings are chosen according to the priorities of the control allocation system. The derivative of the cost function is then the sum of the derivatives of each of its parts, so that:

$$\frac{\partial J}{\partial \mathbf{x}} = w_{tm} \frac{\partial f_t(\mathbf{x})}{\partial \mathbf{x}} + w_{mm} \frac{\partial f_m(\mathbf{x})}{\partial \mathbf{x}} + w_{am} \frac{\partial f_a(\mathbf{x})}{\partial \mathbf{x}} + w_{cm} \frac{\partial f_c(\mathbf{x})}{\partial \mathbf{x}} \tag{4.27}$$

4.5.3 Setting up the constraints and finding their derivatives

The cost function has now been defined and the final part of setting up the optimisation problem is to define the constraints. The constraints are defined and normalised here.

Equality constraints

As seen in section 3.4.1, there are three equality constraints that are imposed on the optimisation. These are given by equations (3.3), (3.5) and (3.11). Therefore:

$$h_1(\mathbf{x}) = (-q_T S C_{D_T}) + \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (4.28)$$

$$h_2(\mathbf{x}) = q_T S \bar{c} (C_{m_0} + C_{m_\alpha} \alpha_T + \mathbf{C}_{m_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} = 0 \quad (4.29)$$

$$h_3(\mathbf{x}) = q_T S b (C_{n_0} + C_{n_\beta} \beta_T + \mathbf{C}_{n_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{n_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (4.30)$$

The derivatives were calculated analytically but are not shown here as they are not discussed in more detail and will result in unnecessary clutter.

Inequality constraints

The inequality constraints imposed on the system are the inequalities given in section 3.4.6. These constraints ensure that the control allocation system does not violate the physical position constraints placed on the system. For the practical implementation with the SQP method, these inequality constraints are converted to the standard \leq form. Due to the large number of inequality constraints only an example is given of the process, showing the first two inequality constraints. The original actuator position constraint for a roll command is described by 4.31 below:

$$\delta_{min} \leq \mathbf{T}_{AA} \delta_{AS} + \mathbf{T}_{AB} \leq \delta_{max} \quad (4.31)$$

This constraint is transformed into two inequality constraints which are normalised according to the rule shown in equation 4.8:

$$g_1(\mathbf{x}) = \frac{\mathbf{T}_{AA} \delta_{AS} + \mathbf{T}_{AB}}{\delta_{max}} - 1 \leq 0 \quad (4.32)$$

$$g_2(\mathbf{x}) = \frac{-\mathbf{T}_{AA} \delta_{AS} - \mathbf{T}_{AB}}{\delta_{min}} + 1 \leq 0 \quad (4.33)$$

These are examples of the constraint equations that ensure that the maximum and minimum actuator positions are not violated. The rest of the inequality constraints have a similar form. The derivatives for these inequality constraints were calculated analytically although they are not provided here for the reasons mentioned previously.

4.5.4 Final optimisation problem formulation

The final optimisation problem formulation is now given as follows:

$$\min J = w_{tm}f_t(\mathbf{x}) + w_{mm}f_m(\mathbf{x}) + w_{am}f_a(\mathbf{x}) + w_{cm}f_c(\mathbf{x}) \quad (4.34)$$

subject to the equality constraints:

$$h_1(\mathbf{x}) : \quad (-q_T S C_{D_T}) + \mathbf{C}_{x_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (4.35)$$

$$h_2(\mathbf{x}) : \quad q_T S \bar{c} (C_{m_0} + C_{m_\alpha} \alpha_T + \mathbf{C}_{m_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{m_{\delta_t}} \mathbf{T}_{TB} = 0 \quad (4.36)$$

$$h_3(\mathbf{x}) : \quad q_T S b (C_{n_0} + C_{n_\beta} \beta_T + \mathbf{C}_{n_\delta} \mathbf{T}_{AB}) + \mathbf{C}_{n_{\delta_T}} \mathbf{T}_{TB} = 0 \quad (4.37)$$

and the inequality constraints:

$$g_1(\mathbf{x}) = \frac{\mathbf{T}_{AA} \delta_{AS} + \mathbf{T}_{AB}}{\delta_{max}} - 1 \leq 0 \quad (4.38)$$

$$g_2(\mathbf{x}) = \frac{-\mathbf{T}_{AA} \delta_{AS} - \mathbf{T}_{AB}}{\delta_{min}} + 1 \leq 0 \quad (4.39)$$

⋮

$$g_m(\mathbf{x})$$

4.6 SQP algorithm practical implementation

The optimisation problem is now set up and is ready to be solved practically. A prior implementation of the SQP algorithm discussed above was developed and used by [28] in his masters thesis and that implementation of the SQP algorithm was used without alteration here as well.

4.7 Conclusion

This chapter described the general multi-objective problem and some considerations for solving such problems. A scalar cost function was chosen for the optimisation problem and some normalisation techniques were discussed. The SQP algorithm was selected as the best method to solve the problem and the algorithm was briefly described. The mathematical formulation of the control allocation problem as an optimisation problem was developed to conform with the cost function and normalisation approaches selected and the derivatives required by the SQP algorithm were calculated. The resulting optimisation problem must now be solved using the SQP algorithm selected. The practical implementation of the control allocation system is described in the next chapter.

Chapter 5

Base system implementation

Given the summary of the optimisation problem formulation described in section 4.5.4 and the SQP optimisation algorithm described in section 4.4, the overall structure of the system can be described. This section starts with an overview of the Matlab implementation, followed by a more detailed description of the implementation methods and issues for each section.

5.1 Overview

This implementation of the control allocation system can be illustrated as shown in figure 5.1. The implementation is divided into four levels. The top level handles the data input required by the control allocation system. It consists of offline data input, online data input and some data processing. The second layer encapsulates the optimisation problem defined in section 4.5.4 and is implemented using Matlab m-files that contain the cost function, equality constraints and inequality constraints as well as the first derivatives of each of these. The third layer is the optimisation layer which contains the SQP optimisation algorithm being used. This layer consists of the parameter setup for the optimisation and the practical implementation of the SQP algorithm itself. The final layer is the system outputs layer. The results generated by the optimisation are used here to generate the necessary mixing vectors as well as the required performance information. The implementation of each of these layers is discussed in more detail in the following sections.

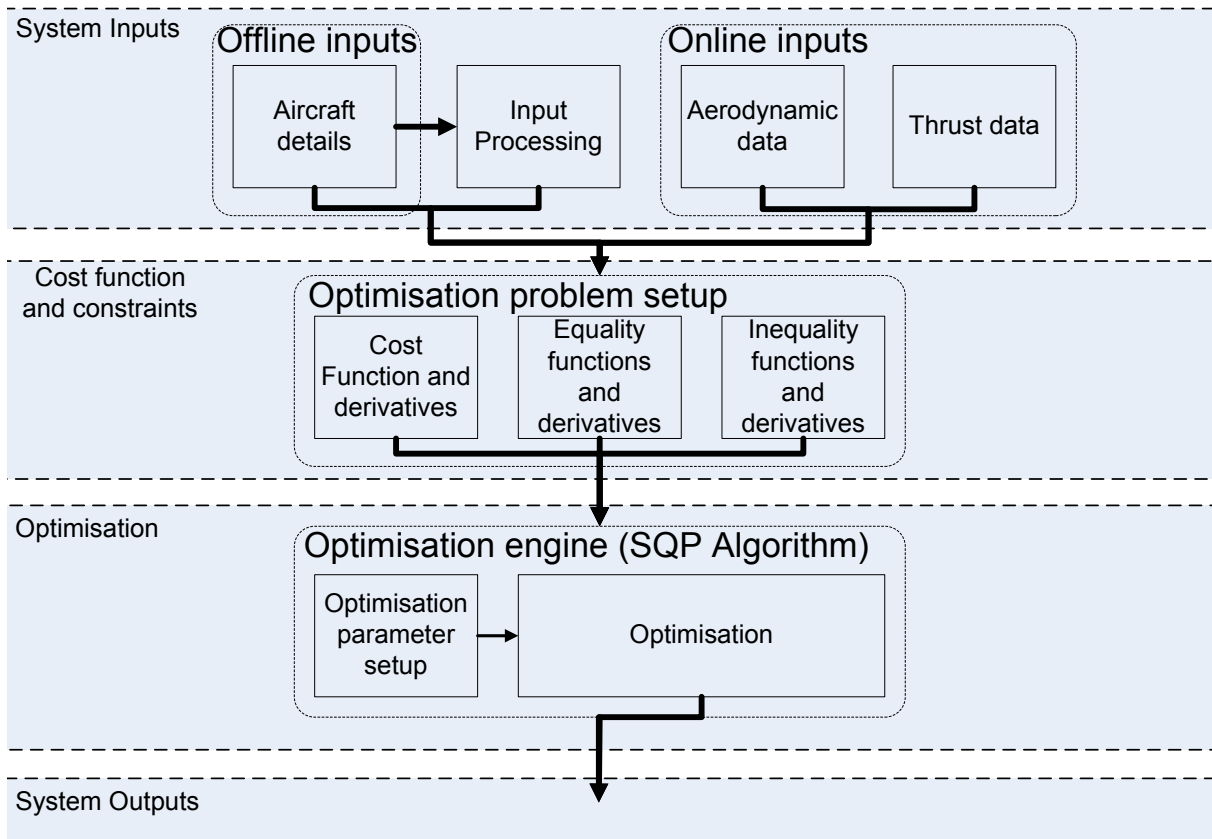


Figure 5.1: Control allocation implementation overview

5.2 System inputs

The system inputs form the interface between the control allocation system and the user as well as the interface between the control allocation system and the FTC system. This interface has been designed to require minimal data about the aircraft and minimal understanding of the underlying control allocation system while still being useful for a wide variety of aircraft. The interface with the user is through the offline inputs and the interface with the FTC system is through the online inputs.

5.2.1 Offline Inputs

The formulation developed in chapter 3 is developed such that it requires the desired performance details of the aircraft as well as certain information about the configuration of the aircraft. The offline input data provides the information needed to calculate the parameters required by the control allocation formulation.

Table 5.1: User input requirements

Aerodynamic actuators	
# aerodynamic actuators,	Actuator position limits
Engines	
# Engines, X_e , Y_e , Z_e , ϕ_e , ψ_e , Max Thrust,	Rotor moments
Physical data	
m, b, S, c, A, e,	
Desired performance data	
L^d , M^d , N^d , T^d	
Further data required for trim	
g , ρ , V_T	

Aircraft data and physical constants

The aircraft data provided by the user describes the physical configuration of the aircraft and is required when calculating the trim biases of the actuators as well as to calculate the effectiveness of the engines in producing forces and moments. The data that is required is listed in table 5.2.1.

This information is provided offline by the user or designer before flight begins and all of this information is generally known. The actuator position limits are given as a vector of position limits so that each actuator can have individual limits. Similarly, the maximum thrust, engine positions, engine offsets and rotor moments are also vectors so that many engines can be considered. All of the above data is captured in an aircraft initialisation template.

Input processing

The aircraft data entered by the user does not directly provide all of the data required by the control allocation system. The information entered by the user is used to generate the required values, such as engine effectiveness coefficients and normalisation data. All of the input processing calculations are performed in a processing module.

Engine effectiveness coefficients The engine information provided by the user describes the physical configuration of the engines relative to body axes. In order to generate the necessary engine coefficients as used in the equations describing the forces and moments that the engines can generate, equations 2.27 and 2.28 from section 2.1.4 are used

to calculate the force coefficients and equations 2.42, 2.43 and 2.44, also from section 2.1.4, are used to calculate the moment coefficients.

Normalisation data The normalisation values required in order to normalise the various goals and sub-goals of the cost function are calculated if necessary. The normalisation data can either be supplied by the user or it can be calculated, given the nominal virtual actuator mixing vectors used by the aircraft.

5.2.2 Online inputs: Aircraft model data

The aircraft model data required by the control allocation system consists of the aircraft's aerodynamic control effectiveness coefficients, engine effectiveness coefficients and the static biases on the forces and moments acting on the damaged aircraft. In practice, this information would be provided by the system I.D. subsystem of the FTC system via the supervisor. The control allocation system requires this data in the following format:

$$C = \begin{bmatrix} C_{l_0} & C_{l_\beta} & C_{l_{\delta_1}} & \dots & C_{l_{\delta_n}} \\ C_{L_0} & C_{L_\alpha} & C_{L_{\delta_1}} & \dots & C_{L_{\delta_n}} \\ C_{y_0} & C_{y_\beta} & C_{y_{\delta_1}} & \dots & C_{y_{\delta_n}} \\ C_{m_0} & C_{m_\alpha} & C_{m_{\delta_1}} & \dots & C_{m_{\delta_n}} \\ C_{n_0} & C_{n_\beta} & C_{n_{\delta_1}} & \dots & C_{n_{\delta_n}} \end{bmatrix} \quad (5.1)$$

and

$$K_e = \begin{bmatrix} K_{e_1} & \dots & K_{e_p} \end{bmatrix} \quad (5.2)$$

where C is a coefficient matrix including the aerodynamic derivatives and actuator effectiveness coefficients of the aircraft. K_e is a vector containing an effectiveness multiplier for each engine. The multiplier can be set from a value between 0 and 1, 0 being a totally failed engine and 1 being a healthy engine. An intermediate value would represent an engine that has lost some percentage of its effectiveness.

Due to the theoretical nature of the work done in this thesis and the fact that the FTC system and system I.D. sub-system were not available, a Matlab function was written to call up pre-defined test cases and provide the data in the above format to the system. Due to the large number of failure cases considered, a further support system was implemented to automatically generate test cases. This allows for the easy generation of a large number of test cases that can be used to evaluate the control allocation system and potentially

the FTC system as well. This additional functionality adds value to the system as a tool for investigating the control allocation problem.

5.3 Cost function and constraints

The implementation of the cost function and constraint equations given in section 4.5.4 is briefly described followed by a description of the process followed in selecting the weights and normalisation values for the various goals of the optimisation.

5.3.1 Cost and constraints

As discussed in chapter 4 on optimisation, the SQP algorithm being used requires the cost function, inequality constraint equations, equality constraint equations and first-order derivatives for each of these. The SQP implementation developed by [28] takes as inputs three files containing this information: a cost module, an equality function module and an inequality function module. The equations defined in chapter 4 are implemented in Matlab code in these modules.

The control allocation system and the equations defining the optimisation problem have been developed so that the system is valid for aircraft with n aerodynamic actuators and p engines. In order for the control allocation system to be usable for n aerodynamic actuators and p engines, the partial derivatives of the equations given in section 4.5 must be implemented practically in such a way as to scale to the problem size. To facilitate this, the partial derivatives are grouped according to what portions of the design vector \mathbf{x} they are associated with. The result is a set of equations that can be implemented in such a way as to fit a general aircraft configuration.

Similarly, the derivatives for the equality and inequality constraints are also implemented to scale with problem size. Additionally, the number of inequality constraints themselves also differs with problem size and these equations are implemented such that this scaling is handled automatically. This automatic problem scaling allows the system to be used for many different aircraft without requiring the user to delve deeper into the control allocation system than the offline-input level shown in figure 5.1.

The fact that this process of automatically scaling the problem was necessary highlights one of the difficulties with gradient-based methods. The derivatives of the cost and constraint functions are required and these can become unwieldy. There is no guarantee that a more complex control allocation formulation will result in derivatives that can easily be implemented in a general way as described above. When implementing these equations,

mistakes are difficult to find and have a large effect on the robustness of the optimisation as well as on correctness of the results. In addition to the difficulty in finding mistakes, any change in the cost or constraint function requires that the derivatives be updated, sometimes making even small adjustments to the cost or constraints a cumbersome process. To avoid some of these problems, numerical methods of finding the derivatives can be employed, although this will be less accurate and more computationally expensive. Calculating the derivatives analytically is therefore preferred.

5.3.2 Weighting and normalisation selection

Section 4.5 gives the mathematical details of the cost and constraint functions. Since the final problem formulation is a scalar multi-objective optimisation problem with many sub-goals, there are a large number of weighting terms that need to be assigned values. Assigning these values will determine the priorities of the goals and sub-goals of the optimisation. These weights will have a large effect on the solutions generated by the optimisation algorithm and they must therefore be chosen with care.

Weights

The weights that apply to the four primary goals of the control allocation system are discussed first, followed by a discussion on the weights selected for the sub-goals that make up these four main goals.

Main goals Table 5.2 lists each of the primary weights and gives a brief description of its importance. From this, it is clear that the w_{mm} weight must always have the highest value, followed by the w_{am} weight. The w_{tm} and w_{cm} weights must be selected so they do not negatively affect the results for the force and moment and adverse effects minimisation more than necessary. The weights are therefore chosen as shown in equation 5.3. The priorities discussed were selected to minimise the error between the desired and achievable system performance after a failure has occurred on an aircraft. If other goals are more important, these priorities will change.

$$w_{mm} > w_{am} \geq w_{tm} > w_{cm} \quad (5.3)$$

The final values used for these weights differs between the different control allocation configurations tested in the next chapter. For more details on the final selections, see section 6.1.

Table 5.2: The weights that prioritise the four main goals of the optimisation

Weighting term	Description
w_{tm}	This term sets the priority of the trim goals in the cost function. These goals are considered secondary to the force and moment goals as they are mainly present to ensure that the trim bank- and side-slip angles are kept in check and to allow for the trade-off between bank angle and side-slip trim angles.
w_{mm}	This term sets the priority of the force and moment error minimisation goals in the cost function. This is the goal with the highest priority since it is the focus of the control allocation system.
w_{am}	This term sets the priority of the adverse effects minimisation goals in the cost function. This goal is of high importance as the adverse effects have a negative effect on aircraft performance. However, this goal is of a lower importance than the force and moment error minimisation.
w_{cm}	This term sets the priority of the minimisation of actuator usage for virtual actuators. This is a secondary goal that will compete with the other more important goals.

Sub-goals Each of the primary goals above is constructed from a number of sub-goals. These sub-goals are weighted to reflect the relative importance of each sub-goal within the context of its primary goal.

- **Trim goal** - The trim goal has sub-goals for minimising β_T and ϕ_T . The weights for these sub-goals are selected so that $w_{t_1} = (1 - \lambda)$ and $w_{t_2} = \lambda$ where $0 \leq \lambda \leq 1$ and is selected by the user depending on which asymmetric trim configuration described in section 3.4.1 is desired.
- **Force and moment error minimisation** - The force and moment error minimisation goal has four sub-goals, one for each virtual actuator. Each virtual actuator is considered to be of equal importance to the aircraft so that the weightings for these four sub-goals are equal.
- **Adverse effects goal** - The adverse effects goal has a large number of sub-goals, five for each virtual actuator. These twenty sub-goals are given equal weights at this stage.

- **Control minimisation** - This goal has a sub-goal for each design variable. Each design variable is given equal weight initially although these weights can be adjusted to manipulate what actuators the optimisation favours.

Normalisation

The sub-goals and constraints are normalised according to the methods selected in section 4.2.4. The normalisation values were selected as follows:

For the trim goals, the maximum allowable angles for β_T and ϕ_T were used for normalisation. The maximum values for these terms are set by the small angle assumption made for the aircraft model developed in chapter 2 and the normalisation values were set to 10° .

For the force and moment goals, f_i^{max} was initially calculated using the effectiveness of the aerodynamic actuators and engines and the maximum deflections of the aerodynamic actuators and the maximum thrust settings. It was however discovered during testing that using these maximum values did not produce the desired results. The optimisation worked for some failure cases but not for others. In an attempt to remedy this situation, the normalisation approach was altered slightly and the maximum expected values were used, i.e. the largest values that are likely to occur for a failure situation. Substituting these values into the normalisation equations improved the general robustness of the system.

The inequality constraints were normalised according to the maximum and minimum actuator deflections allowed for the aerodynamic actuators and according to the maximum thrust that the engines can produce.

5.4 Optimisation layer

The optimisation layer contains the SQP optimisation algorithm used to solve the control allocation problem. The optimisation algorithm has certain parameters that need to be selected based on the problem being solved and a start vector must be chosen for the optimisation to begin with.

Parameter selection

The parameters that are required by the SQP algorithm being used are listed and described in table 5.3. These parameters need to be selected so that they are suitable for

Table 5.3: SQP optimisation algorithm parameters

Parameter	Description
ϵ_0	The maximum deviation from zero allowable - reduces problems due to numerical errors.
k_{max}	The maximum number of SQP iterations allowed before the SQP algorithm stops.
k_{smax}	The maximum number of simplex iterations allowed before the SQP algorithm stops.
R	The penalty parameter that determines how harshly constraint violations are penalised in the descent function.
ϵ_{pcd}	The potential constraint deviation which is used to identify which constraints are considered active or epsilon active.
ϵ_{pcv}	The potential constraint violation which is used to determine at what point a constraint is violated.
ϵ_{cv}	The convergence parameter value which is used to determine when the design has converged.
$\epsilon_{\alpha_{min}}$	The minimum allowable step size.

the specific problem being solved. The values that were selected for these parameters are listed in table 5.4.

Start vector selection options

The SQP optimisation algorithm requires an initial design vector to begin with. This initial design vector is called the start vector and some considerations in choosing this start vector are given here.

The start vector for the optimisation may have a dramatic effect on the results of the optimisation since the SQP algorithm converges to a local minimum. [28] evaluated the performance of the SQP optimisation algorithm being used for a particular problem using various start vectors and showed that even small variations in start vector can have a large effect on the solution. The start vector can affect the final results as well as how quickly the optimisation converges on those results. For this reason, the start vector must be carefully selected. Some possibilities are discussed.

Table 5.4: Optimisation algorithm parameter values

Parameter	Value
ϵ_0	1×10^{-11}
k_{max}	300
k_{smax}	1000
R	1
ϵ_{pcd}	0.002
ϵ_{pcv}	0.001
ϵ_{cv}	1×10^{-11}
$\epsilon_{\alpha_{min}}$	1×10^{-11}

The three possibilities that are considered are a random start vector, a zero start vector and a warm start vector. A random start vector, while useful for evaluating the general performance and robustness of the system for a single failure case, does not make for a good platform from which to compare many differing failure cases to one another due to the random nature of the starting points as well as the fact that infeasible starting points may be selected. For an in-depth investigation into the performance of the control allocation system for a specific failure case, the random start vector would be suitable. For the investigation done here a fixed start vector used for every problem would prove more useful for comparing the control allocation system for different failure scenarios. From the two remaining options of start vector, a decision must be made on whether to use a zero start vector or warm start vector. A warm start vector that is constructed using the nominal healthy aircraft mixing and biasing vectors will provide a good real-world test for the system while also allowing for the comparison between failure cases. Since the failures that are being considered will result in some deviation from these nominal values, a warm start vector using nominal values is expected to give better performance than a zero start vector.

5.5 System Outputs

The algorithm provides certain information after control allocation has taken place. This information can be divided into two categories: control mixing information and performance information.

5.5.1 Control mixing information

If the control allocation is successful, the algorithm will provide the control system with eight control mixing vectors: mixing vectors for aerodynamic actuators for roll, pitch, yaw and thrust and mixing vectors for thrust actuators for roll, pitch, yaw and thrust.

5.5.2 Performance information

Once control allocation is complete, the algorithm will provide the control system with information regarding the capabilities of the reconfigured aircraft. This information includes the following:

- The moment generating capabilities for roll, pitch and yaw.
- The thrust capabilities.
- The adverse affects caused by a roll, pitch, yaw or thrust command.
- Maximum allowable virtual roll, pitch, yaw and thrust commands that do not saturate the physical actuators.

This information is sufficient to allow the supervisor to make a decision as to whether or not the control system of the aircraft needs to be reconfigured. If it is necessary for the control system to be reconfigured, this performance information along with the information provided by the system I.D. sub-system is sufficient to allow successful reconfiguration so that the aircraft is used most effectively given the new limits on its abilities.

5.5.3 Optimisation histories and other useful data

In addition to the outputs required by the FTC system, the current implementation also stores histories of the optimisation process so that detailed analysis can be done of failure cases. This information includes histories of the total cost function value, the penalties associated with goals over the course of the optimisation, the changes in the design variables, step sizes and penalty parameters.

