



TECHNICAL UNIVERSITY OF CRETE
DEPARTMENT OF ELECTRONIC AND COMPUTER ENGINEERING

**‘Development of a Complete
Fault Detection and Identification (FDI)
And Fault Tolerant Control (FTC) Scheme
For a fixed wing UAV’**

**by
George Limnaios**

Supervisor:
Professor George Stavrakakis
Guidance Committee:
Professor Michael Zervakis
Professor Nikolaos Tsourveloudis

M.Sc. Thesis

October 2011

**Αφιερωμένο στους γονείς
και την Οικογένειά μου
για την υπομονή και την υποστήριξή τους**

Table of Contents

Chapter 1- Introduction	1
1.1 Why Fault-Tolerant Control?	1
1.2 Motivation for Fault –Tolerant Flight Control	2
1.3 Fault –Tolerant Flight Control for UAVs	8
1.4 Purpose of the thesis	11
 Chapter 2 – UAV Simulation Model	 13
2.1 Nonlinear Model of the UAV	13
2.2 Definition of the Frames	13
2.2.1 Navigation Frame	13
2.2.2 Euler Angles	13
2.2.3 Body Frame	13
2.2.4 Quaternion representation	15
2.2.5 Wind frame	16
2.3 Wind Disturbance	17
2.4 Rigid Body Equations of Motion	18
2.5 Model of the Aircraft total forces and moments	21
2.5.1 Model of the Aerodynamics forces and moments	21
2.5.2 Model of the thrust (engine) forces and moments	22
2.6 Control Vector and Actuators	23
2.7 Sensors	24
 Chapter 3 - Modeling of Faults on an Aircraft	 26
3.1 Fault-Failure	26
3.2 Faults and Failures on an Aircraft System	27
3.2.1 Sensor Faults/Failures	27
3.2.2 Actuator Faults/Failures	28
3.2.3 Structural Faults/Failures	31
3.3 Focus of the thesis	33
 Chapter 4 - Review on Fault Tolerant Control (FTC) and Fault Detection and Identification (FDI)	 34
4.1 Fault Detection and Identification	36
4.1.1 Model Based FDI	37

4.1.1.1	Parameter Estimation based methods	39
4.1.1.2	Parity Equation based Methods	40
4.1.1.3	Observer based Methods	42
4.1.1.4	Non linear Systems	47
4.1.1.5	Residual Evaluation	51
4.1.1.6	On Fault Isolation	52
4.1.1.7	On Fault Identification	54
4.2	Fault Tolerant Control (FTC)	56
4.3	Discussion on FDI-FTC method for UAVs	58

Chapter 5 - Nonlinear Fault Detection and Identification

System based on a two step method

61

5.1	Multiple Model Adaptive Estimation (MMAE)	61
5.2	Design of the Extended Kalman Filters	62
5.3	Design of the Hypothesis Conditional Probability Computation Module	65
5.4	Parameter Estimation method for control surface damage Fault Detection and Isolation	71
5.5	The need for a supervision system	84

Chapter 6 - Nonlinear Fault Tolerant Flight Control System

89

6.1	Non-linear Dynamic Inversion based controller	90
6.1.1	Derivation of a Dynamic Inversion Controller	90
6.1.2	NDI applied on a UAV	91
6.1.3	Fault Tolerance in an NDI control system	96
6.1.3.1	Damaged Control Surface Case	97
6.1.3.2	Stuck Control Surface Case	101

Chapter 7. Summary and conclusions

109

Bibliography

112

Appendix

125

List of Tables

Table 2.1 Definition of Inertia Coefficients	20
Table 4.1 Comparison of different FDI methods and their properties	55
Table 4.2. Comparison of FTC techniques and their properties	60