5.6 Conclusion

The control allocation system has now been practically implemented. The Matlab implementation can automatically handle a wide variety of aircraft configurations and the data

input and output interfaces required have been defined. The optimisation algorithm has been set up for the problem at hand and a number of support structures for testing large number of failure cases are now available. The system is now ready for testing and this is the focus of the next chapter.

Chapter 6

Base system testing and results

The previous chapter described the practical implementation of the control allocation system. The testing of this system is now completed according to the testing procedure described below. The overall performance of the system is discussed and some more detailed performance evaluations are done for each of the test aircraft.

6.1 Testing procedure

In order to investigate the properties of the control allocation system, a testing procedure that investigates every facet of the problem is desired. To this end, a number of configurations of the control allocation system will be tested. These tests will be done on two very different aircraft types, with each aircraft undergoing simulated failures from a number of types of failure categories. The control allocation system configurations, aircraft types and failure categories are described below.

6.1.1 Control allocation system configurations

Three different configurations are tested to highlight the effects of considering different goals in the control allocation process. The three configurations are:

- **Configuration 1** - Control allocation taking into account only trim calculations, moment generating ability and adverse moments. For this configuration, the weights for the primary goals are selected as follows: $w_{mm} = 0.8$, $w_{am} = 0.1$, $w_{tm} = 0.1$ and $w_{cm} = 0$. Additionally, the adverse force and moment sub-goals are weighted such that the adverse forces are ignored.

- **Configuration 2** - Control allocation taking into account trim calculations, moment generating ability and adverse forces and moments. For this configuration, the weights for the primary goals are selected as follows: $w_{mm} = 0.8$, $w_{am} = 0.1$, $w_{tm} = 0.1$ and $w_{cm} = 0$. Additionally, the adverse force and moment sub-goals are weighted such that the adverse forces are taken into account.
- **Configuration 3** - Control allocation taking into account trim calculations, moment generating ability, adverse forces and moments as well as control minimisation. For this configuration, the weights for the primary goals are selected as follows: $w_{mm} = 0.5$, $w_{am} = 0.1$, $w_{tm} = 0.1$ and $w_{cm} = 0.3$.

The weights for the configurations are not tuned for a specific aircraft and are kept constant throughout testing, unless otherwise stated. This is done to make direct comparisons between aircraft possible. It is expected that comparing these configurations will reveal the impact that each component has on the solutions provided. It is also expected that different configurations will have a larger impact on some aircraft types than on others.

6.1.2 Tested aircraft

Two aircraft are used to test the base algorithm. The test aircraft are the Modular UAV mentioned in chapter 1, which was designed by the CSIR, and the Variable Stability UAV. The configurations of these aircraft are briefly described below, along with a motivation for choosing these particular aircraft.

- **Modular UAV** - This is a fixed-wing aircraft designed for high actuator redundancy using highly effective actuators. It has a left and a right aileron (δ_{al} and δ_{ar}), a left and a right flaperon (δ_{fl} and δ_{fr}), a left and a right elevator (δ_{el} and δ_{er}), a left and a right rudder (δ_{rl} and δ_{rr}) and a left and a right engine (δ_{tl} and δ_{tr}). In addition to having redundant actuators, the placement of the actuators means that some of them have fairly high effectiveness outside of their normal usage. See Appendix A.1 for a full description of the aircraft.
- **Variable stability UAV (VSA)** - This is a blended-wing body UAV with fewer actuators than the Modular UAV. It has a left and a right aileron (δ_{al} and δ_{ar}), a left and a right elevon (δ_{el} and δ_{er}), a left and a right rudder (δ_{rl} and δ_{rr}) and a single engine (δ_t). Since this UAV has no tail, the ailerons and elevons are responsible for both pitch and roll. Another consequence of having no tail is that the rudder effectiveness is also fairly low. See Appendix A.2 for a full description of the aircraft.

These two particular aircraft were chosen in order to demonstrate the effects of the aircraft type on the results of the control allocation system. The Modular UAV has high actuator redundancy while the VSA UAV has moderate actuator redundancy. It is therefore expected that the control allocation system will be more successful in minimising force and moment errors as well as adverse effects for the modular UAV than for the VSA UAV. Comparing the results obtained for these two aircraft will provide valuable insight into what goals of the control allocation system are the most beneficial to a specific type of aircraft.

6.1.3 Failure categories

Each configuration is tested using a number of failure cases. The types of failures that were tested can be broken down into categories so that meaningful comparisons can be made between them. The following categories are defined:

- **Single actuator soft failures** - Soft failures are defined as failures where the failed actuator is not failed at an extreme position. Examples would include actuators stuck at zero offset or small deviations from zero. Compensating for singular failures of this type is not expected to be problematic for an aircraft with sufficient actuator redundancy although they could become problematic for aircraft with weak actuator redundancy.
- **Single actuator extreme failures** - Extreme failures are defined as failures where the actuator fails at an extreme angle. Examples include failures where the actuator remains stuck at a large deflection angle or at its maximum deflection. Compensating for singular failures of this type is expected to be problematic even for aircraft with moderate redundancy.
- **Multiple actuator soft failures** - Multiple actuator soft failures are defined as failures where two actuators have failed simultaneously with one actuator failed at zero offset and the other actuator at a moderate angle. Multiple actuator failures, even of the soft variety, can become problematic for any aircraft, especially if multiple actuators responsible for the same task are failed. With these types of failures, it can be expected that performance will be reduced.
- **Multiple actuator extreme failures** - Multiple actuator extreme failures are defined as failures where two actuators have failed simultaneously with one actuator failed at zero and one actuator failed at a large deflection angle. Failure cases with more than two simultaneous actuator failures are also categorised as extreme

failures. These types of failures are expected to be problematic for any type of aircraft and performance reductions are expected.

- **Other failures** - This failure category is used for failure cases not covered above. Engine failures are handled here as well as failures that result in changes to the aircraft parameters without resulting in a loss of actuators.

Using the failure types defined above, the control allocation system can be thoroughly tested. The results of these tests can be used to investigate the effectiveness of the control allocation system developed here as well as some of the properties of the control allocation problem in general.

6.2 Additional notes on the testing procedure followed

The following points should be noted when considering the results given in the rest of this chapter.

- All failure cases are performed for a fixed trim airspeed and altitude. The airspeed for the Modular UAV was 22 m.s^{-1} and for the VSA UAV it was 20 m.s^{-1} .
- The weights for the control allocation configurations are not tuned for a specific aircraft but selected based on the performance of a small number of initial test cases. These weights remain unchanged for all failure cases unless otherwise stated.
- Unless otherwise stated, all results for individual test cases are the results for control allocation configuration 2.
- Adverse effects are minimised as follows for the results in this chapter: The difference between the adverse effects for a nominal virtual actuator and the reconfigured virtual actuator is minimised.
- The results for control allocation system configuration 3 were not investigated in as much detail as the other two configurations and these results are not discussed along with the other configurations. A brief overview is given of the results towards the end of the chapter.

6.3 Testing

The rest of this chapter describes the results generated for the testing procedure described above. The control allocation system is validated using the healthy aircraft configuration

of the Modular UAV and the failure categories defined above are then each handled separately.

6.3.1 Validation of optimisation and problem parameters

Before the full test suite is run including all the system configurations and failure categories, the control allocation system is validated using the nominal no-failure case of the Modular UAV. The nominal, healthy case for this aircraft is run starting with the design vector initialised to zero. This test will show whether or not the control allocation system converges to a solution that is comparable to that of the manually calculated solution to the problem.

Trim angles

The trim angles of the healthy aircraft are shown in table 6.1. The angle of attack, α , calculated by the control allocation system is very similar to the angle of attack calculated by hand. The side-slip and bank angles are both very small although not zero. All of these angles are well within the small angle assumption made for the aircraft model and the deviations from the nominal values are small. The deviation in angle of attack from the manually calculated value is not unexpected since the control allocation algorithm factors in all of the available actuators, not only the elevators and engines, as is done manually. The control allocation algorithm therefore has more options with regard to trimming the aircraft, such as generating lift using the flaps. Given these differences in the manner in which the control allocation system calculates the trim variables, a slight difference in the trim angle of attack is understandable. The deviations in bank-angle and side-slip angle from zero, however, are unexpected for a healthy aircraft as these angles are added to the cost function as goals to be minimised. These two angles are small (hundredths of a degree) and since the goal of trimming the aircraft is to find the approximate trim conditions so that the virtual actuator mixings do not violate the position constraints of the actuators, these small deviations from zero should not be problematic.

Actuator usage

Looking at the way that the control allocation system uses the actuators that are available on the aircraft, the two areas of interest are the biasing on the actuators for trim and the gains assigned to mix the virtual roll, pitch, yaw and thrust commands to the physical actuators. Table 6.2 shows the trim settings calculated by the control allocation system compared to the nominal trim settings calculated manually.

Table 6.1: Trim angles for the Modular UAV test aircraft for the nominal case

	α (deg)	β (deg)	ϕ (deg)
CA	2.09	0.0282	0.0523
Nominal	2.24	0	0

Table 6.2: Comparison between nominal and CA actuator biases for the Modular UAV

	Actuator biases (deg)							
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	-0.236	0.247	0.409	0.371	-3.99	-4.05	0.762	-0.958
Nominal	0	0	0	0	-4.27	-4.27	0	0

	(%)	
	δ_{tl}	δ_{tr}
CA	9.16	9.53
Nominal	9.3604	9.3604

It can clearly be seen that the control allocation system distributes control effort amongst all of the available actuators. Looking at actuators a_5 and a_6 , which are the left and right elevators respectively, it is clear that the the control allocation system uses values very similar to those that are calculated manually which is expected since these are the primary pitching actuators of the aircraft. The ailerons and flaps are used symmetrically (keep in mind that aileron deflections are defined such that positive deflections result in negative moments) to generate lift, explaining the discrepancy in angle of attack seen in table 6.1. The rudders are also deflected by a small amount. This may seem unusual until it is considered that the Modular UAV's rudders are offset vertically from the c.g of the aircraft such that the rudders are capable of generating a pitch moment. The control allocation system calculates the solutions based on the actuator effectiveness provided by the control coefficients and as such will use any actuator that provides the necessary force or moment. Adjusting the weights of the control allocation system or adding additional goals and constraints, such as drag minimisation, should suppress these sorts of allocations. The thrust settings are approximately what is expected, although they are not perfectly symmetrical. The results given for the trim settings for the aircraft appear to

be acceptable so far as approximate trim settings are concerned. Perfectly symmetrical results for a healthy aircraft would be preferable although practically, these small deviations from symmetry should not cause any problems and the forces and moments caused by the sum of these actuator deflections are zero. The trim results shown for trim angles and trim actuator biases are all slightly off symmetry. The deviations from symmetry are for the most part small, although the intuitive solution to the problem is to use symmetric actuator deflections for a healthy aircraft. Running the trim optimisation alone without considering the additional goals of control allocation results in trim settings that are symmetrical, indicating that the trim optimisation is capable of achieving symmetric results. The full control allocation problem is more complex and the optimisation is set up to find solutions quickly. Selecting the optimisation parameters to be more strict in terms of zero-tolerances and constraint violations should result in symmetric results.

The virtual actuator mixing vectors generated by the control allocation system are shown in table 6.3. This table once again illustrates that the control allocation system uses all of the available actuators to find a solution.

Looking at the mixing gains for the virtual roll actuator, δ_A , The largest portion of the roll command is provided using differential ailerons. The second largest contribution is from differential flap usage which is not unexpected given the flaps' high roll effectiveness. Once again, the rudders are used unconventionally. This is due to a combination of the fact that the rudders are capable of providing some roll moment, as well as the fact that the control allocation system is attempting to match the adverse effects caused when the nominal mixing is used. The control allocation system can also be set to attempt to remove adverse forces altogether. The rudders in this instance primarily reduce adverse yaw and side-force errors.

Looking at the virtual pitch command, the elevators are the primary pitching actuators, similarly to the manually selected actuators, with an additional portion of the pitching moment command going to the rudders. The fact that the system assigns some portion of the pitch command to the rudders is understandable since the rudders are capable of pitching the aircraft nose up. This may however not be desirable since the rudders are only capable of pitching the aircraft in one direction. This does highlight a potential problem with the current control allocation formulation. The control allocation system currently takes as a goal the desired force or moment for the virtual actuator in a single direction, in the case of the virtual pitch actuator, nose up. It calculates the new virtual commands based on this desired value and assumes that the virtual actuator will be able to pitch in the opposite direction by reversing the command given. Clearly this mixing through the rudders will therefore result in small errors. In a practical application, the

Table 6.3: Comparison between nominal and CA selected mixing vectors for the Modular UAV

		Actuator usage gains (% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_A	CA	67.6	67.6	-45.9	45.9	-1.11	1.11	-8.22	-8.22
	Nominal	100	100	0	0	0	0	0	0
δ_E	CAZ	-0.577	0.577	-1.35	-1.35	95.6	95.6	-20.5	20.5
	Nominal	0	0	0	0	100	100	0	0
δ_R	CA	-9.81	-9.81	-12.2	12.2	-23.9	23.9	91.0	91.0
	Nominal	0	0	0	0	0	0	100	100
δ_T	CA	0	0	0	0	0	0	0	0
	Nominal	0	0	0	0	0	0	0	0

		δ_{tl}	δ_{tr}
δ_A	CA	3.27	-3.27
	Nominal	0	0
δ_E	CA	0	0
	Nominal	0	0
δ_R	CA	-21.05	21.05
	Nominal	0	0
δ_T	CA	50	50
	Nominal	50	50

penalties associated with actuator usage should be adjusted or additional constraints added to ensure that the control allocation system does not generate impractical virtual actuators such as this one.

Looking at the virtual yaw command, all of the aerodynamic actuators take some portion of the control command. The rudders are the primary yawing actuators with large gains but the ailerons, flaps and elevators are all assigned fairly significant gains as well. Removing all of the actuators except the rudders from the virtual actuator results in larger errors in both yaw and side-force, revealing that the combination of ailerons, flaps and elevators are used to generate some yaw while keeping adverse effects minimised.

Finally, the virtual thrust actuator generated is exactly the same as the nominal virtual thrust actuator.

Table 6.4: Force and moment errors for the CA results

Force or moment error	
(%)	
δ_A	0.184
δ_E	0.083
δ_R	0.138
δ_T	0.0401

Resulting errors

Given the above mixing gains that are then used to define the virtual actuators, the errors for the primary forces and moments as well as the adverse forces and moments for each virtual actuator can be calculated. The errors for the primary moments are given in table 6.4. The errors for the adverse forces and moments for this case are negligible and are therefore not shown.

A note on adverse error normalisation

In order to make it possible to gauge the effects of adverse force and moment errors, the errors are normalised. The normalisation procedure is as follows: The moment errors are normalised according to the moments of inertia of the aircraft and the force errors are normalised according to the mass of the aircraft. This results in values that describe the acceleration caused by adverse force and moment errors. These errors are then scaled to provide values that reflect the accelerations due to the errors for a 5° virtual command.

The following reference values were chosen for the accelerations. For the angular acceleration: 0.01 rad.s^{-2} , which would result in an error in angle of approximately 28° if held for 10 seconds, is chosen. For linear acceleration: 0.1 m.s^{-2} is chosen, which would result in a positional deviation of 5m if held for 10 seconds. These choices are used as reference values and errors on these orders of magnitude are considered significant. Much smaller errors are considered zero.

6.3.2 Single actuator soft failures

The first suite of tests run for the control allocation system is the single actuator soft failure category. The nature of these failures is described in section 6.1.3 and the specific

failure cases used for each aircraft are given in appendices B.1 and B.2. The overall performance of the control allocation system is first discussed, followed by a discussion of the performance of the system for each aircraft, where a few test cases are discussed in more detail.

Overall system performance

Table 6.5 gives an overview of the performance of the system. This table compares the performance of the system between the three system configurations for the two different aircraft being considered. The number of cases that fall below 5% error, between 5%–10% error, between 10%–20% error, between 20%–50% error and the number of cases that cannot be solved for are shown, along with the approximate percentage of failure cases that these numbers represent. The percentage error is based on the largest error of the four virtual commands.

The Modular UAV's high actuator redundancy results in excellent performance for this failure category, with most of the solutions falling within the 5% error range and the system successfully finds a solution for every test case. There are a few test cases that have larger than desired errors and these cases will be discussed in more detail in the next section. Taking into account adverse forces in the control allocation system (configuration 2) only has a small effect on the average number of iterations required to find a solution and the error distribution is not significantly altered. The effects on adverse forces are shown in figure 6.1, which shows the average errors for primary and adverse forces and moments. The system is clearly capable of dramatically reducing the average adverse lift without having a dramatic negative affect on the primary forces and moments. In fact, while the average error for the virtual yaw command is slightly increased, the error for the virtual pitch command is actually reduced. This result is unexpected and it is thought that this is due to the fact that the addition of the adverse forces to the optimisation has changed the relative weights associated with the primary moment errors, resulting in a slightly better combination of relative weights.

The VSA UAV's moderate actuator redundancy results in moderate performance of the control allocation system when compared to the Modular UAV, with just under 60% of cases having errors smaller than 5%. The addition of the adverse forces into the system for configuration two results in slightly poorer results, with minimal benefits with respect to adverse forces reduction. This is illustrated in figure 6.2, which shows that only a small reduction in adverse forces is achieved at the expense of larger primary force and moment errors. Table 6.5 also shows that the majority of the cases that previously managed errors between 5% and 20% have now shifted to the larger error slots.

Table 6.5: Control allocation performance overview for failure category 1

	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	57	57	Modular UAV
error < 5%	51 (89)	49 (86)	
5% < error < 10%	0	2 (3.5)	
10% < error < 20%	0	2 (3.5)	
20% < error < 50%	6 (11)	4 (7)	
50% < error	0	0	
No solution	0	0	
Average # of Iterations	17.3	19.60	
	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	43	43	VSA UAV
error < 5%	25 (58)	23 (56)	
5% < error < 10%	4 (10)	0	
10% < error < 20%	3 (7)	1 (2)	
20% < error < 50%	4 (10)	8 (19)	
50% < error	1 (2)	4 (10)	
No solution	6 (14)	7 (16)	
Average # of Iterations	25.76	28.95	

Comparing the difference in performance of the system between the two aircraft, it is noteworthy that the VSA UAV already shows the effect of actuator redundancy on system performance. Even for the most lenient of failure categories, the system takes more SQP iterations to find solutions for the VSA UAV and the overall error distribution is markedly worse than that of the Modular UAV.

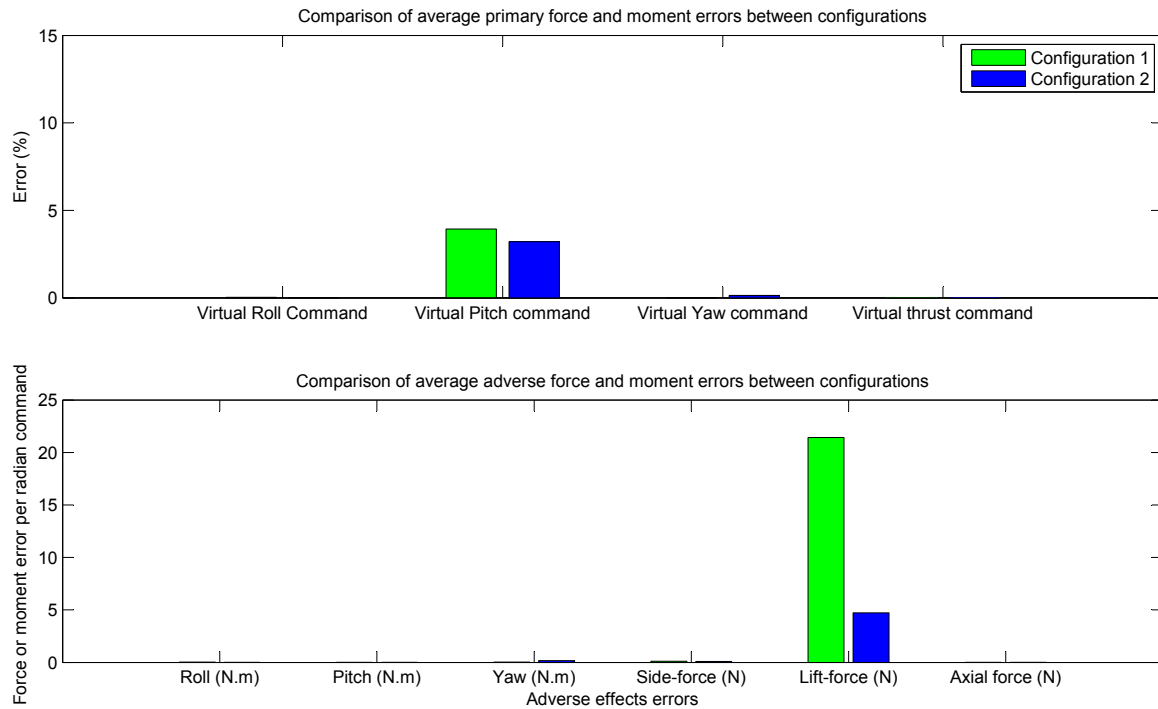


Figure 6.1: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 1 showing the benefit of taking into account adverse forces for the Modular UAV

Detailed analysis

In this section, each aircraft is discussed in slightly more detail. Figures 6.3 and 6.4 provide more detail as to the distribution of errors for specific failure cases. From these figures, it is possible to identify specific failure cases that perform well or poorly. From these figures, some failure cases are identified for closer inspection.

Modular UAV

It is clear from figure 6.3 that for the Modular UAV, the only failure cases that are problematic for the soft failure category are cases 30, 31, 32, 37, 38 and 39. These cases all involve elevator failures at non-zero positive angles, as can be seen in Appendix B.1, which gives a description for each failure case. In order to understand the reasons behind the poor performance of the system for these failure cases, failure case 31 is selected as a representative case from those listed above is examined in more detail. A closer look is also taken at failure case 46, which demonstrates a situation where taking actuator dynamics into account would be beneficial.