List of Figures

Figure 1.1 Fault-tolerant schemes for electronic hardware	4
Figure 1.2 Flight Control System Architecture of the Eurofighter Typhoon	5
Figure 1.3 Accidents statistics	6
Figure 1.4 DHL A300B4 emergency landing after being hit by a missile in Baghdad, 2003	7
Figure 1.5 Kalila Air emergency landing after losing one engine, 2004	8
Figure 1.6 Failure sources for UAVs: (a) US military based on 194000 flight hours and (b) IAI military based on 100000 flight hours.	9
Figure 1.7 Mishap Rate Comparisons	10
Figure 1.8 Aerobatic UAV developed by ETH Zurich	12
Figure 2.1 Euler angles	14
Figure 2.2 Euler angles and frame transformations	14
Figure 2.3 3D representations of the Euler angles	14
Figure 2.4 Angle of attack and sideslip angle definition	16
Figure 2.5 Modeling atmospheric turbulence as filtered white noise	17
Figure 2.6 Orientation of the linear and angular velocity components, external forces and moments, angle of attack and sideslip angle in relation to the body-fixed reference frame of the aircraft	19
Figure 3.1 Classification of faults according to location	26
Figure 3.2 Types of Sensors faults/failures	27
Figure 3.3 Types of Actuator faults/failures	29
Figure 3.4 After an Actuator fault the system may become unstable if no reconfiguration takes place	31
Figure 4.1 Classification of FTC	34
Figure 4.2 A general structure of Active Fault Tolerant Controller (AFTC)	35

Figure 4.3 Areas of fault tolerant control research	35
Figure 4.4 Classification of FDI methods	36
Figure 4.5 Schematic description of model based fault diagnosis scheme	38
Figure 4.6 Schematic description of the parameter identification scheme	40
Figure 4.7 Principle of Observer-based residual generation	42
Figure 4.8 Fault estimation based FDI	46
Figure 4.9 The H_{∞} -based fault estimation problem	46
Figure 4.10 Two schemes of structured residual sets for the isolation of three faults:	
(a) Dedicated scheme and	
(b) Generalized residual scheme	53
Figure 4.11 Classification of Fault-Tolerant Flight Control Methods with recent examples	57
Figure 5.1 MMAE with Extended Kalman Filtering. Each Filter monitors its assigned actuator	62
Figure 5.2. Aileron 1 fault detection	67
Figure 5.3. Aileron 1 faulty deflection estimation (Stuck failure at 0.02)	68
Figure 5.4. Aileron 1 floating failure detection	69
Figure 5.5. Aileron 1 faulty deflection estimation (Floating failure)	69
Figure 5.6. Pitch rate estimation compared with measured and true values	73
Figure 5.7. Roll rate estimation compared with measured and true values	73
Figure 5.8. Yaw rate estimation compared with measured and true values	74
Figure 5.9. Average Residuals computed during an elevator structural failure at t=35s (3500 time sample)	77
Figure 5.10. Failure Probabilities computed by the MMAE filter during the elevator structural failure at t=35s (3500 time sample)	78

Figure 5.11. Average Residuals computed during no failure conditions with significant excitation of control surfaces	78
Figure 5.12 a. Estimation of elevator effectiveness coefficient	79
Figure 5.12 b. Estimation of pitch rate coefficient	79
Figure 5.12 c. Estimation of angle of attack coefficient	80
Figure 5.12 d. Estimation of bias coefficient	80
Figure 5.13 Estimation of elevator effectiveness for rolling moment (C_{lde}) coefficient in the case of a left (red) and right (blue) elevator failures	81
Figure 5.14 Fuzzification of input variable C_{mde}	82
Figure 5.15 Fuzzification of input variable C_{lde}	83
Figure 5.16 Fuzzification of output variable FaultEstimation	83
Figure 5.17. Average Residuals computed during a stuck elevator at last position failure conditions	84
Figure 5.18. MMAE Filter probabilities computed during a stuck elevator at last position failure conditions	85
Figure 5.19. MMAE Filter left elevator deflection estimation against real deflection, computed during a stuck elevator at last position failure conditions	86
Figure 5.20. Fault detection logic implemented in the supervision module	87
Figure 5.21. Overview of the proposed Fault Detection and Identification (FDI) module	88
Figure 6.1. Non-linear Dynamic Inversion general scheme	91
Figure 6.2. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure	97
Figure 6.3. Estimated versus true C_{ma} parameter (closed loop identification)	98
Figure 6.4. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure	98
Figure 6.5. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure with elevator doublet applied for identification purposes after the failure	99