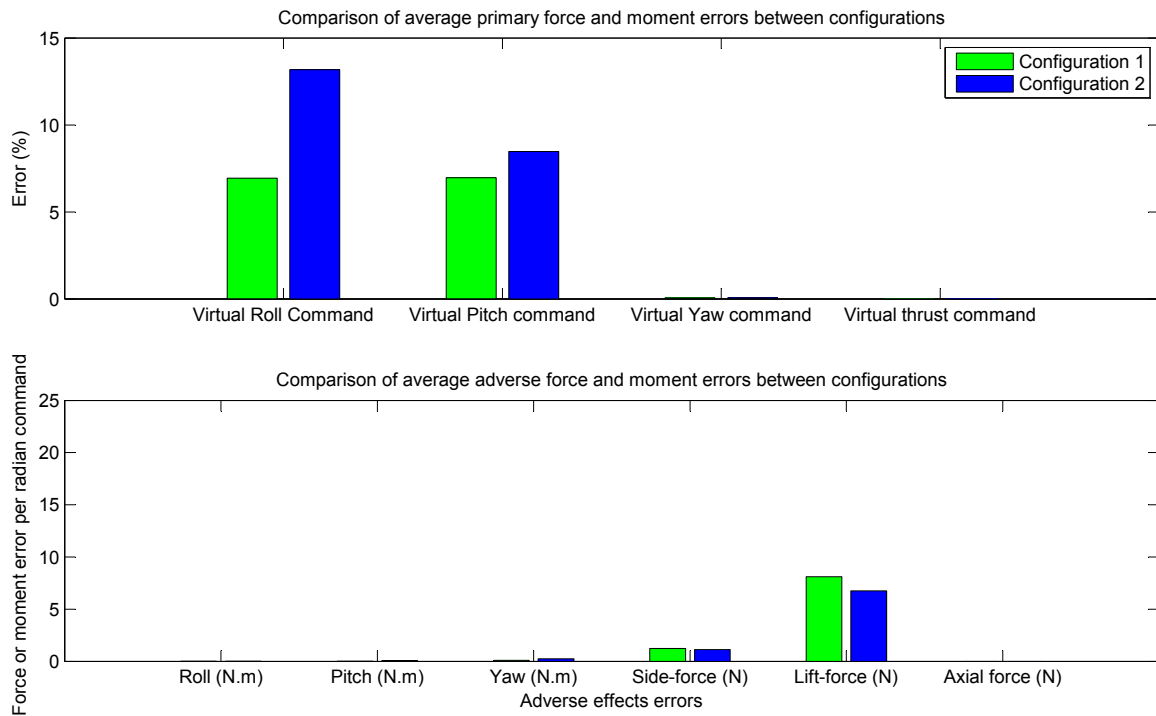


Figure 6.2: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 1 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV

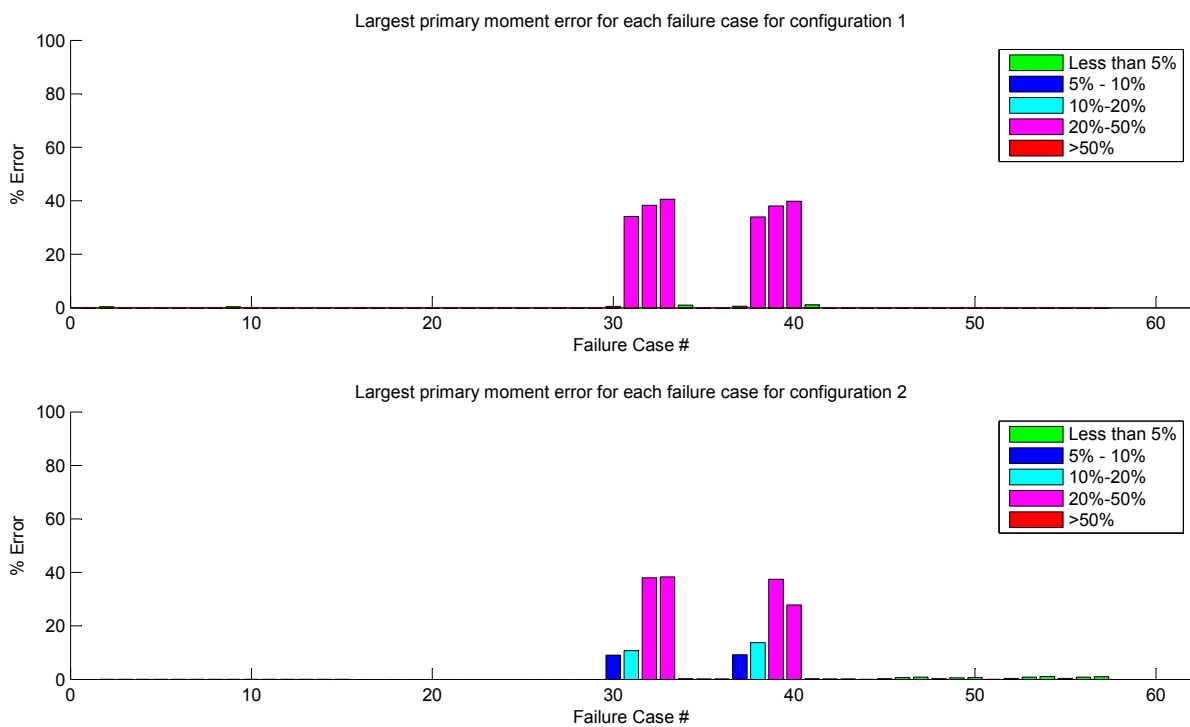


Figure 6.3: A plot of the largest primary moment errors for each failure case for failure category 1 (Modular UAV)

Modular UAV Failure case 31: Failed Left elevator at 5°

Table 6.6 shows the trim angles and table 6.7 shows the trim actuator settings that the control allocation system generates for a left elevator failure at 5 degrees. The trim angle of attack is smaller than nominal with small side-slip and bank angles also present. In order to generate the lift lost due to the smaller angle of attack, the ailerons and flaps are deflected downward in a symmetric configuration. The pitch bias introduced by the failure is compensated for with a large deflection of the right elevator to almost 66% of its maximum value. The asymmetric values for ailerons, flaps, rudders and thrust complete the trimming process, removing any adverse forces or moments caused by the differential elevator combination that results from the left elevator failure.

The nominal mixing vectors for virtual roll, yaw and thrust commands are still valid and only the mixing vector for the virtual pitch command is reconfigured, as shown in table 6.8. Looking more closely at the mixing gains used for the virtual pitch actuator, it is clear that the right elevator usage is lower than expected given that it is the only elevator still functional. This is due to the fact that the trimming process has used up most of the right elevators throw, leaving just over 5° of movement before it saturates in the one direction. Consequently, the gain for the virtual actuator cannot be increased while still ensuring that the position constraints are not violated. In an attempt to compensate for the reduced pitch effectiveness of the aircraft as well as to reduce adverse effects, the rest of the available actuators are also assigned significant gains. Deflecting only the right elevator would result in both adverse roll and yaw affects. The ailerons are used to remove adverse roll and the rudders reduce adverse yaw as well as providing some pitch moment. The flaps will provide a combination of lift, yaw and roll to cancel these adverse effects. Differential thrust is also employed in order to reduce adverse yaw.

Table 6.6: Trim angles for failure case 31 (Modular UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	1.46	0.298	-0.0961
Nominal	2.2429	0	0

Concerning the errors for the four virtual commands, obviously only the virtual elevator will be affected since the other three virtual actuators are not reconfigured. Table 6.9 shows the relevant errors. The aircraft has clearly lost a large percentage of its pitching ability for positive pitch commands and the adverse lift forces associated with the new virtual pitch actuator are also significant.

Table 6.7: CA-selected actuator biases for failure case 31 (Modular UAV)

		Actuator biases							
		(deg)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA		-1.42	1.27	1.85	2.98	5	-9.89	10.0	8.67
Nominal		0	0	0	0	-4.27	-4.27	0	0

		(%)	
		δ_{tl}	δ_{tr}
CA		10.1	8.61
Nominal		9.3604	9.3604

Table 6.8: CA-selected mixing vectors for failure case 31 (Modular UAV)

		Actuator usage gains							
		(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_E	CA	-23.8	17.4	21.9	23.8	0	102.0	-43.8	15.8
	Nominal	0	0	0	0	100	100	0	0

		δ_{tl}	δ_{tr}
δ_E	CA	-24.75	24.75
	Nominal	0	0

A possibility that can be investigated to improve the performance of the control allocation system for elevator failures is to trim the aircraft for different airspeeds to see if some pitching ability can be recovered at a higher or lower airspeed. This failure case is clearly problematic for the control allocation system due to the limited pitch-moment redundancy available if an elevator fails at a deflection angle that counteracts the pitch trim of the aircraft.

Table 6.9: Force and moment errors for failure case 31 (Modular UAV)

	Moment error	Adverse lift error
	(%)	(m.s ⁻² per 5° command)
δ_E	38.0	0.3923

Modular UAV failure case 46: Failed left rudder at 7.5°

The trim angles for this failure case are given in table 6.10. There are no significant deviations from the nominal values for this failure case. The trim actuator settings for the Modular UAV for this failure are given in table 6.11. The right rudder is clearly used to offset a large portion of the yaw bias introduced by the failure. While it is desired that slow actuators are used to trim the aircraft so that fast actuators can return to their default positions, the rudder failure not only introduces a yaw bias but also a side-force bias that must be removed. For this reason, differential thrust is not the best option for trimming the aircraft and only a small portion of the yaw bias is removed in this way. Asymmetric elevator deflections are used to reduce the pitch moment to zero while also aiding in reducing the yaw moment. The remaining actuators each have small deflections which help to complete the trimming process.

Looking at table 6.12, the virtual actuator mixings can be evaluated. The virtual roll and pitch actuators remain unchanged and only the virtual yaw actuator is reconfigured. The virtual yaw actuator uses a gain of approximately 180% on the right rudder, some differential thrust is also employed and the rest of the actuators are used primarily to reduce adverse effects although they also provided a small amount of yaw. The gain on the right rudder cannot be increased since the position constraints placed on the actuator would then be violated. The resulting virtual actuator uses a combination of fast (rudder) and slow (differential thrust) actuators in order to generate the desired yaw moment. This is clearly not ideal and this occurs because actuator dynamics are not taken into account in the current control allocation formulation. Taking into account actuator dynamics will resolve this issue.

The final errors for the virtual yaw actuator are all small, with the most significant value being the primary yaw moment error, which is 0.867%. The adverse effects errors are all negligible.

Table 6.10: Trim angles for failure case 46 (Modular UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	2.28	0.2	0.0724
Nominal	2.24	0	0

Table 6.11: CA-selected actuator biases for failure case 46 (Modular UAV)

Actuator biases (deg)								
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	0.801	0.529	0.682	-1.09	-1.19	-4.47	7.5	-5.95
Nominal	0	0	0	0	-4.27	-4.27	0	0

Actuator biases (%)		
	δ_{tl}	δ_{tr}
CA	9.18	9.5
Nominal	9.36	9.36

Table 6.12: CA-selected mixing vectors for failure case 46 (Modular UAV)

Actuator usage gains (% of δ mixed to actuator)									
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_R	CA	-8.9	-9.43	-11.3	11.5	-41.8	3.01	0	181.0
	Nominal	0	0	0	0	0	0	100	100

		δ_{tl}	δ_{tr}
δ_R	CA	-19.75	19.75
	Nominal	0	0

VSA UAV

Figure 6.4 shows the errors for the specific failure cases considered for the VSA UAV. It is clear that the failures that are the most problematic all fall within the first 30 failure cases considered. Looking at Appendix B.2, it can be seen that these failure cases deal with aileron failures (1-14) and elevon failures (15-28). The cases that perform well are the rudder failures (30-42). This is not unexpected since the ailerons and elevons play a dual role as pitching and rolling actuators and losing these actuators therefore results in a loss of a large percentage of the aircraft's pitching and rolling ability. Additionally, the rudder effectiveness is fairly low and the ailerons and elevons are capable of producing a yaw moment so that rudder failures are easily recovered from. Two failure cases are considered in more detail below.

VSA UAV Failure case 2: Failed left aileron at 2.5°

This failure may seem innocuous but it results in a reconfiguration of actuator usage that produces large errors in the resulting primary forces and moments produced. Looking at the errors shown in table 6.16, it is clear that the greatest effect of the failure is the loss of roll effectiveness. The ailerons are by far the most effective actuators in producing both a pitch moment and a roll moment for this UAV.

The trim settings for this failure are shown in tables 6.13 and 6.14. The failure of the left aileron at 2.5° requires a larger deflection of the right aileron along with a large deflection of the left elevon in order to trim the aircraft. While a more intuitive solution to the problem would be to deflect the remaining aileron to -2.5° and to trim with the elevons, the control allocation algorithm generates the results seen in table 6.14 since it favours the more effective actuators. It therefore uses the remaining aileron as much as possible as it is the most effective actuator.

Table 6.15 shows how the virtual actuators are realised. The virtual roll actuator is constructed using differential elevons, which are approximately half as effective as the ailerons in rolling the aircraft. The large trim deflection of the left elevon limits the gain that can be used without violating the position constraints placed on the actuators. Since adverse effects are being minimised, the gain that can be used on the right elevon is similarly limited as a large difference in the deflections between the left and right elevons would generate adverse effects. The remaining aileron's gain is set low for the same reason. The rudders are used to reduce adverse effects.

The virtual pitch actuator is constructed using the remaining aileron and the two elevons, with some rudder deflection used to reduce adverse forces and moments.

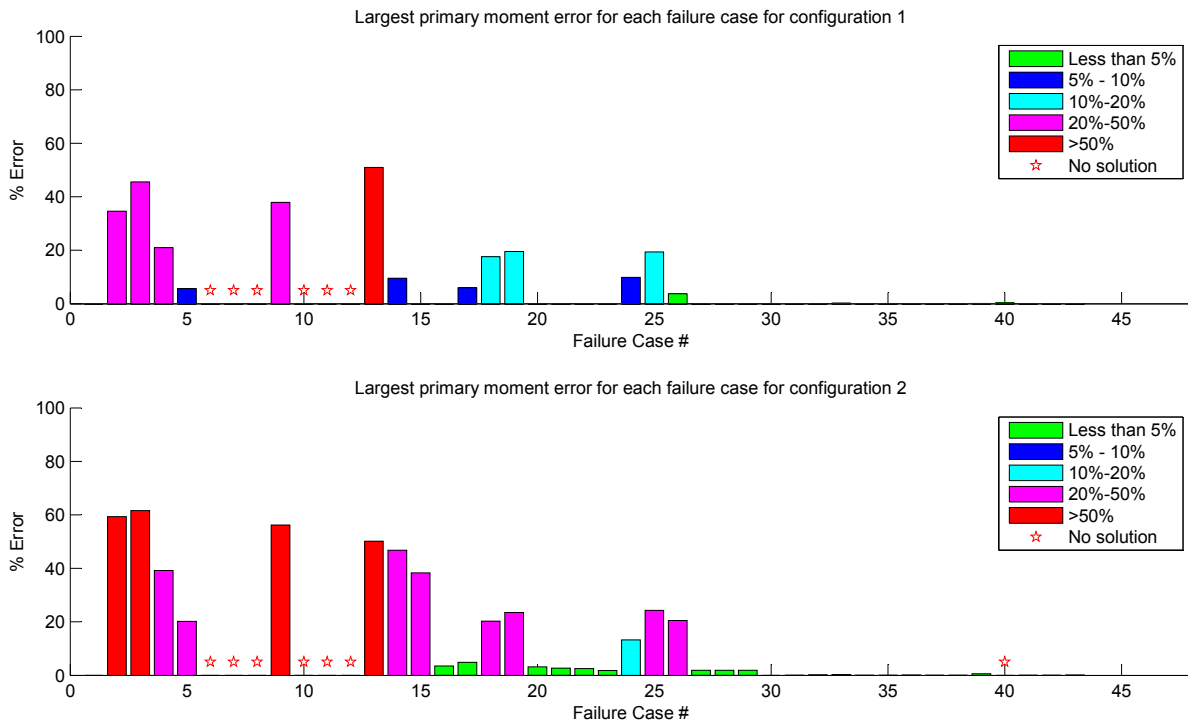


Figure 6.4: A plot of the largest primary moment errors for each failure case for failure category 1 (VSA UAV)

Table 6.13: Trim angles for failure case 2 (VSA UAV)

	α	β	ϕ
	(deg)	(deg)	(deg)
CA	3.21	0.869	-0.051
Nominal	1.99	0	0

Table 6.16 shows the errors for these two virtual actuators. The virtual roll actuator has lost a large percentage of its rolling ability and it now has a significant adverse pitch-moment and adverse side-force error as well. The virtual pitch actuator performs better although it has a large adverse yaw moment and adverse lift force present. The control system of the aircraft would need to be reconfigured with these new performance parameters taken into account.

If these results are compared to the results generated for configuration 1 of the control allocation system, it can be seen that attempting to reduce adverse forces has a negative effect for this aircraft. The virtual roll actuator errors for configuration 1 are shown in table 6.17 as the most significant changes occur here. There is clearly a smaller error for the primary moment of the virtual roll command while the adverse pitch and adverse

Table 6.14: CA-selected actuator biases for failure case 2 (VSA UAV)

	Actuator biases						
	(deg)						(%)
	δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
CA	2.5	-6.01	-7.35	-0.865	0.658	0.457	19.6
Nominal	4.314	-4.314	0	0	0	0	20.34

Table 6.15: CA-selected mixing vectors for failure case 2 (VSA UAV)

	Actuator usage gains							
	(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
δ_A	CA	0	6.85	-119.0	113.0	7.65	-2.45	0
	Nominal	100	100	-100	100	0	0	0
δ_E	CA	0	34.8	153.0	84.6	12.3	12.3	0
	Nominal	-100	100	100	100	0	0	0

Table 6.16: Force and moment errors for failure case 2 (VSA UAV)

	Moment errors	Adverse moment errors		Adverse force errors	
	(%)	(rad. s^{-2} per 5° command)		(m. s^{-2} per 5° command)	
		Pitch	Yaw	Side	Lift
δ_A	61.6	0.124		0.28	
δ_E	42.7		0.259		1.12

Table 6.17: Force and moment errors for failure case 2, system configuration 1 (VSA UAV)

	Moment errors (%)	Adverse moment errors (rad.s ⁻² per 5° command)		Adverse force errors (m.s ⁻² per 5° command)	
		Pitch	Yaw	Side	Lift
δ_A	45.63	0.0316		0.2	0.43

side-force errors are actually reduced. This however comes at the price of a large adverse lift error.

VSA UAV Failure case 21: Failed right elevon at -7°

This failure case is investigated in order to illustrate another factor that influences the performance of the control allocation system. The VSA UAV has only moderate control redundancy and this elevon failure is at a reasonably large angle, yet the system performs excellently. Failure cases 18 and 25 fail elevons at the same magnitude but in the opposite direction and figure 6.4 shows that these cases do not perform as well. The trim results for this failure are given in tables 6.18 and 6.19.

Looking at the trim settings for the actuators, it is clear that this particular failure has not made the trimming process significantly more difficult and large deflection angles are not needed to compensate for the failure due to the fact that the direction of the deflection aids the trimming process. The most effective pitching and rolling actuators, the left and right ailerons, still retain a large amount of throw so that the virtual actuators all still perform well. Table 6.20 shows the gains for the virtual actuators and it can be seen that large increases in the gains over the nominal values are used in order to maintain the roll and pitch performance of the virtual actuators. Table 6.21 shows the errors for this failure case. The primary moment errors for the virtual actuators are small but both have significant adverse effects errors. The virtual roll actuator has a significant adverse pitch moment and adverse side force while the virtual pitch actuator has a significant adverse yaw moment and adverse lift force.

Table 6.18: Trim angles for failure case 21 (VSA UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	3.22	1.4	0.195
Nominal	1.99	0	0

Table 6.19: CA-selected actuator biases for failure case 21 (VSA UAV)

	Actuator biases						δ_t (%)
	δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	
CA	2.49	-5.76	-7	-1.64	1.17	1.14	19.6
Nominal	4.314	-4.314	0	0	0	0	20.34

Table 6.20: CA-selected mixing vectors for failure case 21 (VSA UAV)

		Actuator usage gains						
		(% of δ mixed to actuator)						
		δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
δ_A	CA	145.0	147.0	0	-1.34	-17.7	-0.501	0
	Nominal	100	100	-100	100	0	0	0
δ_E	CA	-164.0	185.0	0	-40.0	8.19	7.63	0
	Nominal	-100	100	100	100	0	0	0

Table 6.21: Force and moment errors for failure case 21 (VSA UAV)

	Moment errors	Adverse moment errors		Adverse force errors	
	(%)	(rad. s^{-2} per 5° command)		(m. s^{-2} per 5° command)	
		Pitch	Yaw	Side	Lift
δ_A	0.977	0.088		0.16	
δ_E	2.56		0.0348		1.36

6.3.3 Single actuator extreme failures

The second category of failures is the single actuator extreme failure set. Section 6.1.3 describes this failure category and appendices B.1 and B.2 give further details of the individual failure cases used for each aircraft. An overview of the performance of the control allocation system for this failure category is given, followed by a brief look at each aircraft.

Overall system performance

Similarly to the previous failure category, the overall system performance is once again given in table form and the results are shown in table 6.22.

The Modular UAV once again performs well due to its high level of actuator redundancy. There are now a few cases where the control allocation system has failed to find a solution but the large majority of cases still perform well. It is noted that the average number of iterations required to find solutions has actually dropped. This is due to the fact that the results for the failed optimisations cannot be relied upon to accurately give performance data since constraints may be violated so these cases are therefore left out of the calculations for average iterations. Once again, system configuration two is successful in reducing the adverse forces while having minimal negative effects on the results, as shown in figure C.1 from appendix C. Taking into account adverse forces (configuration 2) has no negative effects on the error distribution of the failure cases and only requires a few more iterations on average to find solutions.

The VSA UAV performance has suffered to a larger extent, with a clear trend toward larger errors. There are still a number of failure cases that perform well but the aircraft is simply not as capable as the Modular UAV with respect to actuator redundancy. Taking into account adverse forces (configuration 2) clearly has a negative effect, with a number of failure cases from error slots between 5% and 20% moving to the 20% to 50% error slot. Figure C.4 from appendix C shows that no benefit is apparent for configuration 2 for the VSA UAV. This is not unexpected since the more extreme failures have used up a larger portion of the redundancy available to the aircraft. The drop in the average number of iterations is explained by the fact that failure cases that are not solved are not taken into account in the averages.

Table 6.22: Control allocation performance overview for failure category 2

	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	57	57	Modular UAV
error < 5%	49 (86)	49 (86)	
5% < error < 10%	0	0	
10% < error < 20%	0	0	
20% < error < 50%	0	0	
50% < error	0 (0)	0 (0)	
No solution	8 (14)	8 (14)	
Average # of Iterations	14.84	17.94	
	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	43	43	VSA UAV
error < 5%	21 (49)	20 (47)	
5% < error < 10%	3 (7)	0	
10% < error < 20%	7 (16)	0	
20% < error < 50%	3 (7)	10 (23)	
50% < error	1 (2)	4 (9)	
No solution	8 (19)	8 (19)	
Average # of Iterations	23.25	17.58	

Modular UAV

The case-by-case performance of the system for the modular UAV is shown in figure 6.5 and it is clear that the system performs well for all cases except for elevator failures which is consistent with the results obtained for the soft failure category. The only change here is that the more severe elevator deflection angles have proven to result in cases where no solution is found due to the fact that the aircraft can no longer be trimmed. The excellent redundancy of the Modular UAV for both the roll and yaw virtual actuator is clearly evident from these results, as even deflections at very large angles are compensated for without difficulty. No specific failure cases are discussed in more detail.

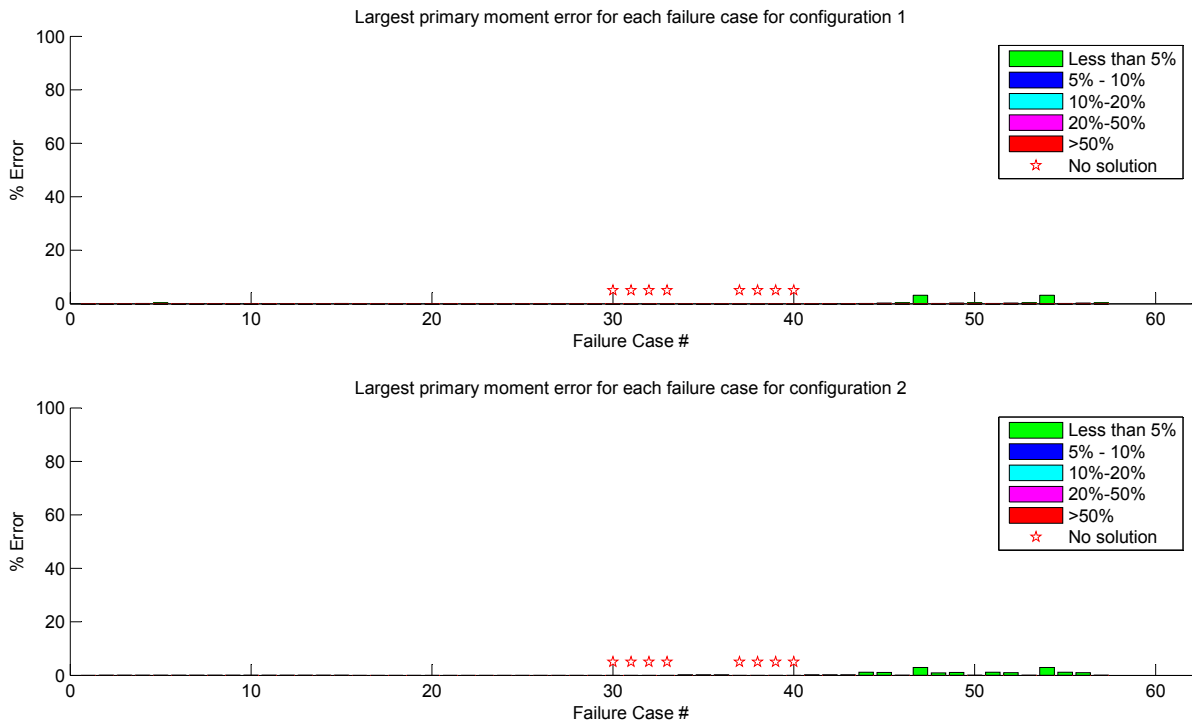


Figure 6.5: A plot of the largest primary moment errors for each failure case for failure category 2 (Modular UAV)

VSA UAV

Figure 6.6 shows the percentage errors for each failure case for the VSA UAV. The failures that are problematic are still the same as those for failure category 1 with a few more failure cases that cannot be solved for. Overall, nothing that is inconsistent with the results from the previous failure category occurs and no individual failure cases are discussed.

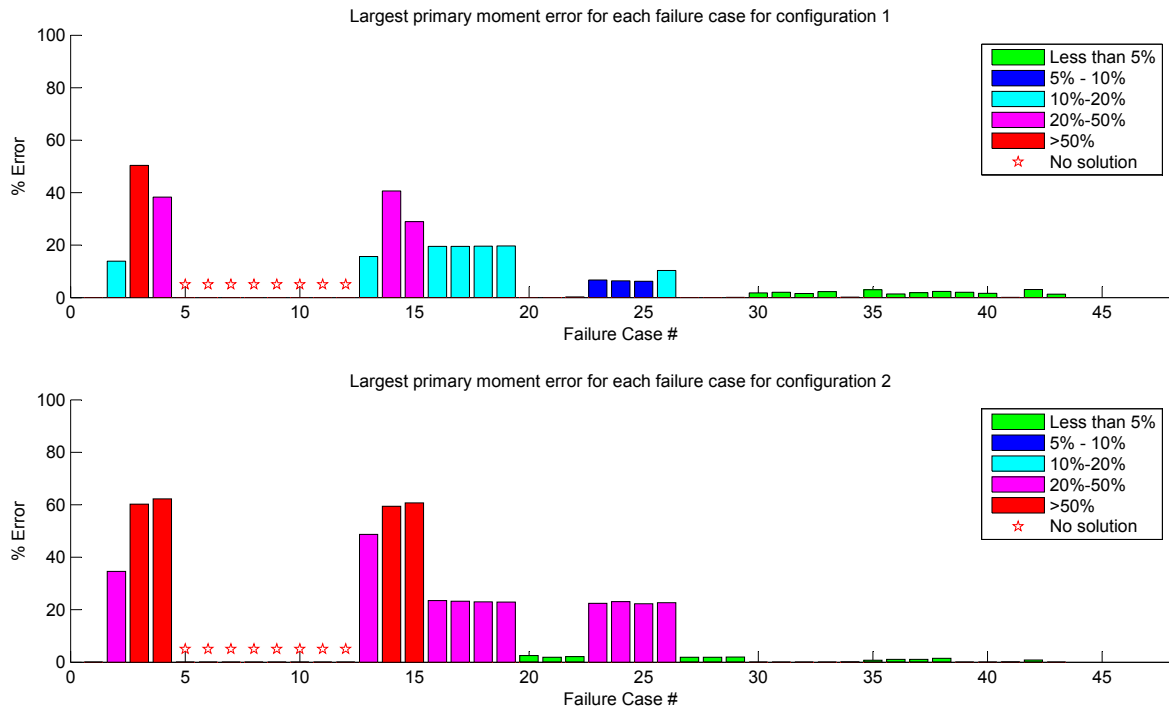


Figure 6.6: A plot of the largest primary moment errors for each failure case for failure category 2 (VSA UAV)

6.3.4 Multiple actuator soft failures

The first test suite with multiple actuator failures is now considered. The nature of these failures is described in section 6.1.3 and appendices B.1 and B.2 provide more detail on the individual failure cases used.

Overall system performance

The overall system performance for both aircraft for multiple actuator failures is given in table 6.23. This table shows the expected shift of more failure cases into the higher error slots as control redundancy is lost due to multiple actuator failures. The Modular UAV still performs well, even when more than one actuator is failed at a time while the VSA UAV clearly has more difficulty with multiple actuator failures. While the Modular UAV still has good performance for the majority of cases, the average number of iterations to find a these solutions has increased. This increase is not evident in the VSA UAV, which takes a fairly consistent number of iterations between failure categories. The reason for this is that, for the single-actuator failures, the Modular UAV only needed to reconfigure a single virtual actuator since its actuators perform a single role by default. For multiple actuator failures, however, it often has to reconfigure multiple virtual actuators. The

VSA UAV on the other hand had to reconfigure multiple virtual actuators even for single actuator failures since actuators perform multiple roles by default for this aircraft.

Taking into account adverse forces (configuration 2) now has a negative effect on the performance of the control allocation system for both the Modular UAV and the VSA UAV. While the Modular UAV did not have any problems reducing adverse forces for failure categories 1 and 2, the reduced actuator redundancy caused by the multi-actuator failures has resulted in a larger number of failure cases having large errors when adverse forces are taken into account for this failure category. The VSA UAV has struggled from the first failure category to reduce adverse forces and this does not change for this failure category. The figures showing the effects on primary force and moment errors and adverse forces for the Modular UAV (figure C.2), and the VSA UAV (figure C.5), can be found in appendix C. These figures show that on average, the Modular UAV is still capable of reducing adverse forces while the VSA UAV shows the same trends as before.

Table 6.23: Control allocation performance overview for failure category 3

	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	365	365	Modular UAV
error < 5%	273 (75)	221 (61)	
5% < error < 10%	5 (1)	19 (5)	
10% < error < 20%	8 (2)	11 (3)	
20% < error < 50%	59 (16)	68 (19)	
50% < error	6 (2)	4 (1)	
No solution	14 (4)	42 (12)	
Average # of Iterations	32.14	48.99	
	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	144	144	VSA UAV
error < 5%	43 (30)	29 (20)	
5% < error < 10%	4 (3)	14 (10)	
10% < error < 20%	4 (3)	3 (2)	
20% < error < 50%	31 (22)	11 (8)	
50% < error	23 (16)	45 (31)	
No solution	38 (27)	41 (29)	
Average # of Iterations	21.5	25.83	

Detailed analysis

Similarly to the two single-actuator failure categories discussed previously, figures similar to figure 6.3 and figure 6.4 were generated for the multiple actuator failure categories. However, due to the large numbers of test cases run, these figures become unwieldy to put into print and they are therefore not included in the main body of the text. Low resolution versions of the figures (figures D.1 and D.3) are given in appendix D and provide an overview of general trends.

Modular UAV

The general trends for multiple actuator failures for the Modular UAV once again show the poor pitch-moment redundancy of the aircraft relative to the roll and yaw redundancy.

Table 6.24: Trim angles for failure case 1 (Modular UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	2.21	0.0352	0.0321
Nominal	2.24	0	0

Almost all of the groups of failure cases that were problematic had an elevator failed at some angle, with the exception being failures of both of the rudders, which also resulted in some cases with significant errors. Two failure cases are considered in more detail: a dual aileron failure which demonstrates the excellent roll redundancy of the Modular UAV and a dual rudder failure which requires differential thrust in order to maintain a viable virtual yaw actuator.

Modular UAV Failure case 1: Failed left and right ailerons at 0°

This failure case illustrates the excellent roll-moment redundancy available to the Modular UAV. Both ailerons are failed at zero degrees and the control allocation system is still able to generate a virtual roll actuator that performs almost exactly the same as the nominal virtual actuator. Table 6.24 shows the trim angles for this failure case and table 6.25 shows the trim settings for the actuators. The dual aileron failure does not significantly alter the trim settings of the aircraft.

Table 6.26 shows the mixing gains for the reconfigured virtual roll actuator and it can be seen that differential flaps are used to generate the virtual roll actuator. A fairly large amount of virtual roll command is also sent to the rudders, which are used to match the previous adverse effects, primarily adverse yaw and adverse side-force.

Table 6.27 shows the errors for the reconfigured virtual roll command and these errors are clearly successfully minimised. The adverse yaw error is significant.

Table 6.25: CA-selected actuator biases for failure case 1 (Modular UAV)

Actuator biases								
(deg)								
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	0	0	0.0177	-0.00635	-4.24	-4.25	-0.0298	-0.0183
Nominal	0	0	0	0	-4.27	-4.27	0	0

(%)		
	δ_{tl}	δ_{tr}
CA	9.25	9.44
Nominal	9.3604	9.3604

Table 6.26: CA-selected mixing vectors for failure case 1 (Modular UAV)

Actuator usage gains									
(% of δ mixed to actuator)									
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_A	CA	0	0	-141.0	141.0	-3.36	3.36	-25.1	-25.1
	Nominal	100	100	0	0	0	0	0	0

		δ_{tl}	δ_{tr}
δ_A	CA	0.93	-0.93
	Nominal	0	0

Table 6.27: Force and moment errors for failure case 1 (Modular UAV)

	Moment error	Adverse yaw error
	(%)	(rad. s^{-2} per 5° command)
δ_A	0.0804	0.037

Modular UAV Failure case 190: Dual rudder failure with both rudders failed at 0°

This failure case deals with a dual rudder failure on the Modular UAV. The control allocation system must now generate a virtual yaw command without the use of either of the primary yawing actuators. There are two possibilities left to the aircraft, differential thrust and differential aileron, flap and elevator deflections. Looking at the trim results, shown in table 6.28 and table 6.29, it is clear that the trim values are not greatly affected by this failure, although, if the results are compared to those in table 6.2 from the nominal aircraft case, it can be seen that the elevators are used to a greater extent to compensate for the loss of pitch that the rudders previously provided.

The mixing vectors for the virtual yaw actuator are shown in table 6.30. There are fairly large gains on all of the remaining aerodynamic actuators. The actuator effectiveness coefficients are shown in appendix A.1 and according to these, deflecting the actuators as shown in table 6.30 will produce a yaw moment. The majority of the yaw moment is however provided using differential thrust, shown by the large differential gains applied to the engines.

The errors for the reconfigured virtual yaw command are given in table 6.31 and show that the yaw error is not zero, although it is quite small given that both rudders have failed. Differential thrust should be capable of reducing this error further if the control allocation system is adjusted further for a specific aircraft as the current weightings are not tuned. Specifically, the engine effectiveness normalisation needs more attention. The adverse side-force error actually indicates a smaller side-force in this instance. The side-force error is based on the side-force of the nominal command and the lack of rudder use for yaw results in a smaller side-force.

Table 6.28: Trim angles for failure case 190 (Modular UAV)

	α	β	ϕ
	(deg)	(deg)	(deg)
CA	2.21	0.159	0.108
Nominal	2.24	0	0

Table 6.29: CA-selected actuator biases for failure case 190 (Modular UAV)

		Actuator biases							
		(deg)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA		-0.0256	-0.0159	0.0274	-0.0165	-4.24	-4.26	0	0
Nominal		0	0	0	0	-4.27	-4.27	0	0

		(%)	
		δ_{tl}	δ_{tr}
CA		9.06	9.62
Nominal		9.3604	9.3604

Table 6.30: CA-selected mixing gains for failure case 190 (Modular UAV)

		Actuator usage gains							
		(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_R	CA	-88.4	-88.5	-110.0	109.0	-215.0	215.0	0	0
	Nominal	0	0	0	0	0	0	100	100

		δ_{tl}	δ_{tr}
δ_R	CA	-37.9	37.9
	Nominal	0	0

Table 6.31: Force and moment errors for failure case 190 (Modular UAV)

	Moment error	Adverse side-force error
	(%)	(m.s ⁻² per 5° command)
δ_R	17.4	0.067

VSA UAV

Looking at the results for the VSA UAV, there are very few types of multi-actuator failures that are properly recoverable for this aircraft. The effectiveness of the actuators simply does not allow the aircraft performance to be fully recovered if either of the ailerons are failed with any combination of other actuators. The only failures that are consistently recoverable for this aircraft, with errors less than 40%, in the category being discussed are elevon and rudder failures. These failures can be recovered from due to the fact that the ailerons provide more powerful roll and pitching ability than the elevons and the ailerons and elevons provide yaw redundancy to compensate for failed rudders. A combination elevator and rudder failure is described below.

VSA UAV failure case 57: Failed right elevon at 0° and failed left rudder at 5°

Tables 6.32 and 6.33 show the trim results for this failure case. The right rudder is used to cancel the yaw caused by the failed left rudder. The ailerons and the remaining elevon are used to trim the aircraft longitudinally. Intuitively, it would make more sense to trim using only the ailerons and leave the left elevon at zero. The control allocation system, however, assigns gains to all effective actuators and this results in an unusual trim configuration. The force and moment errors are still zero but adding further constraints or weighting the actuators differently should be investigated to force the algorithm to generate more conventional results.

Since the elevons are used for both pitch and roll, the loss of the elevon necessitates the reconfiguration of both the virtual roll and the virtual pitch actuators. The virtual roll actuator is now realised using larger gains on the left and right ailerons with the right rudder used to match the adverse forces for the nominal virtual roll command. The virtual pitch actuator uses both the ailerons as well as the remaining elevon to generate the required pitch moment and the rudder is used to cancel the adverse effects caused by the asymmetric deflections. The virtual yaw actuator is now generated using a gain of approximately 200% for the remaining rudder, with the ailerons and left elevon used to cancel adverse effects.

Table 6.35 shows the force and moment errors for the three reconfigured virtual actuators. The virtual roll and virtual yaw actuators clearly perform well with respect to their primary moment errors.

Table 6.32: Trim angles for failure case 57 (VSA UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	3.01	0.769	0.388
Nominal	1.99	0	0

Table 6.33: CA-selected actuator biases for failure case 57 (VSA UAV)

	Actuator biases						
	(deg)						(%)
	δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
CA	4.8	-6.6	-2.56	0	5	-5.08	19.6
Nominal	4.314	-4.314	0	0	0	0	20.34

Table 6.34: CA-selected mixing vectors for failure case 57 (VSA UAV)

	Actuator usage gains							
	(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
δ_A	CA	143.0	143.0	-0.608	0	0	-17.1	0
	Nominal	100	100	-100	100	0	0	0
δ_E	CA	-126.0	163.0	73.3	0	0	28.0	0
	Nominal	-100	100	100	100	0	0	0
δ_R	CA	7.09	-2.06	13.1	0	0	198.0	0
	Nominal	0	0	0	0	0	100	100

Table 6.35: Force and moment errors for failure case 57 (VSA UAV)

	Moment errors	Adverse moment errors			Adverse force errors	
	(%)	(rad.s ⁻² per 5° command)			(m.s ⁻² per 5° command)	
		Roll	Pitch	Yaw	Side	Lift
δ_A	1.03			0.014	0.15	
δ_E	1.71			0.076		0.625
δ_R	0.77	0.0264				

6.3.5 Multiple actuator extreme failures

The most extreme failure category is now considered. The nature of the extreme multi-actuator failures is described in section 6.1.3 and appendices B.1 and B.2 describe the individual failure cases considered. The overall performance of the system is described followed by some more detail on each aircraft.

Overall system performance

Table 6.36 shows the overall performance of the control allocation system for both test aircraft for this failure category. The largest change from the previous category for the Modular UAV is the increase in the percentage of failure cases that cannot be solved. Taking into account adverse forces now results in over a quarter of the test cases becoming unsolvable. This indicates that the Modular UAV actuator redundancy is now being taxed more heavily. The overall performance of the VSA UAV, although worse than for failure category 3, is not significantly different. This is due to the fact that all of the failure types that are problematic for this UAV were already severe enough for the previous failure category so that solutions were difficult to achieve. A small number of cases from the failure types that were previously handled well are now more problematic but this does not significantly alter the error distribution.

The figures comparing the effects of configuration 1 and 2 with respect to the primary force and moment errors and the adverse forces can be seen in C, where figure C.3 shows the effects for the Modular UAV and figure C.6 shows the effects for the VSA UAV. The results remain consistent, with the Modular UAV being capable of reducing adverse forces while the VSA UAV is not.

Table 6.36: Control allocation performance overview for failure category 4

	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	453	453	Modular UAV
error < 5%	311 (69)	256 (57)	
5% < error < 10%	20 (4)	20 (4)	
10% < error < 20%	12 (3)	8 (2)	
20% < error < 50%	24 (5)	31 (7)	
50% < error	15 (3)	16 (4)	
No solution	70 (15)	117 (26)	
Average # of Iterations	34	55.65	
	System configuration 1	System configuration 2	
	# (%)	# (%)	
# Cases	177	177	VSA UAV
error < 5%	48 (27)	29 (16)	
5% < error < 10%	0	18 (10)	
10% < error < 20%	3 (2)	2 (1)	
20% < error < 50%	28 (16)	18 (10)	
50% < error	33 (19)	45 (25)	
No solution	64 (36)	64 (36)	
Average # of Iterations	26.7	20.62	

Detailed analysis

Once again, the performance of the control allocation system is evaluated on a case-by-case basis using figures similar to figure 6.3. These figures are once again too large to place in the text but were used to identify the trends for the Modular UAV and the VSA UAV with respect to this failure category. The low resolution versions of these images are given in appendix D, where figure D.2 gives an overview of the trends for the Modular UAV and figure D.4 gives an overview of the trends for the VSA UAV.

Modular UAV

The cases that are problematic for the modular UAV for multiple actuator extreme failures remain consistent with what is seen previously. Elevator failures perform poorly while

most other combinations of actuator failures are at least partially recoverable with many of them performing very well. A test case where four actuators have failed is considered in more detail below.

Modular UAV failure case 452: Four failed actuators

This failure case fails multiple actuators, one primary actuator from each of the three moment-generating virtual actuators and one flap. This case illustrates the ability of the control allocation system to handle a large number of actuator failures at once. Additionally, since all three moment-generating virtual actuators need to be reconfigured, it is possible that different virtual actuators may use large deflections of the same actuator so that if these commands were to be issued simultaneously, saturation of the actuator may occur sooner than expected.

Firstly, the trim results are given in tables 6.37 and 6.38. The trim results are as expected, with the remaining elevator used to a large extent for pitch trim. The rudder is used to counter the yaw moment caused by the large deflection of the right elevator and the right aileron and left flap are used to generate lift and to counter the roll moment caused by the large deflection of the elevator. A small amount of differential thrust is also used to remove some yaw moment.

The virtual actuator mixing gains are given in table 6.39. Large gains are assigned to the remaining actuators, as expected since these actuators must be used to compensate for the failed actuators. The virtual roll command is realised using the remaining aileron and flap while the rudder is used to reduce adverse yaw. The virtual pitch command is generated using the right elevator, while the remaining rudder, aileron and flap are used to reduce adverse yaw and adverse roll. The virtual yaw actuator is generated using the remaining rudder, with the elevator used to reduce adverse pitch errors.

Differential thrust is used for all three virtual actuators. The gain is small for the virtual yaw actuator although it is quite large for the virtual pitch actuator. For these two virtual actuators, it is used to reduce adverse yaw. For the virtual yaw actuator, some differential thrust is used to provide the last portion of required yaw as the remaining rudder gain cannot be increased further without violating the position constraints.

Looking at the elevator and rudder usage for the virtual pitch and virtual yaw commands, it is clear that both of these virtual actuators now make use of fairly large gains for these actuators. Issuing a large pitch command to the virtual pitch actuator will therefore limit the effectiveness of the virtual yaw actuator and vice versa. This failure case shows that for certain failure scenarios, different virtual actuators require large deflections of

Table 6.37: Trim angles for failure case 452 (Modular UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	1.97	0.676	-0.0431
Nominal	2.24	0	0

Table 6.38: CA-selected actuator biases for failure case 452 (Modular UAV)

	Actuator biases							
	(deg)							
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	0	1.11	1.39	0	0	-7.19	5.33	0
Nominal	0	0	0	0	-4.27	-4.27	0	0

	(%)	
	δ_{tl}	δ_{tr}
CA	10.1	8.61
Nominal	9.3604	9.3604

the same actuators so that issuing these virtual commands simultaneously can result in premature saturation of the actuators.

The resulting errors for the virtual actuators are given in table 6.40. The errors once again show that the virtual pitch actuator is the most problematic and it has a larger error than the other two virtual actuators. The primary errors for the virtual roll and yaw actuators are small while the virtual pitch actuator error is slightly bigger although still acceptable. The virtual roll actuator has an adverse yaw error that may affect performance while the virtual pitch actuator has an adverse lift error and the virtual yaw actuator has an adverse roll error. These adverse errors are all within the ranges where they begin to affect the aircraft.

Table 6.39: CA-selected mixing vectors for failure case 452 (Modular UAV)

		Actuator usage gains							
		(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_A	CA	0	126.0	-100.0	0	0	-5.07	-13.5	0
	Nominal	100	100	0	0	0	0	0	0
δ_E	CA	0	5.06	17.9	0	0	156.0	-39.0	0
	Nominal	0	0	0	0	100	100	0	0
δ_R	CA	0	-0.626	0.766	0	0	41.4	193.0	0
	Nominal	0	0	0	0	0	0	100	100

		δ_{tl}	δ_{tr}
δ_A	CA	4.02	-4.02
	Nominal	0	0
δ_E	CA	-48.7	48.7
	Nominal	0	0
δ_R	CA	-22.45	22.45
	Nominal	0	0

Table 6.40: Force and moment errors for failure case 452 (Modular UAV)

	Moment errors	Adverse moment errors			Adverse force errors	
	(%)	(rad.s ⁻² per 5° command)			(m.s ⁻² per 5° command)	
		Roll	Pitch	Yaw	Side	Lift
δ_A	1.63			0.022		
δ_E	16.5					0.084
δ_R	2.36	0.0099				

VSA UAV

The general trends for the VSA UAV for multiple actuator extreme failures are consistent with the results for the previous failure categories with primarily elevon and rudder failures performing well. Almost all failure cases involving aileron failures resulted in very large errors or no solution. One failure case where an actuator fails at a large angle while the aircraft performance can be maintained is shown below.

VSA UAV failure case 166: Failed right elevon at -12.5° and failed left rudder at 0°

This failure is very similar to the failure shown for the soft failure category for this aircraft. In this case however, the right elevon is failed at 12.5° and the left rudder is failed at 0° .

The trim settings for this failure are given in tables 6.41 and 6.42. It is immediately clear why this particular extreme failure is handled so well even though the UAV has only moderate redundancy. The right elevon is failed at just the right angle so that the aircraft can be trimmed using the two elevons and a larger-than-normal angle of attack. The left and right ailerons are then almost unused and this leaves them free to generate the virtual roll and pitch actuators.

Table 6.43 shows the reconfigured virtual roll, pitch and yaw actuators. The virtual roll actuator is realised using the two ailerons, with the rudder reducing adverse effects errors. The virtual pitch actuator constructed using the two ailerons as well as the remaining elevon and the virtual yaw actuator using a large gain on the remaining rudder, with the ailerons and remaining elevon used to reduce adverse effects.

The resulting force and moment errors for the reconfigured virtual actuators are given in table 6.44 and show that most of the errors are very small. There are a few significant adverse effects errors but the largest of which are the adverse pitch and adverse lift errors for the virtual pitch actuator.