Figure 6.6. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure	100
Figure 6.7. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure	100
Figure 6.8. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure	101
Figure 6.9. Fault probabilities calculated from the MMAE algorithm of the FDI subsystem (closed loop). The detection and isolation is achieved in 1 second	102
Figure 6.10. Estimated versus real right elevator deflection. The estimated values are valid only after the failure injection	103
Figure 6.11. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 0.25 deg from trim	103
Figure 6.12. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 0.25 deg from trim. The failure is injected during excitation of the system	104
Figure 6.13. Fault probabilities calculated from the MMAE algorithm of the FDI subsystem (closed loop). The detection and isolation is achieved in 0.6 second	105
Figure 6.14. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system	106
Figure 6.15. Adaptive NDI based fault tolerant controller commanded control surface deflections	106

Figure 6.16. INDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system	107
Figure 6.17. INDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system. The estimated fault deflection accuracy was chosen 50% away from true value	108
Figure 7.1 Design cycle for reliability improvement of UAVs	111

Περίληψη

Στην παρούσα εργασία παρουσιάζεται η σχεδίαση ενός ολοκληρωμένου συστήματος αυτόματης αναγνώρισης βλαβών (Fault Detection and Identification, “FDI”) και ελέγχου παρουσία βλαβών (Fault Tolerant Control, “FTC”), σε ένα μη επανδρωμένο αεροσκάφος σταθερής πτέρυγας. Τα μη επανδρωμένα αεροσκάφη έχουν προκαλέσει το ενδιαφέρον των ερευνητών καθώς η αρχική επιτυχής εφαρμογή τους για στρατιωτικούς κυρίως σκοπούς έχει δείξει τις τεράστιες δυνατότητες της αξιοποίησής τους για εμπορικούς σκοπούς (όπως τηλεπικοινωνιακές εφαρμογές, ερευνητικές αποστολές, περιβαλλοντική επιτήρηση, ασφάλεια συνόρων, έγκαιρη προειδοποίηση και συντονισμός κατάσβεσης πυρκαγιών, μεταφορά εμπορευμάτων κ.α). Απαραίτητη προϋπόθεση για την εμπορική χρήση των αεροσκαφών αυτών ωστόσο είναι η ασφαλής και χωρίς εμπόδια ενσωμάτωσή τους στον διεθνή και εθνικό αέριο χώρο, κάτι που δεν είναι άμεσα εφικτό τόσο εξαιτίας της έλλειψης διαδικασιών πιστοποίησης (Airworthiness standards) όσο και κανονισμών χρήσης. Η συνύπαρξη των μη επανδρωμένων αεροσκαφών με τα ευρέως χρησιμοποιούμενα επανδρωμένα αεροσκάφη και η χρήση τους πάνω από πυκνοκατοικημένες περιοχές προϋποθέτει την επίτευξη εκ μέρους τους σημαντικών επιδόσεων ασφάλειας συγκρίσιμων με αυτά των επανδρωμένων αεροσκαφών. Οι στόχοι αυτοί ωστόσο δεν είναι εύκολο να επιτευχθούν με την υπάρχουσα τεχνολογία που χρησιμοποιεί η αεροδιαστημική τεχνολογία ευρέως όπως την εναλλαγξιμότητα υλικού (Hardware redundancy), την χρήση δηλαδή πολλαπλών συστημάτων με το ίδιο αντικείμενο ως εφεδρικά. Ο κύριος λόγος είναι ότι τα μη επανδρωμένα αεροσκάφη έχουν περισσότερους περιορισμούς κόστους και ωφέλιμου φορτίου. Είναι αυτοί οι περιορισμοί σε συνδυασμό με την αυξημένη υπολογιστική ισχύ που παρέχουν οι σύγχρονοι μικροεπεξεργαστές που καθιστούν απαραίτητη και δυνατή τη χρήση εναλλαγξιμότητας λογισμικού (software or soft

redundancy), την χρήση δηλαδή αλγορίθμων που θα ελέγχουν την λειτουργία του αεροσκάφους και θα εντοπίζουν έγκαιρα τις όποιες βλάβες. Η έγκαιρη αυτή αναγνώριση βλαβών θα πρέπει φυσικά να συνοδεύεται από έναν μηχανισμό προσαρμογής του συστήματος ελέγχου ώστε να είναι δυνατή η διατήρηση του συστήματος σε ελέγξιμη πτήση και η συνέχιση της αποστολής.

Η μέχρι τώρα χρήση των μη επανδρωμένων αεροσκαφών καταδεικνύει ως κύριους παράγοντες πρόκλησης ατυχημάτων την απώλεια αντίληψης κατάστασης (situation awareness) εκ μέρους των χειριστών και τις βλάβες στα συστήματα προώθησης (κινητήρας) και αυτομάτου ελέγχου (πηδάλια, αισθητήρες). Στην συγκεκριμένη εργασία έχει δοθεί ιδιαίτερη έμφαση στην τελευταία κατηγορία βλαβών και συγκεκριμένα στις βλάβες των πηδαλίων ελέγχου καθώς είναι οι πιο κρίσιμες αφού αλλάζουν άρδην τη συμπεριφορά του συστήματος και δεν είναι εύκολα αντιμετωπίσιμες με την προσθήκη επιπλέον υλικού όπως στην περίπτωση των βλαβών σε αισθητήρες. Πράγματι η εισαγωγή της τεχνολογίας MEMS έχει κάνει δυνατή την ενσωμάτωση πολλών φθηνών αισθητήρων και την κοινή χρήση τους μέσω αλγορίθμων (sensor fusion). Κάτι τέτοιο δεν είναι δυνατό για την περίπτωση των πηδαλίων ελέγχου (αν και αεροσκάφη με πολλαπλά πηδάκια ελέγχου έχουν αρχίσει να σχεδιάζονται). Οι βλάβες που εξετάζονται καλύπτουν όλες τις πιθανές βλάβες πηδαλίων όπως η ακινητοποίηση σε ορισμένη θέση (Stuck failure), η ολική απώλεια πηδαλίου (floating actuator) και η δομική βλάβη πηδαλίου.

Σε αντίθεση με ήδη υπάρχουσες μελέτες στη βιβλιογραφία, η μοντελοποίηση των βλαβών είναι πολύ πιο ρεαλιστική. Οι βλάβες εισάγονται αυτοτελώς σε κάθε επιφάνεια ελέγχου (π.χ δεξί elevator) και όχι στον συνδυασμό των πηδαλίων (δεξί και αριστερό elevator συγχρόνως). Επίσης οι δομικές βλάβες έχουν προσομοιωθεί ως αλλαγές στους αεροδυναμικούς συντελεστές της συγκεκριμένης επιφάνειας ελέγχου σε συνδυασμό με αλλαγές σε άλλους αεροδυναμικούς συντελεστές και με την εισαγωγή νέων όρων που συνδέουν τις ροπές μεταξύ των αξόνων (cross-coupling). Η εισαγωγή των

βλαβών αυτών γίνεται απευθείας στο μη-γραμμικό μοντέλο του αεροσκάφους και όχι σε κάποια γραμμικοποιημένη προσέγγισή του γύρω από κάποιο σημείο ισορροπίας (equilibrium point). Επίσης στο αεροσκάφος δεν προστίθεται επιπλέον εξοπλισμός εκτός του συμβατικού (π.χ. δεν περιλαμβάνονται επιπλέον αισθητήρες θέσης των πηδαλίων).

Το σύστημα αναγνώρισης βλαβών είναι μη-γραμμικό και αποτελεί τον συνδυασμό δύο τεχνικών: της αναγνώρισης παραμέτρων (parameter identification) και της εκτίμησης πολλαπλών μοντέλων (Multiple Model Adaptive Estimation). Η τελευταία αποτελείται από μια συστοιχία επαναληπτικών Extended Kalman Filters (EKFs) που αναγνωρίζει την περίπτωση κάποιας ακινητοποιημένης επιφάνειας ελέγχου με την εφαρμογή μιας διαδικασίας probability ratio test και παρέχει μια φιλτραρισμένη (smoothed) εκτίμηση του διανύσματος κατάστασης και κυρίως των γωνιακών ταχυτήτων. Η εκτίμηση των γωνιακών ταχυτήτων με επίπεδο θορύβου σαφώς μικρότερο αυτού που επιτυγχάνεται μέσω των χαμηλής ποιότητας rate gyros, κάνει δυνατό τον υπολογισμό μέσω κάποια τεχνικής διαφόρισης των γωνιακών επιταχύνσεων που χρησιμοποιούνται στον αλγόριθμο αναγνώρισης παραμέτρων. Η αναγνώριση παραμέτρων γίνεται σε πραγματικό χρόνο μέσω ενός αλγορίθμου ελαχίστων τετραγώνων (Least squares estimation). Η απότομη μεταβολή των αεροδυναμικών συντελεστών που αναγνωρίζονται αποτελεί μια φυσική ένδειξη δομικής βλάβης των πτερυγίων ελέγχου. Ο συντονισμός και η επεξεργασία των αποτελεσμάτων των δύο παραπάνω αλγορίθμων γίνεται σε έναν κεντρικό αλγόριθμο επίβλεψης (supervision module) όπου τα ευρήματα αξιολογούνται με διάφορους τρόπους (χρήση σταθερών ορίων (thresholds) και ασαφή λογική (Fuzzy logic) πριν εξαχθεί η πληροφορία της αναγνώρισης βλάβης και των χαρακτηριστικών της. Το ανωτέρω σύστημα αναγνώρισης βλαβών αξιολογήθηκε τόσο σε ανοιχτό (open-loop) όσο και σε κλειστό βρόγχο (closed-loop), παρουσία ενός συστήματος