Table 6.41: Trim angles for failure case 166 (VSA UAV)

	α (deg)	β (deg)	ϕ (deg)
CA	3.75	0	0
Nominal	1.99	0	0

Table 6.42: CA-selected actuator biases for failure case 57 (VSA UAV)

	Actuator biases						δ_t (%)
	δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	
CA	-0.0349	0.0517	-12.5	-12.5	0	0.0131	19.6
Nominal	4.314	-4.314	0	0	0	0	20.34

Table 6.43: CA-selected mixing vectors for failure case 57 (VSA UAV)

	Actuator usage gains							
	(% of δ mixed to actuator)							
		δ_{al}	δ_{ar}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}	δ_t
δ_A	CA	141.0	141.0	-0.328	0	0	-19.3	0
	Nominal	100	100	-100	100	0	0	0
δ_E	CA	-142.0	163.0	41.8	0	0	16.4	0
	Nominal	-100	100	100	100	0	0	0
δ_R	CA	6.58	-1.49	12.2	0	0	197.0	0
	Nominal	0	0	0	0	0	100	100

Table 6.44: Force and moment errors for failure case 57 (VSA UAV)

	Moment errors	Adverse moment errors			Adverse force errors	
	(%)	(rad. s^{-2} per 5° command)			(m. s^{-2} per 5° command)	
		Roll	Pitch	Yaw	Side	Lift
δ_A	0.243				0.16	
δ_E	6.01		0.73			0.84
δ_R	0.29	0.052				

6.3.6 Other failure cases

In order to run the large numbers of failure cases that were considered in the above sections, the process of generating failures and running the optimisation to find solutions was largely automated. There are some failure conditions that were not included in the above test suite due to this automation process and these failures warrant a brief mention. The types of failures specifically not included are engine failures and changes in the aircraft parameters with no actuator loss. Since the VSA UAV only has one engine there is no solution for an engine failure for this aircraft. Engine failures are therefore only considered for the Modular UAV, which has two engines. Failures that involve changes in the aircraft parameters without failing actuators were not tested.

Engine failure

A failure case is specifically generated to illustrate the asymmetric trimming configurations discussed in section 3.4.1. The failure case is as follows: The healthy configuration of the Modular UAV is used, with the left engine effectiveness set to zero. The failure case is first solved using a value of $\lambda = 0.5$, which places equal weight on the bank angle and side-slip angles. The gain is then shifted to $\lambda = 1$ and $\lambda = 0$ to see what the effects are.

The results for the trim settings for $\lambda = 0.5$ are shown in tables 6.45 and 6.46, where the trim angles and trim actuator settings are given. When the two angles are given equal weight in the cost function, the control allocation system clearly favours using some bank angle to offset the side-forces produced by the rudders, which are used to offset the yaw caused by the right engine, which is set to double its normal thrust setting to compensate for the failed engine.

Table 6.45: Trim angles for the Modular UAV test aircraft for $\lambda = 0.5$

	α (deg)	β (deg)	ϕ (deg)
CA	2.89	-0.45	2.25
Nominal	2.2429	0	0

Looking at the mixing vectors for the virtual actuators, given in table 6.47, it is clear that only the virtual thrust actuator is affected, as is expected. The virtual thrust actuator clearly now assigns all of the virtual thrust command to the remaining engine and the aerodynamic actuators are used to reduce the adverse yaw caused by this thrust differential. This case illustrates the need to take actuator dynamics into account somewhere in

Table 6.46: CA-selected biasing vectors for a left engine failure (Modular UAV)

Actuator biases								
(deg)								
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	2.5	-0.59	-0.533	-3.34	-2.52	-6.42	-7.42	-7.51
Nominal	0	0	0	0	-4.27	-4.27	0	0

(%)		
	δ_{tl}	δ_{tr}
CA	0	18.7
Nominal	9.3604	9.3604

Table 6.47: CA-selected mixing vectors for a left engine failure (Modular UAV)

Actuator usage gains									
(% of δ mixed to actuator)									
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
δ_T	CA	31.9	31.9	45.3	-45.3	-3.25	3.25	6.34	6.34
	Nominal	0	0	0	0	0	0	0	0

		δ_{tl}	δ_{tr}
δ_T	CA	0	100
	NOM	50	50

the system, as discussed in section 3.4.4. The actuators will clearly respond much more quickly than the engines.

The optimisation is run again with $\lambda = 1$ and the trim angles are shown in table 6.48. Changing the value of λ clearly has an effect, with a larger side-slip angle and a smaller bank-angle being used to trim the aircraft, although the bank angle is not reduced to zero.

Looking at the final setting, with $\lambda = 0$, the trim angles are shown in table 6.49. These trim angles are not significantly different from those generated for $\lambda = 1$. Clearly, the settings for λ do not give the exact configurations discussed in section 3.4.1. The concept of using differing amounts of bank angle and side-slip angle for asymmetric trim based on penalising these angles is however shown to be sound given the change in trim angles

Table 6.48: Trim angles for the Modular UAV test aircraft for $\lambda = 1$

	α (deg)	β (deg)	ϕ (deg)
CA	3.03	-2.42	1.79
Nominal	2.2429	0	0

between $\lambda = 0.5$ and $\lambda = 1$. With further tuning of the weights and normalisation parameters for the control allocation system, finer control over these asymmetric flight configurations may be possible.

Table 6.49: Trim angles for the Modular UAV test aircraft for $\lambda = 0$

	α (deg)	β (deg)	ϕ (deg)
CA	2.99	-2.32	1.82
Nominal	2.2429	0	0

6.3.7 Control allocation system configuration 3

Configuration 3 of the control allocation system is described at the start of this chapter. This configuration applies a penalty on actuator usage by penalising mixing vector gains. The results for this configuration were consistent throughout the failure categories that were investigated and followed the trend that was expected of this configuration. The results for this configuration are not given in detail but the general trend is described below.

The primary force and moment errors for all failure cases were increased due to the fact that the weight assigned to the force and moment goal was reduced and a penalty was placed on mixing gains. For the average gains of the mixing vectors, a large reduction was evident.

These results were obtained using the same weighting applied to all of the mixing gains. Although not tested, it is expected that this goal of the control allocation system will be more useful if individual actuators are penalised. An example of where this could prove useful is the virtual pitch actuator of the Modular UAV which was shown to use the rudders to provide a pitch moment. By assigning a large penalty to rudder usage for the

virtual pitch actuator, the control allocation system will avoid using the rudders where possible, thus resulting in a more conventional virtual actuator. Further investigation of this goal is required.

6.3.8 Conclusion

General trends

Having run such a large number of failure cases across a number of different failure categories and control allocation system configurations, it is possible to highlight the general trends that are evident in the performance of the system. The largest trend that is evident from the results is the effect of actuator redundancy on the performance of the system. This is shown between aircraft as well as for specific aircraft. The modular UAV is clearly more suited to control allocation than the VSA UAV. The control allocation system performs better in all of the failure categories for the Modular UAV and it is also capable of meeting the further objective of reducing adverse forces for this UAV. The control allocation system is not capable of significantly reducing adverse forces for the VSA UAV and performs better if adverse forces are ignored for this aircraft. The higher level of actuator redundancy of the Modular UAV clearly allows for more objectives to be considered in the control allocation process. Having said that, the virtual pitch actuator redundancy of the Modular UAV could do with improvement as the pitch redundancy of the aircraft was consistently shown to be the problem area for all of the failure categories. The following trends are evident with respect to what causes difficulties for the control allocation system:

- Failures that are so severe that the aircraft cannot be trimmed - These failures are unrecoverable and no solution exists for the control allocation system to find. When these types of problems are encountered, the optimisation is unable to satisfy the equality constraints that are used to trim the aircraft and the optimisation fails.
- Failures that can be trimmed but the achievable moments are negligible - These failures are not so severe that the aircraft cannot be trimmed, but the act of trimming the aircraft uses up all of the available actuator throw for one or more of the rotational moments. In these cases, the aircraft can be trimmed by the system but one or more of the virtual roll, pitch or yaw actuators will not be recoverable.
- Less severe extreme failures - In these cases, the aircraft is trimmed and at least some throw is left on all of the virtual actuators although performance is severely reduced so that control system reconfiguration would probably become necessary.

As can be seen from the failure cases that have been highlighted, these problems can occur in a number of situations. Which of these situations is likely to become problematic for a particular aircraft depends upon the actuator redundancy available to the aircraft.

The following strengths of the control allocation system are evident from the results shown:

- Ability to handle a variety of aircraft and aircraft configurations with no alteration to the control allocation system itself.
- Ability to handle both single and multiple actuator failures.
- Ability to handle both aerodynamic actuator failures as well as engine failures.
- Ability to handle a large number of failure cases using only knowledge of the actuator effectiveness and force and moment biases.
- It can be used to provide design feedback on the level of actuator redundancy available to an aircraft.

The next chapter introduces a few additional facets of the control allocation problem that can be considered and describes how some of these are implemented and tested.

Chapter 7

Control allocation system expansion

In the previous two chapters, the implementation and testing of the basic control allocation system was described. The system was shown to be effective for a wide variety of failure categories on two different aircraft with varying levels of actuator redundancy. The system was also shown to be adjustable with respect to what goals are considered by testing three different control allocation configurations. There are however still some areas where this system can be improved. Section 3.4.4 mentioned some issues with regard to actuator dynamics. Sections 3.4.1 and 3.4.2 mention the possibility of using the control allocation system for an unconventional aircraft and it was posited that this could be done with minimal alteration to the system. These two issues are considered in this chapter and an attempt is made to address them. In addition to considering these issues, a cursory investigation into the link between the trim optimisation and the force and moment optimisation is carried out. The purpose of this investigation is to establish the extent of the interaction between these two elements of the optimisation and whether it might be beneficial to separate them. Next, a brief overview is given as to how these additions were added to the system and finally, the results of testing these three additional facets of the control allocation problem are described.

7.1 Proposed additions

7.1.1 Actuator dynamics: Frequency-based allocation

This section provides some details of the expansion of the control allocation system to provide different actuator sets for fast or slow actuators and control commands. This enhancement of the system can be used to separate the fast and slow actuators within the control allocation system so that the issues mentioned in section 3.4.4 can more easily

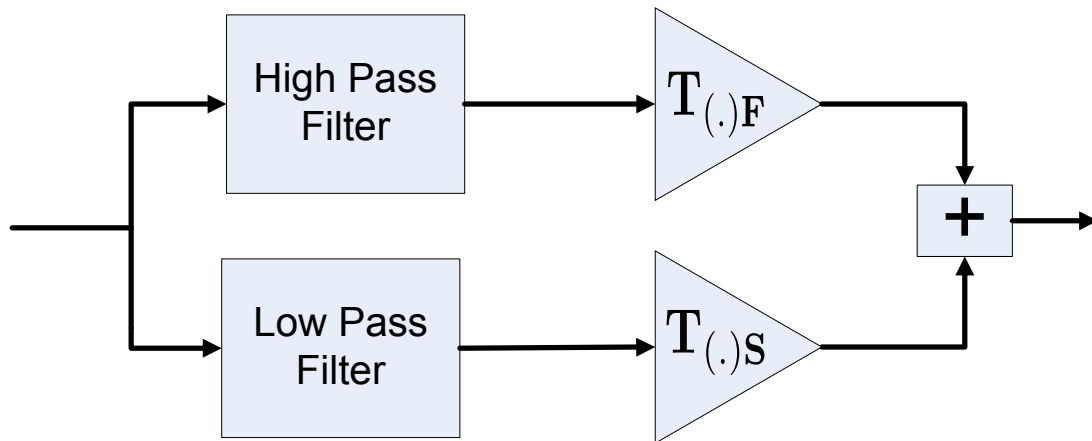


Figure 7.1: Frequency-based allocation with complimentary filters

be handled. There are a number of possible methods of doing this, two of which are discussed here. The first method to be discussed uses weightings applied to actuator usage in order to favour certain actuators over others and the second method alters the coefficients supplied to the optimisation engine for the actuators in order to favour certain actuators over others.

The idea in both cases is that multiple actuator sets are generated for different frequency elements in commands. These sets are then used by the control system through a set of complimentary filters, as shown in figure 7.1, where $\mathbf{T}_{(.)\mathbf{F}}$ and $\mathbf{T}_{(.)\mathbf{S}}$ are the sets of fast and slow mixing vectors.

The mixing vectors that are used to mix virtual command signals to physical actuators are then expanded to include fast and slow actuator sets. For the aircraft considered in this thesis, all of the aerodynamic actuators are considered fast actuators and the engines are considered slow actuators so that frequency allocation could be done merely by using the aerodynamic actuators for the high frequency commands and the engines for the low frequency commands. However, in a general case, the aircraft may have some aerodynamic actuators that are slower than others, a typical example being flaps. For this reason, fast and slow actuator sets are defined for both aerodynamic actuators as well as engines.

Generating frequency-dependent virtual actuators allows the aircraft to perform high-frequency manoeuvres using high-bandwidth actuators and slow or steady-state manoeuvres using the low-bandwidth actuators without the controller needing to make a distinction between the two types of manoeuvres. Furthermore, it will allow fast actuators to return to their default positions (typically zero offset) for steady-state manoeuvres and it will ensure that the control system uses slow actuators to trim the aircraft.

Actuator usage weightings The first method attempted involved selecting the weightings applied to the actuators in the cost function in order to favour certain actuators. The optimisation was run once with the weightings selected so that using slow actuators has a high cost, thus resulting in a fast actuator set. The optimisation was then run again with fast actuator usage resulting in a high cost, generating a actuator set that favours slow actuators.

This method allows the use of all the actuators and thus should not result in any problems in terms of finding solutions that were previously possible. It was discovered, however, that it is difficult to find the correct balance between actuator usage weights and moment optimisation weights, resulting in less than optimal results. Additionally, creating a complete separation between fast and slow actuators is not possible using this method, since even actuators associated with a high cost will be used if their effectiveness is high enough. For this reason, another method was sought to implement the frequency-based allocation method.

Coefficients The second method attempted and the one used in the final implementation involves changing the coefficients sent to the optimisation engine in order to favour certain actuators. This method removes the complexity of adjusting weights between moment and actuator usage goals and allows the algorithm to completely ignore actuators if their coefficients are set to zero. Since the effectiveness of the actuators that are to be ignored for a specific actuator set are set to zero, this method also allows for the complete separation between fast and slow actuators. This method provides consistent results but it is possible that, since actuators are essentially removed from the optimisation, the range of possible solutions is reduced. This can become problematic in certain situations if, for example, there are no actuators to form a slow virtual pitch actuator. The pitch trim command will then be filtered out by the high pass filter and not reach the fast pitching actuator that is available. To solve this issue, care must be taken to ensure that virtual actuators always retain some effectiveness, even if actuators that are categorised as fast are used for slow virtual actuator sets.

Complimentary filters The virtual commands generated by the control system must now be distributed between the fast and slow actuator sets. This is accomplished outside of the control allocation system with the use of complimentary filters. Commands that pass through a high pass filter go to the fast actuator set while commands that pass through a low pass filter go to the slow actuator set. A possible design for these filters is presented by [29] and can be described as follows:

The fast actuator is driven as follows:

$$\delta_F = \delta - x \quad (7.1)$$

while the slow actuator is driven as follows:

$$\delta_S = K_a x \quad (7.2)$$

where $\dot{x} = -\frac{x}{\tau} + \frac{\delta}{\tau}$ and $K_a = \frac{K_1}{K_2}$.

The gain K_a ensures that the net effect of the two actuator sets does not vary as the load of actuation passes from one set to another. K_1 is the effectiveness of the fast actuator set and K_2 is the effectiveness of the slow actuator set in generating moments.

The time constant τ defines the period over which the actuation transfer takes place and it is selected so that it is much greater than the time constant of the slower actuator set.

7.1.2 Unconventional UAV: A ducted-fan UAV

The control allocation system developed in chapter 3 and implemented in chapter 5 has some factors that limit the configuration of aircraft that it can be applied to. The largest limitation is that of the trim constraints which have been developed for a fixed-wing aircraft with its primary thrust axis along the X_B -axis. The applicability of the control allocation system to an unconventional aircraft is investigated in this section.

A ducted-fan UAV, shown in figure 7.2, is used as an unconventional aircraft test case. The ducted-fan UAV being considered is described in detail in Appendix A.3 and only some important points are mentioned in this section.

The UAV is not a fixed-wing UAV and consequently its aerodynamic characteristics are distinct from a fixed-wing UAV. Additionally, its primary thrust axis is in the Z_B -axis. These fundamental differences require that a different set of equations be developed for the trim goals and constraints that are taken into account by the control allocation algorithm. The basic model of the ducted-fan UAV as presented by [29] is analysed and from this, the equality constraints used by the control allocation system are adjusted to fit this aircraft.

Linear Near-hover ducted-fan trim equations

The equations given by [29] for the ducted-fan UAV are generalised for use with the control allocation system, resulting in the following six trim equations:



Figure 7.2: The SLADe ducted-fan UAV

$$\theta_T = \frac{qS(C_{x_0} + \mathbf{C}_{x_\delta} \mathbf{T}_B)}{-mg} \quad (7.3)$$

$$\phi_T = \frac{qS(C_{y_0} + \mathbf{C}_{y_\delta} \mathbf{T}_B)}{mg} \quad (7.4)$$

$$\mathbf{C}_{z_{\delta_t}} \mathbf{T}_{TB} + mg = 0 \quad (7.5)$$

$$qS(C_{l_0} + \mathbf{C}_{l_\delta} \mathbf{T}_B) = 0 \quad (7.6)$$

$$qS(C_{m_0} + \mathbf{C}_{m_\delta} \mathbf{T}_B) = 0 \quad (7.7)$$

$$qS(C_{n_0} + \mathbf{C}_{n_\delta} \mathbf{T}_B) + \mathbf{C}_{n_{\delta_t}} \mathbf{T}_T \mathbf{B} = 0 \quad (7.8)$$

Equations 7.3 and 7.4 are added as goals to the trim goal section and replace the β_T and ϕ_T terms used for the fixed-wing UAVs. Equations 7.5 to 7.8 replace the equality constraints used for the fixed-wing UAVs. These are the only changes required to apply the control allocation system to the ducted-fan UAV and the rest of the force and moment equations for the virtual actuators remain unchanged.

7.1.3 Splitting the optimisation

The control allocation system up to this point has been implemented as a single optimisation problem that must solve the trimming problem, force and moment error minimisation, adverse effects minimisation and control minimisation. The three primary goals: Trimming, force and moment error minimisation and adverse effects minimisation are linked through the position constraints given in section 3.4.6. This link is illustrated graphically in figure 7.3.

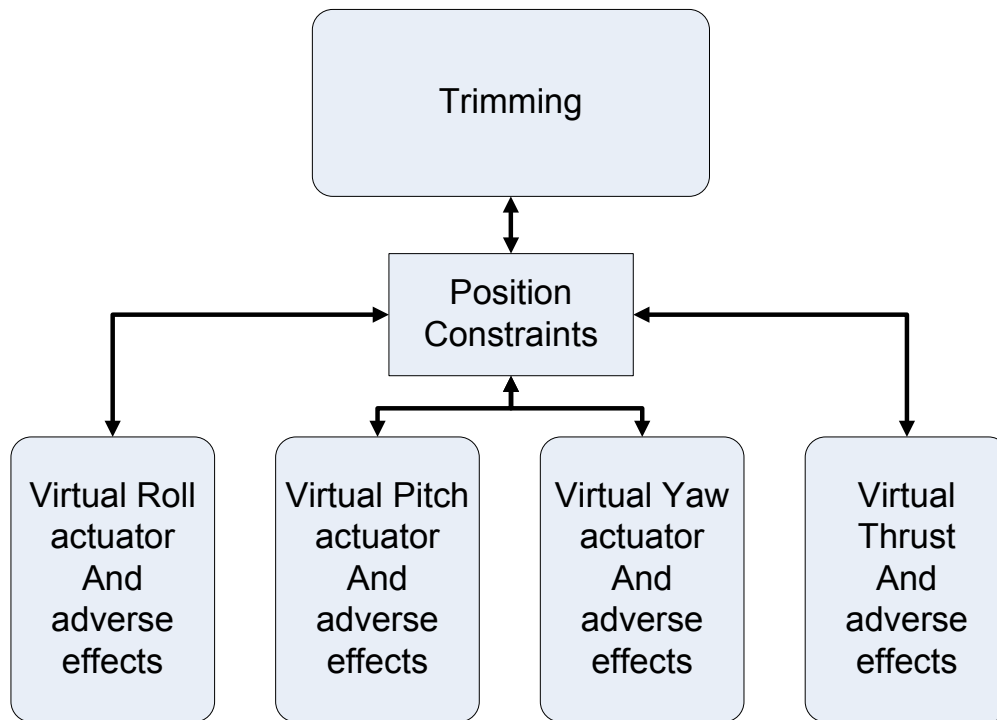


Figure 7.3: Trim bias and moment optimisation link

Splitting this optimisation into two different parts will make it possible to evaluate how strongly linked they are during the optimisation process. The main goal in splitting the optimisation is to determine whether this has any significant effect on the results and on the performance of the optimisation. The following procedure was followed for performing this evaluation: A failure case was run taking into account only the trim bias optimisation. The results from this optimisation were then used to warm-start the default control allocation system for the same failure case to investigate whether the force and moment optimisation requirements will result in the control allocation system reconfiguring the trim biases in order to reach the desired goals of the force and moment optimisation.

A failure case where the failure results in trim biases that are close to saturating an actuator essential to one of the virtual actuators is used to illustrate this process. For failure cases where no essential actuators are close to saturation, the link between the trim and force and moment goals is not expected to manifest.

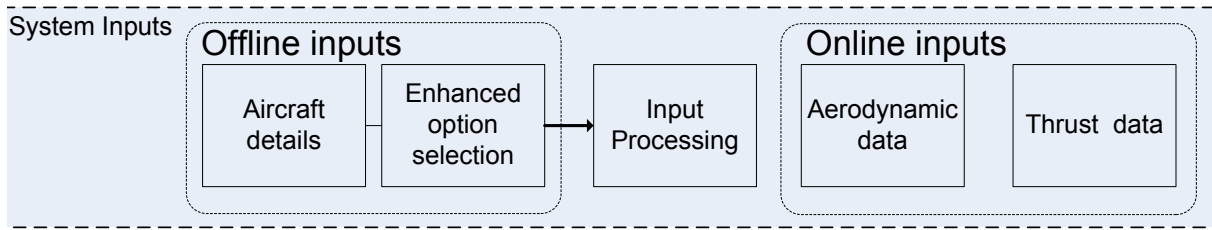


Figure 7.4: Expanded system input layer

7.2 Implementation

The previous section described some interesting additions to the control allocation problem and how these additions might be implemented. This section deals with the practical implementation of these enhancements to the system. In the next section, each of these additions is tested using representative failure cases to illustrate the benefits of these additions.

7.2.1 Method of extension

The basic control allocation system developed in chapter 3 and implemented in chapter 5 achieves the primary goal set out in the introduction to this thesis: A control allocation system that can be used as part of an FTC system that is applicable to a wide range of aircraft configurations and failure categories. The additions discussed in this chapter are some interesting enhancements that can be considered. It was decided, however, that these enhancements should not interfere with the operation of the base system. The enhancements were therefore implemented in such a way that they could be toggled on or off by the user. The end result is that the system can be used in its basic form or with any one of the enhancements discussed in this chapter turned on at a time. This made it possible to assess the effectiveness of each of these enhancements without altering the effectiveness of the base system.

An additional Matlab file was written to encapsulate the extra inputs required by the enhancements. In this file, the user can toggle the various enhancements on or off. The file takes some user inputs in order to toggle the enhancements and some additional data is required for some of the specific enhancements. The input layer to the expanded system is now as shown in figure 7.4. The input processing of the data was altered as well.

Additional user inputs

Additional user inputs are required for the frequency-based allocation. The method requires knowledge of which actuators are considered to be fast actuators and which actuators are considered to be slow actuators. In addition, not all of the virtual actuators on an aircraft will have viable actuators for both fast and slow actuator sets for all of the virtual actuators, the user must identify which virtual actuators are to be considered when generating fast or slow actuator sets. This information is captured in vectors, where the user toggles an actuator as fast or slow and similarly, toggles a virtual actuator as being a candidate for a fast/slow actuator set or not.

Pre-processing

The processing module was altered so that when an enhancement is toggled, the relevant actuator effectiveness coefficients are altered to emphasise the desired actuators.

Equality constraints alterations

For the test involving the unconventional ducted-fan UAV only, the equality constraints were also altered to accommodate this aircraft.