ελέγχου που θα μπορούσε να κρύψει τα συμπτώματα των βλαβών.

Με βάση τα ευρήματα του συστήματος αναγνώρισης βλαβών, το σύστημα ελέγχου σχεδιάστηκε ώστε να λειτουργεί τόσο σε συνηθισμένες (nominal) όσο και σε έκτακτες συνθήκες παρουσία βλαβών. Το σύστημα αρχικά σχεδιάστηκε με βάση τη μέθοδο της μη-γραμμικής δυναμικής αντιστροφής (Non-linear Dynamic Inversion, "NDI"). Η μέθοδος αυτή μπορεί εύκολα να προσαρμοστεί σε περίπτωση βλάβης με την αλλαγή των παραμέτρων με τις νέες που έχουν εκτιμηθεί στην περίπτωση των δομικών βλαβών ενώ στην περίπτωση που η βλάβη είναι ακινητοποιημένη επιφάνεια ελέγχου, η επίδρασή της υπολογίζεται στον νόμο ελέγχου με τον ίδιο τρόπο που υπολογίζεται η επίδραση των διαφόρων μεταβλητών κατάστασης και η απαιτούμενη ενέργεια καταμερίζεται στις εναπομείναντες υγιείς (healthy) επιφάνειες. Η μέθοδος NDI παρέχει πολλά πλεονεκτήματα στην περίπτωση του FTC, λόγω της εύκολης προσαρμογής των συντελεστών της (που στην πραγματικότητα είναι οι αεροδυναμικοί συντελεστές του αεροσκάφους), του καταμερισμού ελέγχου (control allocation) που φυσικά εμπεριέχει και της δυνατότητας προσαρμογής της επιθυμητής απόδοσης του συστήματος μέσω γραμμικών μεθόδων στον εξωτερικό βρόγχο (outer loop). Η επίδοση του συστήματος αξιολογήθηκε μέσω προσομοιώσεων σε συνάρτηση με εκείνη του συστήματος αναγνώρισης βλαβών αφού σε αντίθεση με άλλες έρευνες στη βιβλιογραφία, η πληροφορία της βλάβης παρέχεται απευθείας από το σύστημα αναγνώρισης και η επίδοση του νόμου ελέγχου εξετάστηκε και κατά την περίοδο προσαρμογής δηλαδή τη χρονική καθυστέρηση αναγνώρισης της βλάβης και ακριβούς εκτίμησης των παραμέτρων της (transient period). Από τις προσομοιώσεις φάνηκε ότι το σύστημα αναγνώρισης βλάβης μπορεί να αναγνωρίσει την ύπαρξη βλάβης (Fault Detection) σε πολύ μικρό χρόνο (350ms) ενώ η ακριβής εκτίμηση των παραμέτρων της βλάβης απαιτεί χρόνο ίσο με 1 second (για μικρές βλάβες). Το σύστημα ελέγχου μπορεί να διατηρήσει τον έλεγχο του

αεροσκάφους και να επιτύχει ικανοποιητική παρακολούθηση των επιθυμητών εντολών παρουσία βλαβών. Τέλος προκειμένου να αντιμετωπιστεί το μεγάλο μειονέκτημα του NDI, δηλαδή η αυξημένη ευαισθησία του στα λάθη μοντελοποίησης (άρα και στις εκτιμώμενες παραμέτρους της βλάβης), εφαρμόστηκε μια τροποποιημένη εκδοχή του NDI, το INDI (Incremental Non-linear Dynamic Inversion) παρουσιάστηκε και αξιολογήθηκε μέσω προσομοιώσεων δείχνοντας αυξημένη ανθεκτικότητα στα σφάλματα των εκτιμώμενων παραμέτρων βλάβης.

Στο κεφάλαιο 1, δίνεται μια εισαγωγική περιγραφή του ελέγχου παρουσία βλαβών και της ανάγκης εφαρμογής του στην αεροδιαστημική τεχνολογία τόσο σε επανδρωμένα όσο και σε μη-επανδρωμένα αεροσκάφη. Τέλος περιγράφεται ο σκοπός της μεταπτυχιακής εργασίας.

Στο δεύτερο κεφάλαιο, περιγράφεται αναλυτικά το μοντέλο του μη-επανδρωμένου αεροσκάφους (Aerobatic UAV του πανεπιστημίου ΕΤΗ της Ζυρίχης). Περιγράφονται επίσης τα μοντέλα αισθητήρων και επενεργητών (actuators) που χρησιμοποιήθηκαν στην προσομοίωση καθώς και τα μοντέλα των περιβαλλοντικών διαταραχών (turbulence).

Στο κεφάλαιο 3, γίνεται αναφορά στις διαφορετικές βλάβες που είναι δυνατό να παρουσιάσουν γενικά σε ένα δυναμικό σύστημα και εξειδικεύεται ο τρόπος μοντελοποίησης και προσομοίωσής τους. Τέλος παρουσιάζεται μέσω ενός απλού παραδείγματος η κρισιμότητα των βλαβών των επενεργητών στην ευστάθεια των ελεγχόμενων συστημάτων.

Στο κεφάλαιο 4 γίνεται μια βιβλιογραφική αναδρομή στην αναγνώριση βλαβών και στον έλεγχο παρουσία βλαβών. Οι διάφορες τεχνικές παρουσιάζονται συνοπτικά μαζί με εφαρμογές τους στον τομέα των επανδρωμένων και μη αεροσκαφών.

Τα κεφάλαια 5, 6 αποτελούν το κύριο μέρος της εργασίας όπου παρουσιάζεται το σύστημα αναγνώρισης βλαβών και το σύστημα ελέγχου πτήσης παρουσία βλαβών. Στο πρώτο, αναλύεται η σχεδίαση του συστήματος αναγνώρισης βλαβών

και ελέγχεται η απόδοσή του σε ανοιχτό βρόγχο (open-loop), χωρίς την παρουσία ελεγκτή. Στο δεύτερο ακολουθεί η σχεδίαση του νόμου ελέγχου για τις διάφορες περιπτώσεις βλαβών των επιφανειών ελέγχου και αξιολογείται η επίδοση ολόκληρου του συστήματος (closed-loop simulation).

Τέλος το κεφάλαιο 7 περιέχει μια συνοπτική ανακεφαλαίωση της εργασίας όπου παρατίθενται τα συμπεράσματα, παρατηρήσεις και προτάσεις για περαιτέρω έρευνα.

Στην εργασία αυτή χρησιμοποιήθηκε ένας αριθμός εργαλείων και αποτελεσμάτων άλλων εργασιών. Για την μοντελοποίηση του αεροσκάφους χρησιμοποιήθηκαν δεδομένα που αναφέρονται στην αναφορά [10] (μοντελοποίηση δυνάμεων και ροπών). Για την προσομοίωση χρησιμοποιήθηκε το Flight Dynamics and Control Toolbox (FDC) [11] που αναπτύχθηκε στο TU Delf και διατίθεται ελεύθερα στο internet. Το toolbox περιλαμβάνει το μοντέλο του αεροσκάφους Beaver και τροποποιήθηκε σημαντικά ώστε να είναι δυνατή η μοντελοποίηση του υπό μελέτη UAV και η εισαγωγή δομικών βλαβών με τροποποίηση των αεροδυναμικών συντελεστών. Οι δομικές αυτές βλάβες μπορούν να εισαχθούν στην επιλεγμένη χρονική στιγμή και το είδος της βλάβης μπορεί εύκολα να τροποποιηθεί μεταξύ των προσομοιώσεων.

Επίσης προστέθηκαν κατάλληλα μοντέλα επενεργητών και αισθητήρων και αναπτύχθηκε ένα εργαλείο εισαγωγής βλαβών (δυνατότητα ακινητοποίησης σε τυχαίες χρονικές στιγμές και τυχαίες θέσεις, προσομοίωση floating actuator κ.α).