7.3 Testing

Each of the additional abilities added to the control allocation algorithm are tested with a few representative test cases.

7.3.1 Combination SLADe test and frequency allocation test

The adaptation of the control allocation system for use with the unconventional SLADe ducted-fan UAV is tested here. In addition to this, the frequency allocation technique described in section 7.1.1 is also tested, as well as the ability of the algorithm to take into account rotor moments.

Validation of control allocation system for SLADe

Similar to section 6.3.1, the control allocation system is initialised with a zero start vector and control is allocated for the healthy aircraft. Table 7.1 shows that the trim angles all

Table 7.1: Trim angles for the SLADe test aircraft

	θ (deg)	β (deg)	ϕ (deg)
CA	0	0	0
Nominal	0	0	0

Table 7.2: Comparison between nominal and CA actuator biases for the SLADe UAV

	Actuator biases							
	(deg)							
	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	δ_8
CA	0	0	0	0	0	0	0	0
Nominal	0	0	0	0	0	0	0	0

	(%)	
	δ_{t1}	δ_{t2}
CA	39.2	39.2
Nominal	39.24	39.24

remain exactly zero, which is the expected result. Table 7.2 gives the actuator biases and the aerodynamic actuators are all kept at a zero deflection angle and the thrust settings are set to nominal values.

Table 7.3 gives the actuator mixing vectors for the virtual actuators and these values are almost all set to exactly the same values as the nominal manually selected values, with the exception being the virtual yaw actuator. The control allocation system assigns some gain to differential thrust. This behaviour was not fully investigated but taking into account actuator dynamics should resolve this issue. Overall, the results show that the control allocation system is working for the healthy SLADe UAV. Next, a failure situation is tested with a failure of actuator δ_1 at 5° .

Failure of actuator δ_1 at 5°

A failure of actuator 1 at 5° will result in roll, yaw and side-force biases. The control allocation system will therefore need to select biasing vectors that remove these biases. Trimming the aircraft results in a roll angle of approximately 24° , as shown in table 7.4.

Table 7.3: Comparison between nominal and CA selected mixing vectors for the SLADe UAV

		Actuator usage gains (% of δ mixed to actuator)							
		δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	δ_8
δ_A	CA	25.0	-25.0	25.0	-25.0	0	0	0	0
	Nominal	25	-25	25	-25	0	0	0	0
δ_E	CAZ	0	0	0	0	25.0	-25.0	25.0	-25.0
	Nominal	0	0	0	0	25	-25	25	-25
δ_R	CA	-12.7	12.7	12.7	-12.7	-12.7	12.7	12.7	-12.7
	Nominal	-12.5	12.5	12.5	-12.5	-12.5	12.5	12.5	-12.5
δ_T	CA	0	0	0	0	0	0	0	0
	Nominal	0	0	0	0	0	0	0	0

		δ_{t1}	δ_{t2}
δ_A	CA	0	0
	Nominal	0	0
δ_E	CA	0	0
	Nominal	0	0
δ_R	CA	0.18	-0.18
	Nominal	0	0
δ_T	CA	50	50
	Nominal	50	50

Table 7.5 shows the actuator biases that accompany this roll angle and the combination of these results in zero forces and moments acting on the aircraft. The combination of aerodynamic actuator deflections remove the roll and yaw biases but add a significant side-force, which is then removed by holding the ducted-fan UAV at a roll angle of approximately 24° . This roll angle is fairly large and may be too large for the simplified aircraft model used for the ducted-fan UAV to remain valid. This test case, however, illustrates that the control allocation system is capable of handling unconventional UAVs.

The actuator gains for the virtual actuators are given in table 7.6 and the primary as well as the adverse force and moment errors for these mixing vectors were all negligible.

Table 7.4: Trim angles for the SLADe test aircraft

	θ (deg)	β (deg)	ϕ (deg)
CA	0	0	23.99
Nominal	0	0	0

Table 7.5: Comparison between nominal and CA actuator biases for the SLADe UAV

	Actuator biases (deg)							
	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	δ_8
CA	0	2.08	-1.96	0.958	0	0.417	-0.292	-0.708
Nominal	0	0	0	0			0	0

	(%)	
	δ_{t1}	δ_{t2}
CA	39.2	39.2
Nominal	39.2	39.2

Table 7.6: Comparison between nominal and CA selected mixing vectors for the SLADe UAV

		Actuator usage gains (% of δ mixed to actuator)							
		δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	δ_8
δ_A	CA	0	-38.1	38.1	-23.8	14.3	-4.76	4.76	9.53
	Nominal	25	-25	25	-25	0	0	0	0
δ_E	CAZ	0	-4.76	4.76	9.53	14.3	-38.1	38.1	-23.8
	Nominal	0	0	0	0	25	-25	25	-25
δ_R	CA	0	19.9	5.44	-14.5	-21.7	19.9	5.43	-14.5
	Nominal	-12.7	12.7	12.7	-12.7	-12.7	12.7	12.7	-12.7
δ_T	CA	0	0	0	0	0	0	0	0
	Nominal	0	0	0	0	0	0	0	0

		δ_{t1}	δ_{t2}
δ_A	CA	-0.111	0.0911
	Nominal	0	0
δ_E	CA	-0.111	0.091
	Nominal	0	0
δ_R	CA	0.25	-0.237
	Nominal	0	0
δ_T	CA	50	50
	Nominal	50	50

Table 7.7: Fast virtual yaw actuator (Frequency allocation)

		Actuator usage gains							
		(% of δ mixed to actuator)							
		δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7	δ_8
δ_R	CA	-12.5	12.5	12.5	-12.5	-12.5	12.5	12.5	-12.5

Table 7.8: Slow virtual yaw actuator (Frequency allocation)

		Actuator usage gains	
		(% of δ mixed to actuator)	
		δ_{t1}	δ_{t2}
δ_R	CA	-35.8	35.8

Frequency allocated virtual actuators

This UAV is used to illustrate the effectiveness of the frequency allocated virtual actuators. Both the Modular UAV and the ducted-fan UAV have a possible slower virtual actuator in the virtual yaw actuator. Differential thrust for either of these aircraft will result in a yaw moment. For the Modular UAV, the yaw moment is a result of the position offsets of the engines along the Y_B -axis. For SLADe, the yaw moment is a result of the rotor moments caused by the counter-rotating rotors. The results obtained are given in tables 7.7 and 7.8.

The virtual yaw actuator for the fast actuator is now as shown in table 7.7 and this virtual actuator is now the same as the default virtual actuator. No differential thrust is used. The slow virtual actuator now uses only differential thrust and the results for this virtual actuator are shown in table 7.8.

7.3.2 Splitting the optimisation

Failure case 31 from failure category 1 is chosen to show the need to perform the trim optimisation and force and moment optimisation simultaneously. The trim values for the aircraft are first calculated without calculating the mixing gains for the virtual actuators and the results are shown in tables 7.9 and 7.10. Generating a virtual pitch actuator using these trim values to ensure that the pitch command does not saturate the actuators will clearly result in a rather weak virtual pitch actuator since the remaining functional

Table 7.9: Trim angles for the Modular UAV test aircraft

	α (deg)	β (deg)	ϕ (deg)
CA	1.87	0.00421	0.0134
Nominal	2.24	0	0

Table 7.10: Comparison between nominal and CA actuator biases for the Modular UAV

	Actuator biases (deg)							
	δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA	-0.578	0.691	0.562	1.58	0	-12.1	4.89	-0.538
Nominal	0	0	0	0	-4.27	-4.27	0	0

	(%)	
	δ_{tl}	δ_{tr}
CA	9.33	9.36
Nominal	9.3604	9.3604

elevator only has approximately 3° of throw. The failure case is solved for again, solving both the trim and force and moment optimisations simultaneously. The resulting trim values are shown in tables 7.11 and 7.12. The remaining elevator trim angle is clearly reduced and the rudders are used to provide some additional pitch moment to trim the aircraft. The angle of attack has also been reduced and the ailerons and flaps are used symmetrically to generate the needed lift.

Table 7.11: Trim angles for the SLADe UAV test aircraft

	α (deg)	β (deg)	ϕ (deg)
CA	1.34	-0.312	-0.235
Nominal	2.24	0	0

Table 7.12: Comparison between nominal and CA actuator biases for the SLADe UAV

		Actuator biases							
		(deg)							
		δ_{al}	δ_{ar}	δ_{fl}	δ_{fr}	δ_{el}	δ_{er}	δ_{rl}	δ_{rr}
CA		-1.51	1.62	2.27	3.31	0	-9.49	9.88	-5.99
Nominal		0	0	0	0	-4.27	-4.27	0	0

		(%)	
		δ_{tl}	δ_{tr}
CA		10.0	8.68
Nominal		9.3604	9.3604

7.4 Conclusion

Some possible enhancements to the control allocation problem were identified and implemented in this chapter. Extensive testing of these enhancements was not carried out but a few test cases were completed to show the feasibility of these enhancements. The link between the trim optimisation and the force and moment optimisation was also investigated. The modification of the control allocation system for the ducted-fan UAV was shown to be successful while requiring minimal alterations to the system. The trim results for the healthy ducted-fan UAV as well as the results for a failure show that the control allocation system can work for unconventional aircraft. The frequency-based allocation method was shown to be feasible and a more detailed investigation of this facet of control allocation will be beneficial. The link between the trim optimisation and the force and moment optimisation was shown to influence the results for certain failure cases.

Chapter 8

Conclusion and recommendations

8.1 Conclusion

This thesis reported the development of a control allocation system for use in a fault-tolerant control architecture for UAVs. The primary goal of the control allocation system was to minimise the difference in performance between the healthy aircraft and the damaged aircraft while taking adverse forces into account.

A simple 6-DOF aircraft model was developed and this model was used to formulate a control allocation system that can be used for a wide variety of different aircraft configurations. The control allocation formulation was matched to the category of multi-objective constrained optimisation and the goals and constraints identified for the problem were formulated in a fashion that allows the problem to be solved using an SQP optimisation method. The control allocation system was successfully implemented in Matlab using an existing SQP optimisation algorithm and the base system was extensively tested using a variety of control allocation system configurations, aircraft configurations and failure categories. The results showed that the control allocation system is capable of handling a number of very different aircraft configurations with little difficulty and no alteration to the system. A number of trends with respect to the types of solutions found were identified, including: The effect of control redundancy on control allocation performance, the effect of control redundancy on adverse forces minimisation and the strengths and weaknesses of the test aircraft with respect to virtual actuator redundancy. The control allocation system was then extended to include capabilities for handling frequency-based allocation and unconventional aircraft. These extensions were implemented and briefly tested. In addition to these tests, the link between the trim optimisation and force and moment optimisation was also tested. The extended control allocation system was shown to be capable of providing virtual actuators with separated fast and slow actuator sets as

well as being capable of handling unconventional aircraft with minimal alteration. The trim and force and moment optimisations were shown to be linked for actuator failures that result in actuator saturation conditions.

Overall, the control allocation system was shown to be effective with minimal tuning of the parameters which determine its properties. The work completed for this thesis provide the following contributions to UAV research in the ESL lab of the University of Stellenbosch:

- The groundwork has been laid for further research into the various facets of control allocation.
- A control allocation tool for use in a fault-tolerant control architecture for UAVs that can be used for a large variety of different aircraft is presented.
- A control allocation tool that can be used offline to investigate the ability of various aircraft to recover from different kinds of failures by revealing deficiencies in control redundancy is presented.

The control allocation problem, while fairly straight-forward in concept, has a large number of facets with respect to implementation methods and control allocation objectives. Some recommendations for further work and improvements based on the control allocation formulation developed in this thesis are presented in the next section.

8.2 Recommendations

First, some recommendations that have been considered in some detail are presented, including potential methods of implementing these recommendations. A number of further recommendations that have not been studied in detail are presented thereafter.

8.2.1 Detailed recommendations

Three detailed recommendations are given: Calculating trim biases on virtual actuator level, maximising the amount of free play after control allocation and generating high and low resolution virtual actuators.

Calculating trim biases on the virtual actuator level

The current control allocation formulation calculates the trim biases of the actuators for individual actuators. When the control allocation system is practically implemented, issues surrounding smooth transition from one actuator configuration to another will need to be handled. The current method of calculating the trim biases of individual actuators will complicate this smooth transition process as the control system will trim the aircraft during flight using the virtual actuators.

Altering the formulation to calculate the trim biases using the virtual actuators instead should be investigated. Calculating the trim biases using the virtual actuators will make it easier to implement the smooth transition between old and new actuator allocations. The basic procedure would be as follows: The trim optimisation, based on the virtual actuators, will calculate the necessary δ_A , δ_E , δ_R and δ_T virtual deflections necessary to trim the aircraft. These trim deflections will need to change during the course of the optimisation as the effectiveness of the virtual actuators is altered by the force and moment optimisation of the control allocation system. The control allocation system must then ensure that the trim virtual deflections leave enough virtual actuator free-play to meet the desired performance requirements without saturating the physical actuators.

This formulation will conform to the usual method of calculating the trim settings of the aircraft and will make smooth transition between the previous and new virtual actuator configurations simpler. The practical implementation of this formulation may present its own difficulties and the effects that it will have on the types of solutions provided is unknown.

Maximising the amount of available free play after control allocation

The algorithm up to this stage requires the user to select a pre-determined amount of virtual actuator free-play, δ_S , that should still be available after control allocation has taken place. This value can be chosen relatively robustly but it is possible that a particular choice will result in the allocation not finding an optimal solution in certain failure cases. This is specifically a problem in cases where an actuator has failed with a large offset position and simply trimming the aircraft requires most, if not all of the available actuation of another actuator. The result is that while it may be possible to trim the aircraft, the force and moment optimisation will fail to find a solution. The aircraft may still be capable of generating actuator sets that meet the performance requirements of all of the virtual actuators except one. However, since the position constraints for the actuator in question will be violated, the optimisation will fail to find a solution and in this case, all

virtual actuators will suffer due to the premature termination of the optimisation.

The initial inequality constraints ensuring the required amount of freely available virtual actuator were defined in section 3.4.6. If the constants δ_{AS} , δ_{ES} , δ_{RS} and δ_{TS} are replaced with design variables, x_{AS} , x_{ES} , x_{RS} and x_{TS} , and these design variables are then placed in the cost function so that the maximum value for the design variable results in the smallest cost, then the amount of free-play left available after control allocation will be maximised relative to the weightings applied to the new sub-goal that is added to the system. The additional design variables are then added to the cost function through a new goal which is defined as follows:

$$f_{fp}(x) = w_{fp1} \left(\frac{x_{AS} - x_{AS}^d}{x_{AS}^d} \right)^2 + w_{fp2} \left(\frac{x_{ES} - x_{ES}^d}{x_{ES}^d} \right)^2 + w_{fp3} \left(\frac{x_{RS} - x_{RS}^d}{x_{RS}^d} \right)^2 + w_{fp4} \left(\frac{x_{TS} - x_{TS}^d}{x_{TS}^d} \right)^2 \quad (8.1)$$

where $x_{(\cdot)S}^d$ is a user-selected desired level of virtual actuator free-play. Making these adjustments should result in better solutions and fewer cases where the optimisation fails to complete. The free-play maximisation addition requires the addition of four design variables, one for each virtual actuator. The corresponding inequality constraints limiting these variables must also be included.

Control sensitivity and resolution

An interesting application of the control allocation system that warrants further investigation is the possibility of applying the system to generating actuator sets that give the controller the option to employ actuator sets with high and low control sensitivity. This problem is briefly discussed and a possible method of implementation is suggested here.

Generating actuator sets with differing levels of control sensitivity gives the controller the ability to make higher resolution adjustments to the behaviour of the aircraft. In order to generate these sets with different levels of control sensitivity, actuators with varying levels of effectiveness are required. Consider the following example:

Given an aircraft with ailerons and flaperons, the following two sets of virtual roll actuators could be realised:

- Low resolution set - This set uses differential ailerons to realise a rolling moment. The high effectiveness of the ailerons in rolling the aircraft mean that they are capable of generating a large rolling moment. However, due to this effectiveness,

control resolution is limited due to the fact that even a small movement in the control surfaces produces a large moment.

- High resolution set - This set uses differential flaperons to realise a rolling moment. The shorter moment arm reduces the effectiveness of the flaperons in generating a rolling moment. This results in a reduced ability in terms of maximum achievable moment. However, the reduced effectiveness means that for the same actuator deflection as the low resolution set, a finer moment adjustment is possible.

The trade-off between the two sets is then as follows: A low resolution set provides a large moment with low control resolution, while a high resolution set provides smaller moments with higher control resolution.

The problem now is how to determine what actuators are suitable for high and low resolution sets. Clearly the actuators that are most effective are suitable for the low resolution set but the possibility of generating a useful high resolution set will not always be available. In some cases there may not be alternative actuators capable of creating a useful moment for the high resolution set. A high resolution virtual actuator will require larger gains in order to realise the desired forces or moments since the actuators it uses will be less effective. Penalising small mixing gains for the actuators should then result in virtual actuators that use less effective actuators and larger mixing gains, resulting in a higher-resolution virtual actuator.

8.2.2 Further recommendations

Some further recommendations for possible future work on the subject of control allocation are listed below:

- The control allocation problem can be extended to take into account non-linearities present in actuator effectiveness. This should improve the accuracy of the predicted performance of the virtual actuators.
- Actuator dynamics can be considered in more detail. A time-based formulation can be considered so that actuator dynamics can be incorporated directly in the control allocation system instead of through complimentary filters as is done in this thesis. Both [30] and [31] propose strategies using the time-based control allocation strategy that make it possible to take into account the frequency characteristics of the actuators without defining filters.

- Unconventional virtual actuators, such as a virtual lift actuator making use of flaps, could be added to the control allocation formulation. Additionally, virtual actuators could be split to be direction-dependant. An example of where this may be useful is the virtual pitch actuator for the Modular UAV. Rudders can be used for the virtual pitch-up actuator while the virtual pitch-down actuator will not contain the rudders.
- The results generated by the control allocation system should be tested on an actual aircraft to determine whether or not the aircraft model is sufficiently accurate to provide real-world performance that is comparable to the predicted performance. Additional tuning of the weights for the goals and sub-goals of the control allocation system should be investigated to optimise the results of the control allocation system.
- Further work can be done to develop the control allocation tool to be used as a design tool for designing aircraft with better actuator redundancy. The possibility of using the control allocation system as part of a multi-objective design optimisation framework can be investigated.
- Further constraining the problem to eliminate undesirable choices for virtual actuator sets, such as using rudders to provide pitch moment, should result in more intuitive solutions to the control allocation problem. Additional constraints to avoid potential problems such as tip-stall, for example, due to using the ailerons to produce lift, should be implemented. Altering actuator position constraints to take into account that different virtual actuators may be using large deflections of the same actuator can be also be considered. Actuator-induced drag could also be taken into account.
- Another interesting application for the control allocation system is to use it to update the virtual actuator mixing as the aircraft moves through different operating regions, such as different altitudes or airspeeds. The control allocation system could be run continuously in the background, receiving updated control coefficients from the system I.D. sub-system and altering the control allocations as necessary.

Appendix A

Aircraft configurations and parameters

The aircraft configurations and parameters for the three test aircraft used in this thesis are given here.

A.1 Modular UAV

The Modular UAV is a UAV developed by the CSIR as a research platform to test FTC methods. The UAV has been designed with redundancy in mind, with the following actuators available:

The eight independent aerodynamic actuators of the aircraft are defined below and the maximum and minimum deflections for all of the actuators are assumed to be $\pm 15^\circ$:

- There is one aileron on each wing. Right and left aileron deflections are defined such that a positive deflection produces a negative rolling moment.
- There is one flap on each wing. Right and left flap deflections are defined such that a positive deflection produces increased lift.
- There is a split elevator. Right and left elevator deflections are defined such that a positive deflection produces a negative pitching moment.
- There is one rudder on each of the two tail-fins. Right and left rudder deflections are defined such that a positive deflection produces a negative yawing moment.

In addition to the aerodynamic actuators, there are two engines on the aircraft, one on each wing. The engines are offset by $\pm 0.5m$ along the y-axis. It is assumed at this stage that the engines are not offset along the z-axis and that their thrust vectors act along the x-axis. The engines therefore only effect the yawing moment and the force along the x-axis. Figure A.1 shows the aircraft.

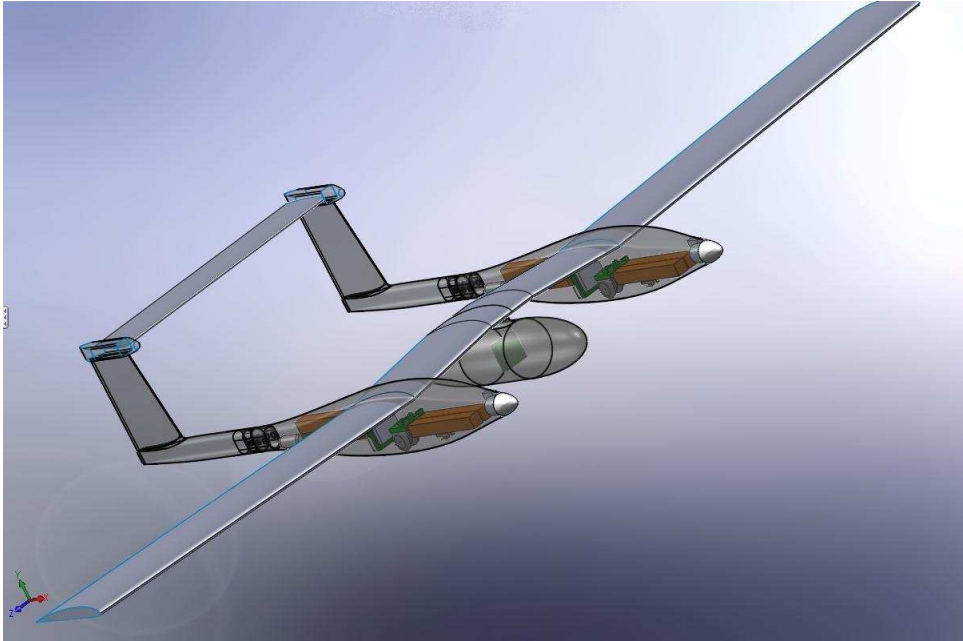


Figure A.1: The Meraka Modular UAV

A.1.1 Engine Specifications

The engine specifications are given in the table A.1.

Parameter	Value
Maximum Thrust	150 (N)
Minimum Thrust	0 (N)

Table A.1: Modular UAV Engine Parameters

A.1.2 Physical Specifications

The physical specifications of the aircraft are given in table A.2.

A.1.3 Aerodynamic Specifications

The aerodynamic specifications of the aircraft are given in tables A.3 and A.4. These parameters were generated using Athena Vortex Lattice (AVL) software.

Parameter	Value
Mass	26.0 (<i>kg</i>)
Wing Span	4.0 (<i>m</i>)
Mean Aerodynamic Chord	0.36 (<i>m</i>)
Wing Reference Area	1.44 (<i>m</i> ²)
Wing Aspect Ratio	11.11
Wing Efficiency Factor	0.85

Table A.2: Modular UAV Physical Parameters

Parameter	Value
C_{D_0}	0.06
C_{L_0}	0.5
C_{L_α}	5.557928
C_{Y_β}	-0.389444
C_{M_0}	-0.05
C_{M_α}	-1.069455
C_{l_β}	-0.071508
C_{n_β}	0.102214

Table A.3: Modular UAV Stability Derivatives

Actuator	C_{L_δ}	C_{y_δ}	C_{l_δ}	C_{m_δ}	C_{n_δ}
Left Aileron	-0.47515	-0.009786	-0.16364	0.062452	0.0057296
Right Aileron	0.47515	-0.009786	-0.16364	-0.062452	0.0057296
Left Flap	0.59232	-0.010199	0.11539	-0.065031	0.003495
Right Flap	0.59232	0.010199	-0.11539	-0.065031	-0.003495
Left Elevator	0.17624	-0.028361	0.0072193	-0.6157	0.0092819
Right Elevator	0.17624	0.028361	-0.0072193	-0.6157	-0.0092819
Left Rudder	-0.03856	0.10766	0.0029221	0.13189	-0.035695
Right Rudder	0.03856	0.10766	0.0029221	-0.13189	-0.035695

Table A.4: Modular UAV Control Derivatives

A.2 Variable stability UAV

The VSA UAV is a UAV based on the Sekwa variable stability UAV designed by the CSIR. The aircraft is a blended-wing body aircraft with no tail and it has the following actuators:

The 6 independent aerodynamic actuators of the aircraft are defined below and the maximum and minimum deflections for all of the actuators are assumed to be $\pm 15^\circ$:

- There is one aileron on each wing. Right and left aileron deflections are defined such that a positive deflection produces a negative rolling moment.
- There is one elevon on each wing. Right and left elevon deflections are defined such that a positive deflection produces increased lift.
- There is one rudder on each of the two wingtip-fins. Right and left rudder deflections are defined such that a positive deflection produces a negative yawing moment.

In addition to the aerodynamic actuators, there is one engine on the aircraft. The engine is only offset along the x-axis and it is assumed that the thrust vector acts along the x-axis.

A.2.1 Engine Specifications

The engine specifications are given in the table A.9.

Parameter	Value
Maximum Thrust	20 (<i>N</i>)
Minimum Thrust	0 (<i>N</i>)

Table A.5: VSA UAV Engine Parameters

A.2.2 Physical Specifications

The physical specifications of the aircraft are given in table A.6.

A.2.3 Aerodynamic Specifications

The aerodynamic specifications of the aircraft are given in tables A.7 and A.8. These parameters were generated using Athena Vortex Lattice (AVL) software.

Parameter	Value
Mass	3.6 (<i>kg</i>)
Wing Span	2.5 (<i>m</i>)
Mean Aerodynamic Chord	0.33 (<i>m</i>)
Wing Reference Area	0.87 (<i>m</i> ²)
Wing Aspect Ratio	7.18
Wing Efficiency Factor	0.75

Table A.6: VSA UAV Physical Parameters

Parameter	Value
C_{D_0}	0.0193
C_{L_0}	0
C_{L_α}	4.6149
C_{Y_β}	-0.14405
C_{M_0}	0
C_{M_α}	-0.64237
C_{l_β}	-0.058161
C_{n_β}	0.017169

Table A.7: VSA UAV Stability Derivatives

Actuator	C_{L_δ}	C_{y_δ}	C_{l_δ}	C_{m_δ}	C_{n_δ}
Left Aileron	-0.21	-0.031	-0.071	0.148	-0.001375
Right Aileron	0.21	-0.031	-0.071	-0.148	0.001375
Left Elevon	0.2572	0	0.03226	-0.097	-0.0007448
Right Elevon	0.2572	0	-0.03226	-0.097	0.0007448
Left Rudder	0.0073	0.04	0.0095	0.004698	-0.00722
Right Rudder	-0.0073	0.04	0.0095	0.004698	-0.00722

Table A.8: VSA UAV Control Derivatives

The data for the VSA UAV used as a test aircraft in this thesis was obtained from the thesis by [32].

A.3 SLADe Ducted fan UAV

The SLADe ducted-fan UAV was designed by the University of Stellenbosch.

The 8 independent aerodynamic actuators of the aircraft are shown in figure A.2 and deflections are defined such that a positive deflection results in an increased blockage of the airflow in the shaded regions. The maximum and minimum deflections of all of the actuators are assumed to be $\pm 15^\circ$.

In addition to the aerodynamic actuators, there are two counter-rotating engines. The thrust vectors of the engines act along the negative z-axis. Figure A.2 shows the aircraft.

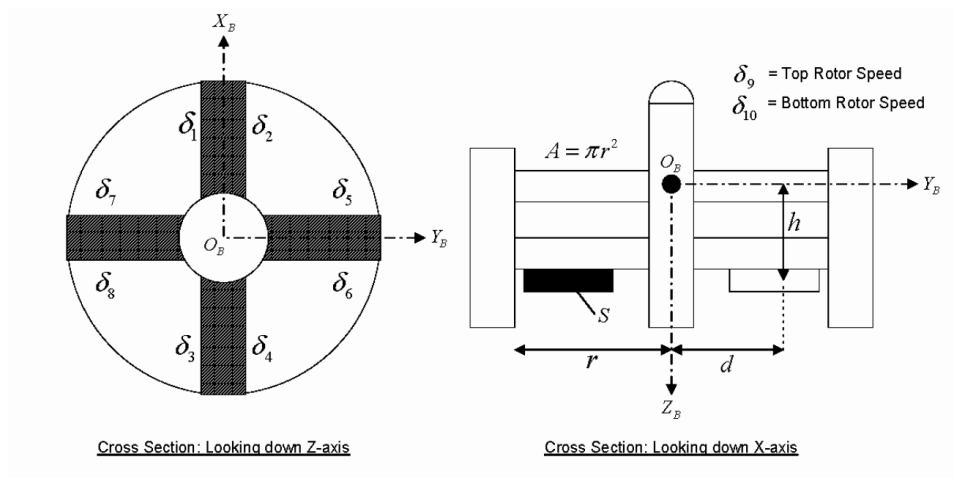


Figure A.2: The SLADe ducted-fan UAV technical diagram

A.3.1 Engine Specifications

The engine specifications are given in the table A.9.

Parameter	Value
Maximum Thrust	20 (N)
Minimum Thrust	0 (N)

Table A.9: VSA UAV Engine Parameters

Symbol	Value	Unit	Symbol	Value	Unit
d	0.2	m	I_x	1.0	kg m ²
h	0.4	m	I_y	1.0	kg m ²
r	0.375	m	I_z	1.0	kg m ²
S	0.035	m ²	a_L	2.4	ND
A	0.44	m ²	k_T	80	kgm/s ²
m	12.0	kg	k_M	2	kgm ² /s ²
A_D	0.5	m ²	C_D	0.5	ND

Appendix B

Aircraft test cases

The failure cases are broken up into four categories. The failures that were used to test the control re-allocation system are listed in each of the categories below for each of the two main aircraft.

B.1 Modular UAV

B.1.1 Failure category 1: Single-actuator failures

Single actuator failures were defined for each individual actuator and for this category those failures were at the following deflection angles: 0° , $\pm 2.5^\circ$, $\pm 5^\circ$ and $\pm 7^\circ$. The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-4	Failed Left aileron from 0° to 7.5°
5-7	Failed Left aileron from -2.5° to -7°
8-11	Failed Right aileron from 0° to 7.5°
12-14	Failed Right aileron at from -2.5° to -7°
15-18	Failed Left flap from 0° to 7.5°
19-21	Failed Left flap from -2.5° to -7°
22-25	Failed Right flap from 0° to 7.5°
26-28	Failed Right flap from -2.5° to -7°
29-32	Failed Left elevator from 0° to 7.5°
33-35	Failed Left elevator from -2.5° to -7°
36-39	Failed Right elevator from 0° to 7.5°
40-42	Failed Right elevator from -2.5° to -7°

Case #	Description
43-46	Failed Left rudder from 0° to 7.5°
47-49	Failed Left rudder from -2.5° to -7°
50-53	Failed Right rudder from 0° to 7.5°
54-56	Failed Right rudder from -2.5° to -7°

B.1.2 Failure category 2: Single-actuator failures at extreme positions

Single actuator extreme failures were defined for each individual actuator and for this category those failures were at the following deflection angles: $\pm 8^\circ$, $\pm 10^\circ$, $\pm 12.5^\circ$ and $\pm 15^\circ$. The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-4	Failed Left aileron from 8° to 15°
5-7	Failed Left aileron from -8° to -15°
8-11	Failed Right aileron from 8° to 15°
12-14	Failed Right aileron from 8° to 15°
15-18	Failed Left flap from 8° to 15°
19-21	Failed Left flap from 8° to 15°
22-25	Failed Right flap from 8° to 15°
26-28	Failed Right flap from 8° to 15°
29-32	Failed Left elevator from 8° to 15°
33-35	Failed Left elevator from 8° to 15°
36-39	Failed Right elevator from 8° to 15°
40-42	Failed Right elevator from 8° to 15°
43-46	Failed Left rudder from 8° to 15°
47-49	Failed Left rudder from 8° to 15°
50-53	Failed Right rudder from 8° to 15°
54-56	Failed Right rudder from 8° to 15°

B.1.3 Failure category 3: Multiple-actuator failures

Multiple actuator soft failures were defined with two actuators failed at a time. The actuators were failed alternating as follows for cases 1-196: $[0^\circ \ 2.5^\circ]$, $[2.5^\circ \ 0^\circ]$, $[0^\circ \ 5^\circ]$, $[5^\circ \ 0^\circ]$, $[0^\circ \ 7^\circ]$, $[7^\circ \ 0^\circ]$ and then the same with negative angles for cases 197-364. The case numbers

and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-7	Left and Right aileron
8-14	Left aileron and left flap
15-21	Left aileron and right flap
22-28	Left aileron and left elevator
29-35	Left aileron and right elevator
36-42	Left aileron and left rudder
43-49	Left aileron and right rudder
50-56	Right aileron and left flap
57-63	Right aileron and right flap
64-70	Right aileron and left elevator
71-77	Right aileron and right elevator
78-84	Right aileron and left rudder
85-91	Right aileron and right rudder
92-98	Left flap and right flap
99-105	Left flap and left elevator
106-112	Left flap and right elevator
113-119	Left flap and left rudder
120-126	Left flap and right rudder
127-133	Right flap and left elevator
134-140	Right flap and right elevator
141-147	Right flap and left rudder
148-154	Right flap and right rudder
155-161	Left elevator and right elevator
162-168	Left elevator and left rudder
169-175	Left elevator and right rudder
176-182	Right elevator and left rudder
183-189	Right elevator and right rudder
190-196	Left rudder and right rudder
197-202	Left and Right aileron
203-208	Left aileron and left flap
209-214	Left aileron and right flap
215-220	Left aileron and left elevator
221-226	Left aileron and right elevator
227-232	Left aileron and left rudder

Case #	Description
233-238	Left aileron and right rudder
239-244	Right aileron and left flap
245-250	Right aileron and right flap
251-256	Right aileron and left elevator
257-262	Right aileron and right elevator
263-268	Right aileron and left rudder
269-274	Right aileron and right rudder
275-280	Left flap and right flap
281-286	Left flap and left elevator
287-292	Left flap and right elevator
293-298	Left flap and left rudder
299-304	Left flap and right rudder
305-310	Right flap and left elevator
311-316	Right flap and right elevator
317-322	Right flap and left rudder
323-328	Right flap and right rudder
329-334	Left elevator and right elevator
335-340	Left elevator and left rudder
341-346	Left elevator and right rudder
347-352	Right elevator and left rudder
353-358	Right elevator and right rudder
359-364	Left rudder and right rudder

B.1.4 Failure category 4: Multiple-actuator failures at extreme positions

Multiple actuator extreme failures were defined with two actuators failed at a time. The actuators were failed alternating as follows for cases 1-224: $[0^\circ \ 8^\circ]$, $[8^\circ \ 0^\circ]$, $[0^\circ \ 10^\circ]$, $[10^\circ \ 0^\circ]$, $[0^\circ \ 12.5^\circ]$, $[12.5^\circ \ 0^\circ]$, $[0^\circ \ 15^\circ]$, $[15^\circ \ 0^\circ]$ and then the same with negative angles for cases 225-448. Cases 449-452 involved failures where more than two actuators were failed 0° . The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-8	Left and Right aileron
9-15	Left aileron and left flap
17-24	Left aileron and right flap

Case #	Description
25-32	Left aileron and left elevator
33-40	Left aileron and right elevator
41-48	Left aileron and left rudder
49-56	Left aileron and right rudder
57-64	Right aileron and left flap
65-72	Right aileron and right flap
73-80	Right aileron and left elevator
81-88	Right aileron and right elevator
89-96	Right aileron and left rudder
97-104	Right aileron and right rudder
105-112	Left flap and right flap
113-120	Left flap and left elevator
121-128	Left flap and right elevator
129-136	Left flap and left rudder
137-144	Left flap and right rudder
145-152	Right flap and left elevator
153-160	Right flap and right elevator
161-168	Right flap and left rudder
169-176	Right flap and right rudder
177-184	Left elevator and right elevator
185-192	Left elevator and left rudder
193-200	Left elevator and right rudder
201-208	Right elevator and left rudder
209-216	Right elevator and right rudder
217-224	Left rudder and right rudder
225-232	Left and Right aileron
233-240	Left aileron and left flap
241-248	Left aileron and right flap
249-256	Left aileron and left elevator
257-264	Left aileron and right elevator
265-272	Left aileron and left rudder
273-280	Left aileron and right rudder
281-288	Right aileron and left flap
289-296	Right aileron and right flap
297-304	Right aileron and left elevator
305-312	Right aileron and right elevator

Case #	Description
313-320	Right aileron and left rudder
321-327	Right aileron and right rudder
328-336	Left flap and right flap
337-344	Left flap and left elevator
345-352	Left flap and right elevator
353-360	Left flap and left rudder
361-368	Left flap and right rudder
369-376	Right flap and left elevator
377-384	Right flap and right elevator
385-392	Right flap and left rudder
393-400	Right flap and right rudder
401-408	Left elevator and right elevator
409-416	Left elevator and left rudder
417-424	Left elevator and right rudder
425-432	Right elevator and left rudder
433-440	Right elevator and right rudder
441-448	Left rudder and right rudder
449	Failed Right flap, Left elevator and Right rudder at 0°
450	Failed Left flap, Right elevator and Left rudder at 0°
451	Failed Left aileron, Left flap, Left elevator and Left rudder at 0°
452	Failed Left aileron, Right rudder, Left elevator and Right flap at 0°

B.2 VSA UAV

B.2.1 Failure category 1: Single-actuator failures

Single actuator failures were defined for each individual actuator and for this category those failures were at the following deflection angles: 0° , $\pm 2.5^\circ$, $\pm 5^\circ$ and $\pm 7^\circ$. The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-4	Failed Left aileron from 0° to 7°
5-7	Failed Left aileron from -2.5° to -7°
8-11	Failed Right aileron from 0° to 7°
12-14	Failed Right aileron from -2.5° to -7°

Case #	Description
15-18	Failed Left elevon from 0° to 7°
19-21	Failed Left elevon from -2.5° to -7°
22-25	Failed Right elevon from 0° to 7°
26-28	Failed Right elevon from -2.5° to -7°
29-32	Failed Left rudder from 0° to 7°
33-35	Failed Left rudder from -2.5° to -7°
36-39	Failed Right rudder from 0° to 7°
40-42	Failed Right rudder from -2.5° to -7°

B.2.2 Failure category 2: Single-actuator failures at extreme positions

Single actuator failures were defined for each individual actuator and for this category those failures were at the following deflection angles: 8° , $\pm 10^\circ$, $\pm 12.5^\circ$ and $\pm 15^\circ$. The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-4	Failed Left aileron from 8° to 15°
5-7	Failed Left aileron from -8° to -12.5°
8-11	Failed Right aileron from 8° to 15°
12-14	Failed Right aileron from -8° to -12.5°
15-18	Failed Left elevon from 8° to 15°
19-21	Failed Left elevon from -8° to -12.5°
22-25	Failed Right elevon from 8° to 15°
26-28	Failed Right elevon from -8° to -12.5°
29-32	Failed Left rudder from 8° to 15°
33-35	Failed Left rudder from -8° to -12.5°
36-39	Failed Right rudder from 8° to 15°
40-42	Failed Right rudder from -8° to -12.5°

B.2.3 Failure category 3: Multiple-actuator failures

Multiple actuator failures were defined with two actuators failed at a time. The actuators were failed alternating as follows for cases 1-66: $[0^\circ 2.5^\circ]$, $[2.5^\circ 0^\circ]$, $[0^\circ 5^\circ]$, $[5^\circ 0^\circ]$, $[0^\circ 7^\circ]$, $[7^\circ 0^\circ]$ and then the same with negative angles for cases 67-132. Cases 133-143

are combinations of aileron, elevon and rudder failures at 0° . The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-6	Left and Right aileron
7-12	Left aileron and left elevon
13-18	Left aileron and right elevon
19-24	Left aileron and left rudder
25-30	Right aileron and left elevon
31-36	Right aileron and right elevon
37-42	Right aileron and right rudder
43-48	Left elevon and left rudder
49-54	Left elevon and right rudder
55-60	Right elevon and left rudder
61-66	Right elevon and right rudder
67-72	Left and Right aileron
73-78	Left aileron and left elevon
79-84	Left aileron and right elevon
85-90	Left aileron and left rudder
91-96	Right aileron and left elevon
97-102	Right aileron and right elevon
103-108	Right aileron and right rudder
109-114	Left elevon and left rudder
115-120	Left elevon and right rudder
121-126	Right elevon and left rudder
127-132	Right elevon and right rudder
133-143	Combinations of aileron, elevon and rudder failures at 0°

B.2.4 Failure category 4: Multiple-actuator failures at extreme positions

Multiple actuator extreme failures were defined with two actuators failed at a time. The actuators were failed alternating as follows for cases 1-88: $[0^\circ 8^\circ]$, $[8^\circ 0^\circ]$, $[0^\circ 10^\circ]$, $[10^\circ 0^\circ]$, $[0^\circ 12.5^\circ]$, $[12.5^\circ 0^\circ]$, $[0^\circ 15^\circ]$, $[15^\circ 0^\circ]$ and then the same with negative angles for cases 89-176. The case numbers and the corresponding failure case are given in table below:

Case #	Description
0	Nominal mode
1-8	Left and Right aileron
9-16	Left aileron and left elevon
17-24	Left aileron and right elevon
25-32	Left aileron and left rudder
33-40	Right aileron and left elevon
41-48	Right aileron and right elevon
49-56	Right aileron and right rudder
57-64	Left elevon and left rudder
65-72	Left elevon and right rudder
73-80	Right elevon and left rudder
81-88	Right elevon and right rudder
89-96	Left and Right aileron
97-104	Left aileron and left elevon
105-112	Left aileron and right elevon
113-120	Left aileron and left rudder
121-128	Right aileron and left elevon
129-136	Right aileron and right elevon
137-144	Right aileron and right rudder
145-152	Left elevon and left rudder
153-160	Left elevon and right rudder
161-168	Right elevon and left rudder
169-176	Right elevon and right rudder

Appendix C

Adverse force minimisation graphs

The graphs that illustrate the adverse force minimisation of control allocation system configuration 2 compared to configuration 1 are shown here. The plots for failure categories 2, 3 and 4 are shown for both the Modular UAV as well as the VSA UAV.

C.1 Modular UAV

C.1.1 Failure category 2

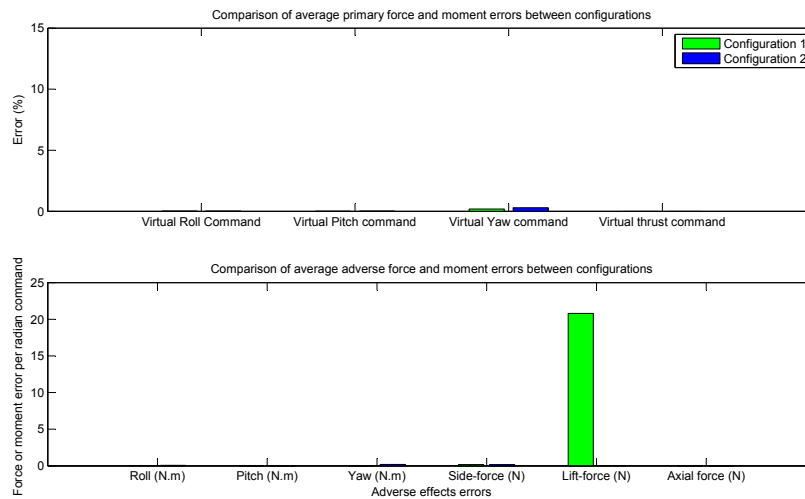


Figure C.1: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 2 showing the benefit of taking into account adverse forces for the Modular UAV

C.1.2 Failure category 3

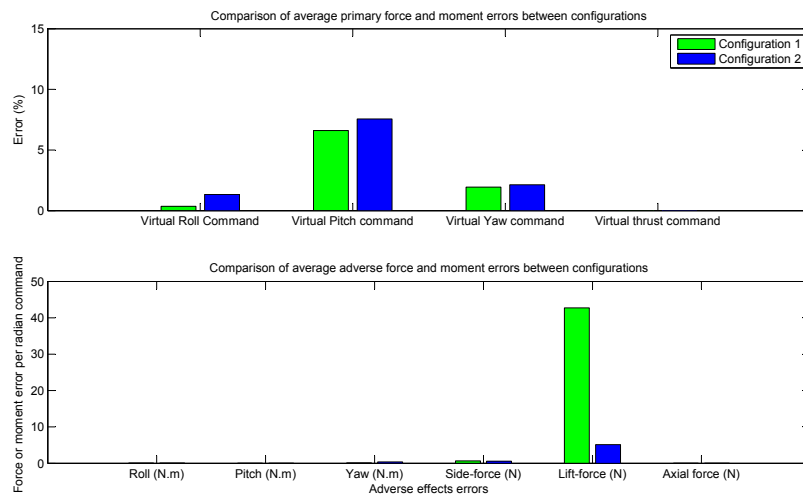


Figure C.2: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 3 showing the benefit of taking into account adverse forces for the Modular UAV

C.1.3 Failure category 4

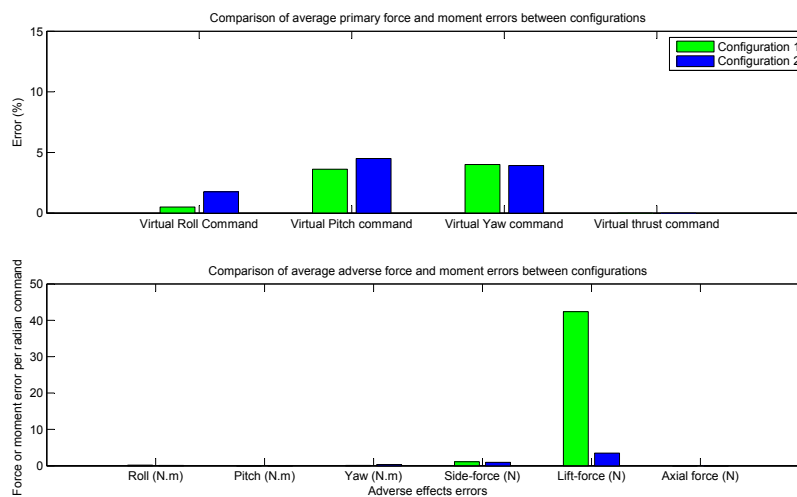


Figure C.3: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 4 showing the benefit of taking into account adverse forces for the Modular UAV

C.2 VSA UAV

C.2.1 Failure category 2

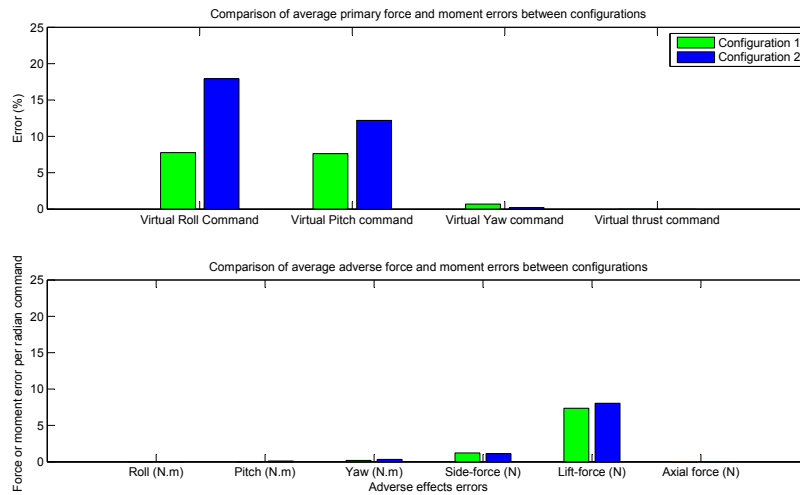


Figure C.4: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 2 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV

C.2.2 Failure category 3

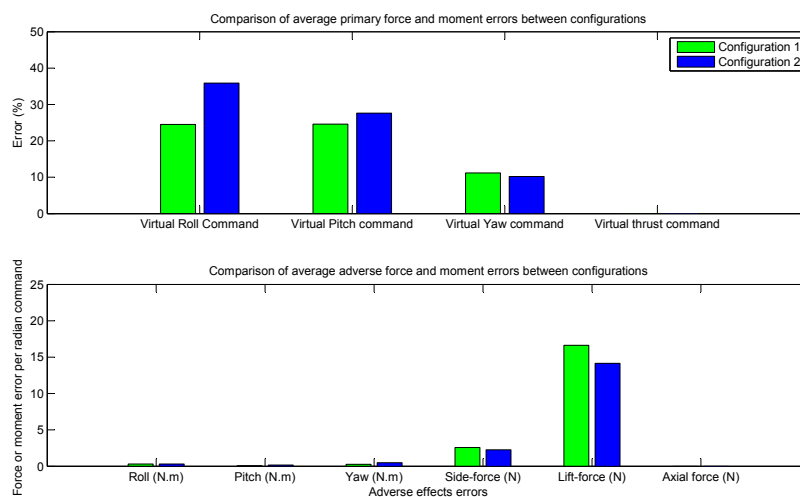


Figure C.5: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 3 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV

C.2.3 Failure category 4

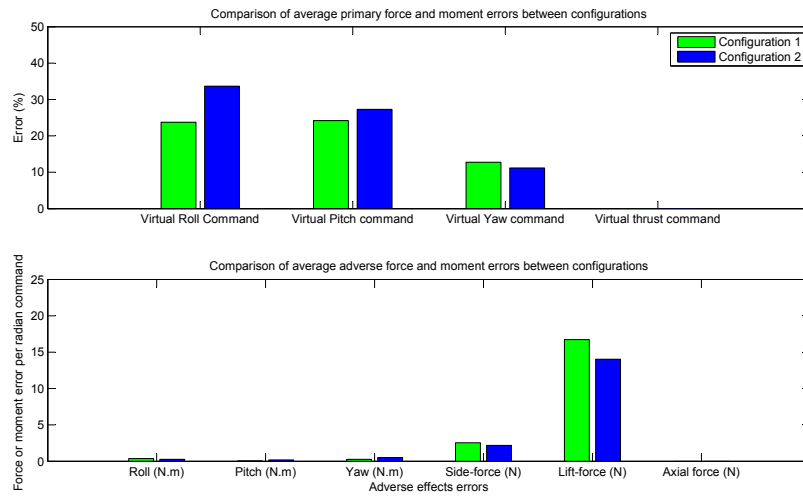


Figure C.6: Comparison between the average primary moment and adverse effects errors for the two control allocation system configurations for failure category 4 showing that there is minimal benefit of taking into account adverse forces for the VSA UAV

Appendix D

Individual case error plots

The individual case error plots for failure categories 3 and 4. Figures D.1 and D.2 give the plots for failure categories 3 and 4 for the Modular UAV while figures D.3 and D.4 give the plots for failure categories 3 and 4 for the VSA UAV. Although the resolution is not high enough for the reader to accurately identify individual failure cases, the plots provide an overview of the groups of failures that performed well or poorly. The higher resolution digital copies of these plots were used to identify the individual cases described in chapter 7.