Κατόπιν αναπτύχθηκαν ως sfunctions τα έξι EKFs και ο αλγόριθμος που εκτελεί επαναληπτικά probability ratio test υπολογίζοντας τη σχετική πιθανότητα κάθε ενός εκ των έξι σεναρίων (πέντε για βλάβη σε έναν επενεργητή και ένα που αντιπροσωπεύει την φυσιολογική λειτουργία). Αναπτύχθηκε επίσης ως sfunction ο αλγόριθμος που εκτελεί εκτίμηση παραμέτρων σε πραγματικό χρόνο καθώς και η λογική της μονάδας επίβλεψης (supervision module). Τέλος αναπτύχθηκε η λογική ελέγχου μέσω απλών PID ελεγκτών (simulink) και Non-

linear Dynamic Inversion (NDI) (sfunction). Όλα τα αρχεία που είναι απαραίτητα για τις προσομοιώσεις περιλαμβάνονται στο συνημμένο CD.

Chapter 1 – Introduction

1.1 Why Fault –Tolerant Control?

Nowadays, control systems are everywhere in our life. They are all around us, often remaining invisible for the eye of most of us. They are in our kitchens, in our DVD players and computers. They are driving the elevators, we have them in our cars, ships, aircraft and spacecraft. Control systems are present in every industry, they are used to control chemical reactors, distillation columns and nuclear power plants. They are constantly and inexhaustibly working, making our lives more comfortable and more pleasant...until the system fails.

Faults in technological systems are events that happen rarely, often at unexpected moments of time. In [1] (Blanke et al., 2006), the following definition of a fault is made:

“Fault in a dynamical system is an un-permitted deviation of the system structure or the system parameters from the nominal situation.”

It is clear from the above definition that a fault is different than a failure. A system can possibly tolerate a fault, however this can eventually lead to a complete failure.

Faults are difficult to foresee and prevent. Their further development into overall system failures may lead to consequences that take different forms and scales, ranging from having to spend 50 euros for a new coffee machine to enormous economical and human losses in safety-critical systems like aircrafts and nuclear power plants.

The idea of Fault-tolerant control can be stated as follows [1]:

“Fault-Tolerant control is a collection of techniques and practices that aim to prevent a fault from causing a failure at the system level.”

Fault tolerant control is an emerging research field in control engineering which is trying to design control systems which can tolerate component malfunctions, while maintaining desirable (or acceptable) performance and stability properties.

1.2 Motivation for Fault –Tolerant Flight Control

Within the aviation community, especially for commercial transport aircraft design, all developments focus on ensuring and improving the required safety levels and reducing the risks that critical failures occur.

The flight control system has been identified as a safety-critical system by the aerospace industry in the sense that catastrophic consequences can result from its failures, such as a control surface runaway (such as a rudder or horizontal stabilizer), loss of control on the pitch axis, lack of control after an engine burst or an oscillatory failure at a frequency critical to the structure. All these failures must be extremely improbable, i.e. with a probability of less than 10^{-9} per flight hour taking under consideration additional qualitative requirements. Specifically for flight control systems, it is required that a catastrophic consequence must not be due to a single failure or a control surface jam or a pilot control jam. This qualitative requirement is on top of the probabilistic assessment according to Federal Aviation Administration Regulations Part 23, 25 and 27 for civil aircraft.

In order to be compliant with Airworthiness requirements for aircraft certification, aerospace industries have invested in the implementation of safety integrity analysis methods like those presented in [2] (Isermann et al. 2002). Safety and reliability are generally achieved by a combination of

- Fault avoidance
- Fault removal
- Fault tolerance
- Fault detection and diagnosis

Fault avoidance and removal has to be accomplished mainly during the design and testing phase. The effects of faults on the reliability and safety are investigated based on a range of methods that include

- Reliability analysis
- Event tree analysis and fault tree analysis
- Failure mode and effect analysis
- Hazard analysis
- Risk classification

After the faults and their effects are identified, modification of the system or equipment specifications (e.g reliability specifications) or system architecture (e.g redundancy) can be applied to ensure that the possible failures are removed or that the risk that they impose to the system is reduced to an acceptable level. The unavoidable failures have to be covered by scheduled maintenance and online supervision and safety methods during operation.