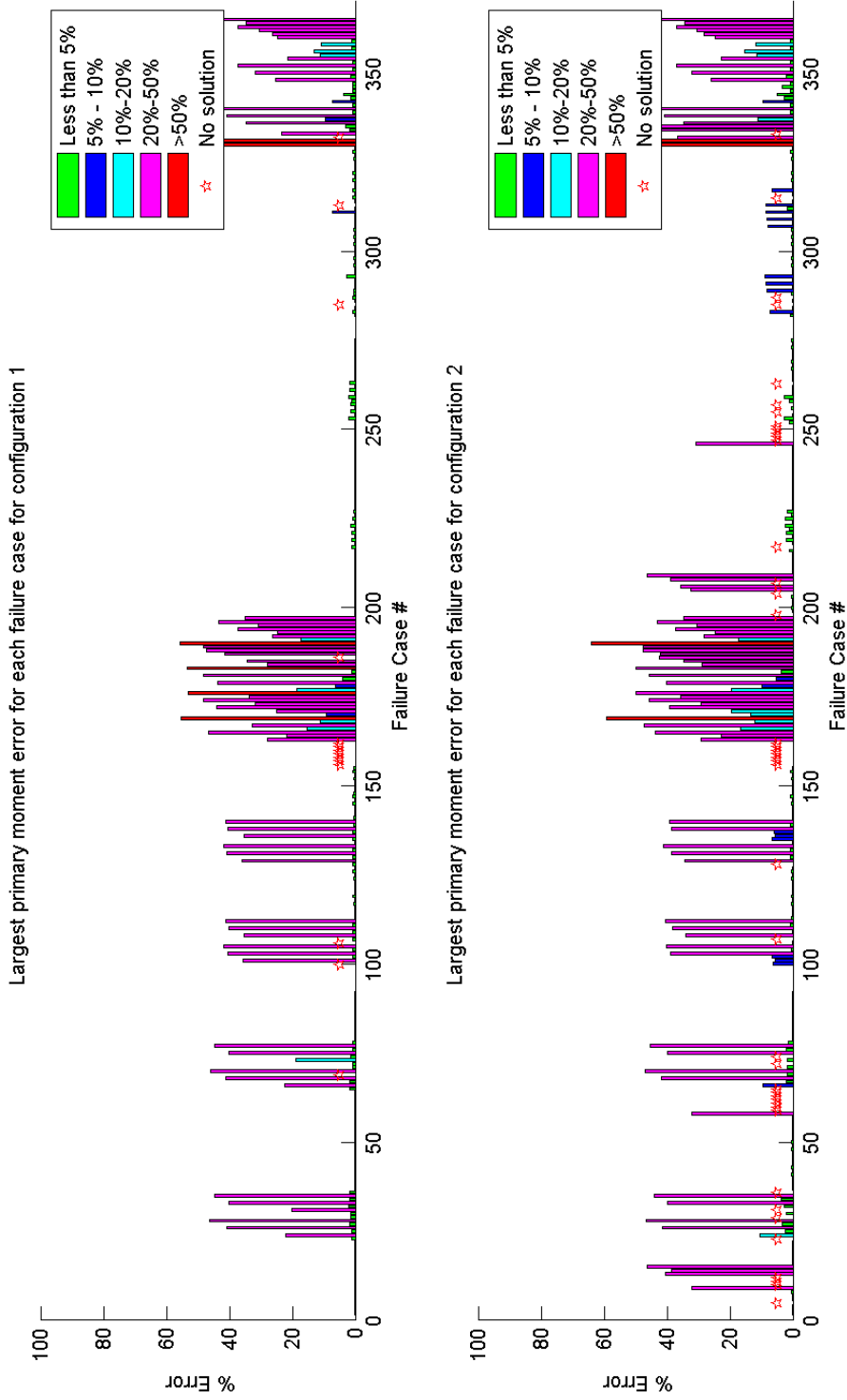


Figure D.1: A plot of the largest primary moment errors for each failure case for failure category 3 (Modular UAV)

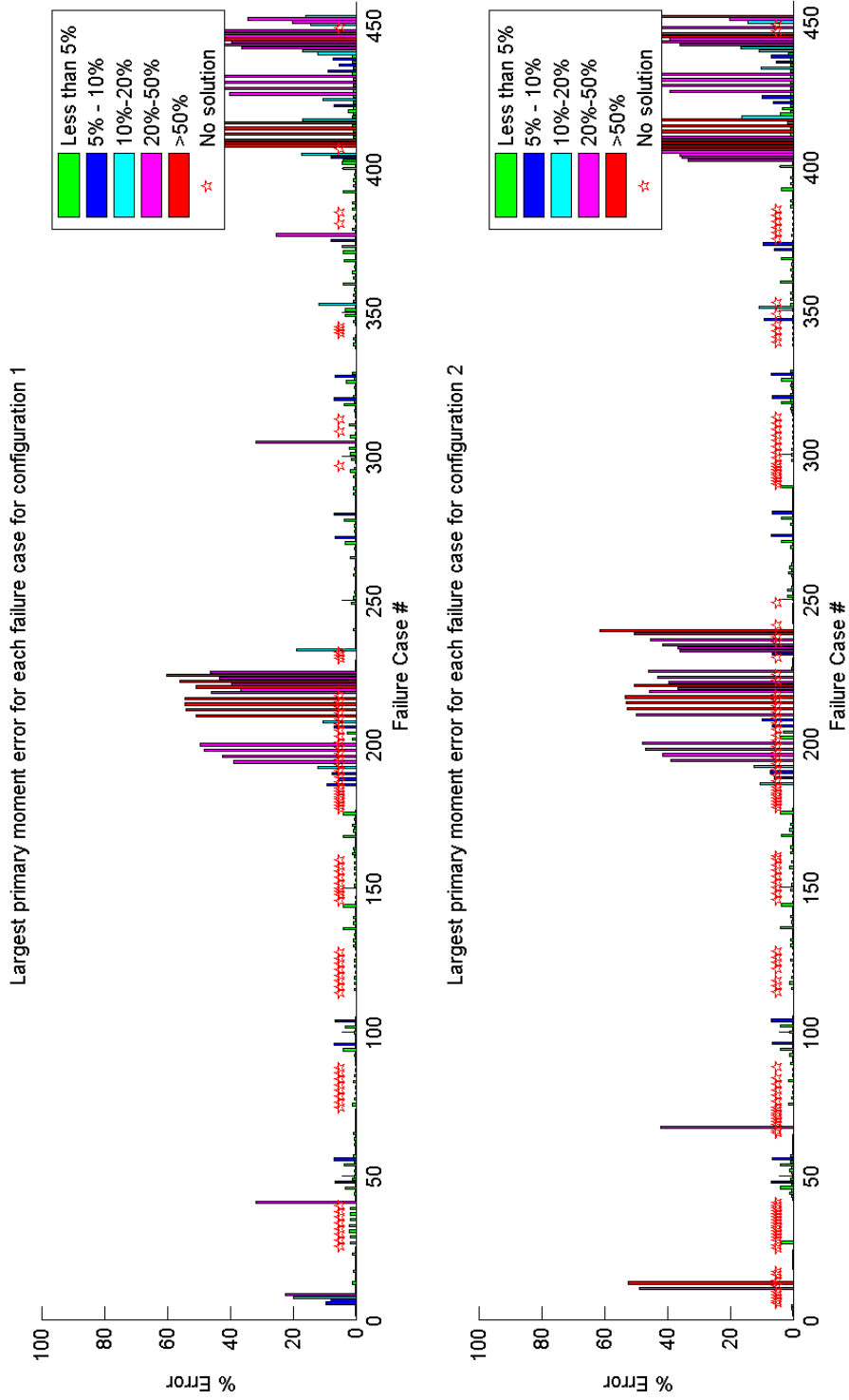


Figure D.2: A plot of the largest primary moment errors for each failure case for failure category 4 (Modular UAV)

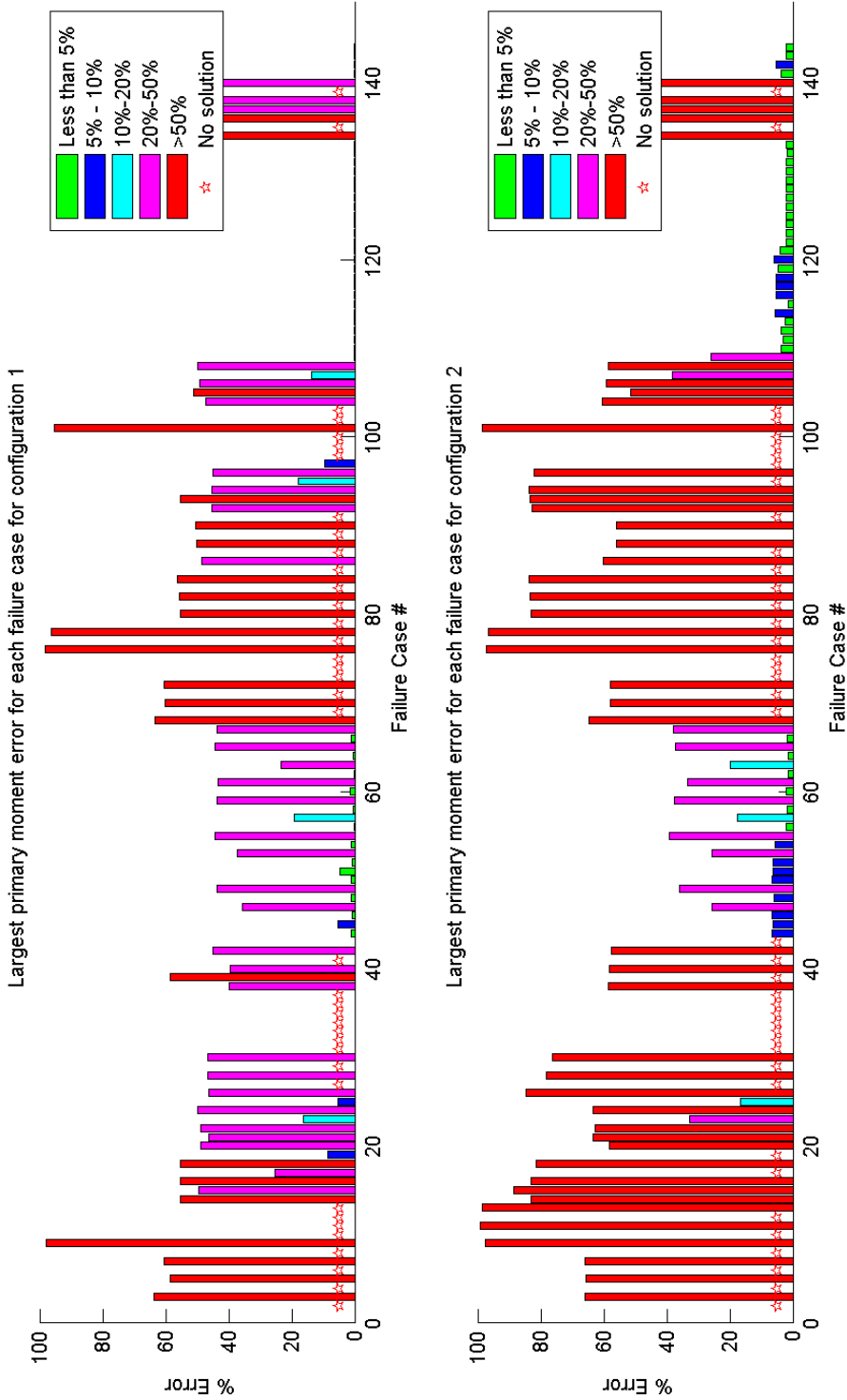


Figure D.3: A plot of the largest primary moment errors for each failure case for failure category 3 (VSA UAV)

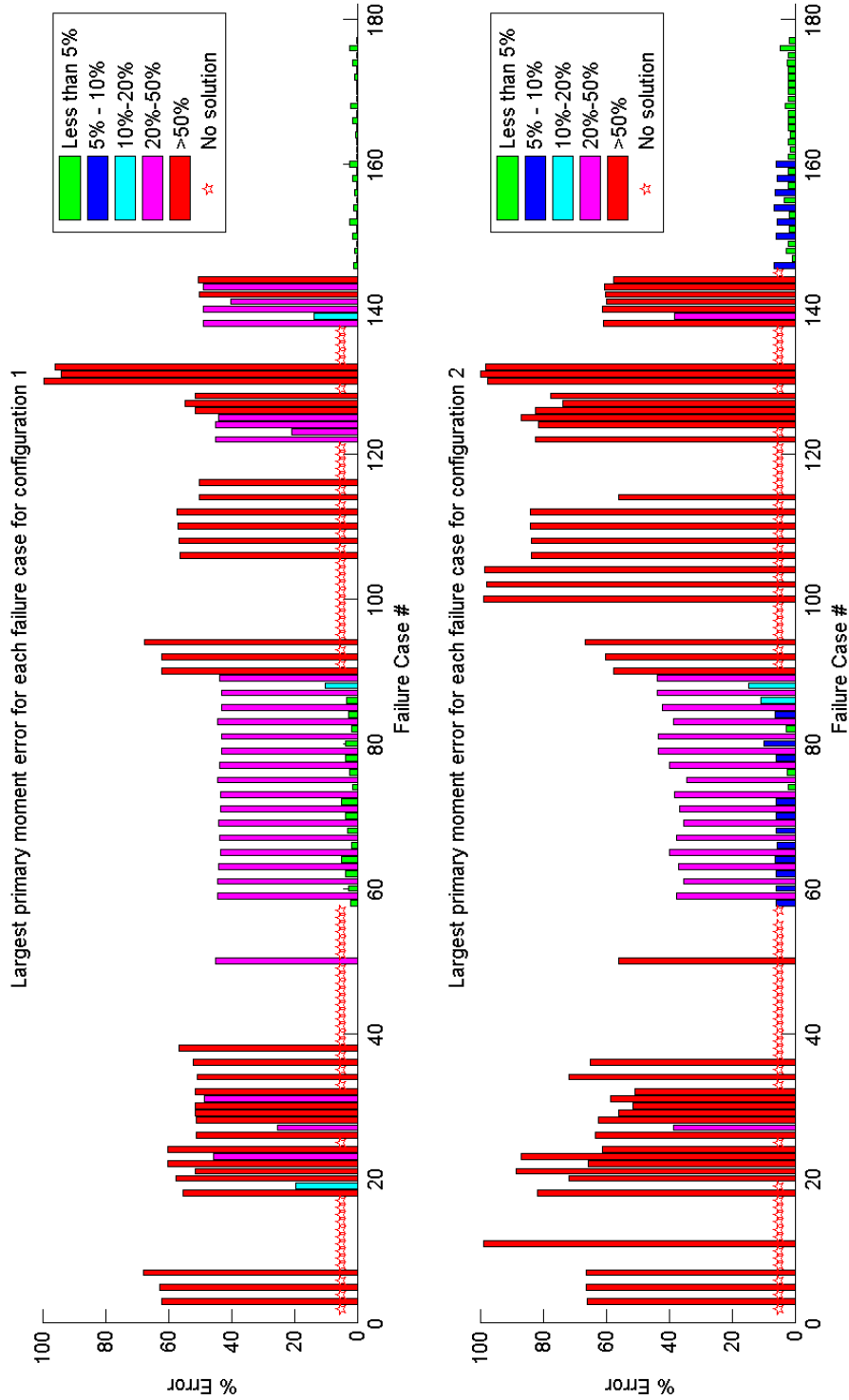


Figure D.4: A plot of the largest primary moment errors for each failure case for failure category 4 (VSA UAV)

List of References

- [1] Peddle, I.: Fault tolerant control: Overview and research at Stellenbosch University. November 2009.
- [2] Oppenheimer, W., Bowman, D. and Bolender, M.: Control allocation for over-actuated systems. *14th Mediterranean Conference on Control and Automation*, pp. 1–6, 2006.
- [3] Houw, W.: *Autonomous Aerobatic Flight of a Fixed-Wing Unmanned Aerial Vehicle*. Master's thesis, Stellenbosch University, 2007.
- [4] Gaum, D.: *Aggressive Flight Control Techniques for a Fixed-Wing Unmanned Aerial Vehicle*. Master's thesis, Stellenbosch University, 2009.
- [5] Peddle, I.: Aircraft dynamics 813: Introductory course to aircraft dynamics.
- [6] CAST: Asymmetric flight. Online.
Available at: http://www.cast-safety.org/pdf/5_asymmetric_flight.pdf
- [7] Zhang, Y. and Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, vol. 32, pp. 229–252, 2008.
- [8] Ducard, G.: *Fault-Tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*. Springer, 2009.
- [9] Boskovic, J. and Mehra, R.: Control allocation in overactuated aircraft under position and rate limiting. In: *Proceedings of the American Control Conference*. 2002.
- [10] Beck, R.: *Application of Control All Methods to Linear Systems with Four or More Objectives*. Ph.D. thesis, Virginia Polytechnic Institute and State University, 2002.
- [11] Härkegård, O.: *Backstepping and Control Allocation with Applications to Flight Control*. Ph.D. thesis, Linköping University, 2003.
- [12] Bodson, M.: Evaluation of optimization methods for control allocation. *Journal of Guidance, Control and Dynamics*, vol. 25, No. 4, pp. 703–711, 2002.
- [13] Burken, J., Lu, P., Wu, Z. and Bahm, C.: Two reconfigurable flight-control design methods: Robust servomechanism and control allocation. *Journal of Guidance, Control and Dynamics*, vol. 24, No. 3, pp. 482–493, 2001.
- [14] Cotting, M. and Burken, J.: Reconfigurable control design for the full X-33 flight envelope. *American Institute of Aeronautics and Astronautics*, 2001.

- [15] Durham, W.: Constrained control allocation: Three-moment problem. *Journal of Guidance, Control and Dynamics*, vol. 17, No. 2, pp. 330–336, 1994.
- [16] Peterson, J. and Bodson, M.: Interior-point algorithms for control allocation. *Journal of Guidance, Control and Dynamics*, vol. 28, No. 3, pp. 471–480, 2005.
- [17] Buffington, J.M.: Modular control law design for the innovative control effectors (ice) tailless fighter aircraft configuration 101-3. Tech. Rep., Air Force Research Laboratory Air Vehicles Directorate Wright-Patterson AFB, 1999.
- [18] Peterson, J. and Bodson, M.: Constrained quadratic programming techniques for control allocation. *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2003.
- [19] Johansen, T.A., Fossen, T. and Berge, S.P.: Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming. *IEEE Transactions on Control Systems Technology*, vol. 12, 2004.
- [20] Peddle, I.: *Autonomous Flight of a Model Aircraft*. Master's thesis, Stellenbosch University, 2005.
- [21] Etkin, B. and Reid, L.: *Dynamics of Flight Stability and Control*. John Wiley & Sons, Inc, 1996.
- [22] Cook, M.: *Flight Dynamics Principles*. Elsevier Ltd, 2007.
- [23] Peddle, I.: *Acceleration Based Manoeuvre Flight Control System for Unmanned Aerial Vehicles*. Ph.D. thesis, Stellenbosch University, 2008.
- [24] Nguyen, N., Krishnakumar, K. and Kaneshige, J.: Dynamics and adaptive control for stability recovery of damaged asymmetric aircraft. *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2006.
- [25] Arora, J.: *Introduction to optimum design*. Elsevier Inc, 2004.
- [26] J.S.Arora, R.M.: Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, vol. 26, No. 6, pp. 369–395, 2004.
- [27] Rao, S.: *Engineering Optimization: Theory and Practice*. John Wiley & Sons, Inc, 2009.
- [28] Jaquet, C.: *Control Surfaces in Confined Spaces: The Optimisation of Trailing Edge Tabs to Reduce Control Surface Hinge Moments*. Master's thesis, Stellenbosch University, 2010.

- [29] Peddle, I., Jones, T. and Treurnicht, J.: Practical near hover flight control of a ducted fan. *Control Engineering Practice*, vol. 17, 2009.
- [30] Härkegård, O.: Dynamic control allocation using constrained quadratic programming. *Journal of Guidance, Control and Dynamics*, vol. 27, 2004.
- [31] Venkataraman, R., Oppenheimer, M. and Doman, D.: A new control allocation method that accounts for effector dynamics. *Proceedings of the 2004 IEEE Aerospace Conference*, vol. 4, pp. 2710–2715, 2004.
- [32] Blaauw, D.: *Flight Control System for a Variable Stability Blended-Wing-Body Unmanned Aerial Vehicle*. Master's thesis, Stellenbosch University, 2009.