Every industry applies its own rules during the design phase to accomplish safety requirements like the V-cycle applied by Airbus [3] (Gupil Phillipe, 2009). Generally, fault tolerance is achieved by:

- A stringent development process both for software and hardware. This development process has to follow specific guidelines that ensure quality assurance and tractability. It is accompanied by system safety assessment to assess the effect of faults and impose modifications on system architecture (e.g degree of redundancy)
- Hardware (and software) redundancy. The use of multiple measurement sensors, actuator devices, power sources and computers in static or dynamic redundancy configuration (usually triplex or quadruple) is common. These redundant components are selected by voter mechanisms or by monitoring and reconfiguration mechanisms as shown in figure 1.1 [2]. In this way monitoring of the system components and automatic management following a failure (reconfiguration) is possible in order to maintain performance or achieve graceful degradation. Redundancy is used also for the software with different software components realizing the control laws and arranged in different levels of complexity.
- Dissimilarity is also a very important point to ensure fault tolerance. For example all Airbus aircraft have at least two types of computer: a primary and a secondary. Their hardware and software are different and they are not developed by the same teams. Moreover, the software components are developed using different software tools and languages.

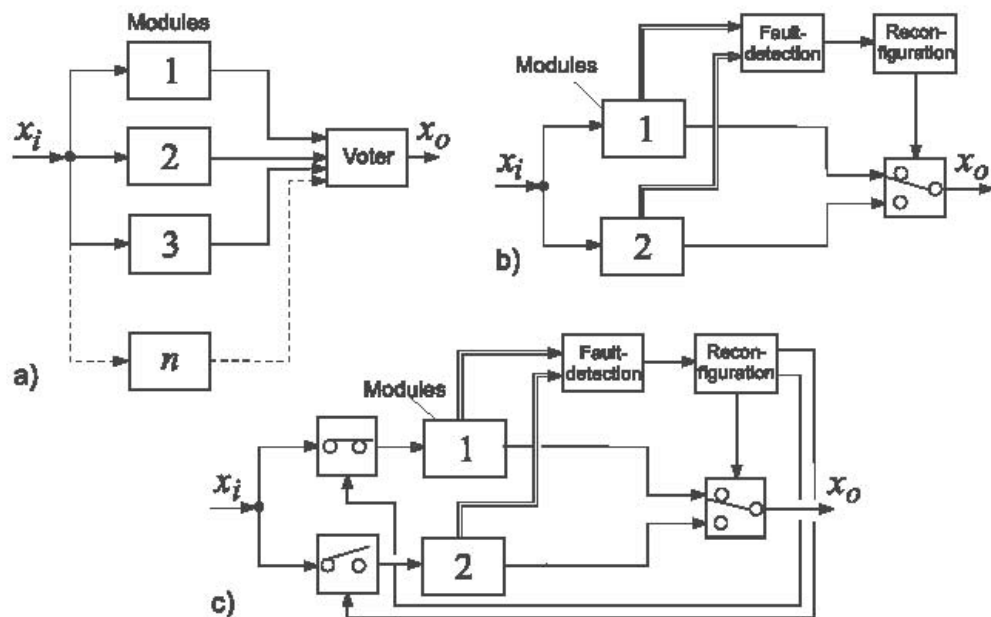


Figure 1.1 Fault-tolerant schemes for electronic hardware [2].

- (a) Static redundancy: multiple redundant modules with majority voting and fault masking, m out of n systems (all modules are active).
- (b) Dynamic redundancy: Standby module that is continuously active (hot standby)
- (c) Dynamic redundancy: Standby module that is inactive (cold standby)

- Installation segregation: critical components (like computers) are not physically installed at the same place on the aircraft, to avoid total loss in the case of any damage. The same reasoning leads to segregation of hydraulic and electrical routes.

All these methods are applied in the design stage and a thorough validation/verification stage follows, ranging from computer simulation, hardware in the loop tests (“iron bird”) and SIL (System Level Integration laboratory), to flight tests. Every industry of course applies its own «golden rules» and a survey of current fault tolerant flight control systems for both civil and military aircraft can be found in [4]. Figure 1.2 displays the flight control system architecture of the Eurofighter Typhoon aircraft which is typical for other military aircraft (F-16 etc). It uses quad-redundant processing and sensor modules as well as four independent signal channels.

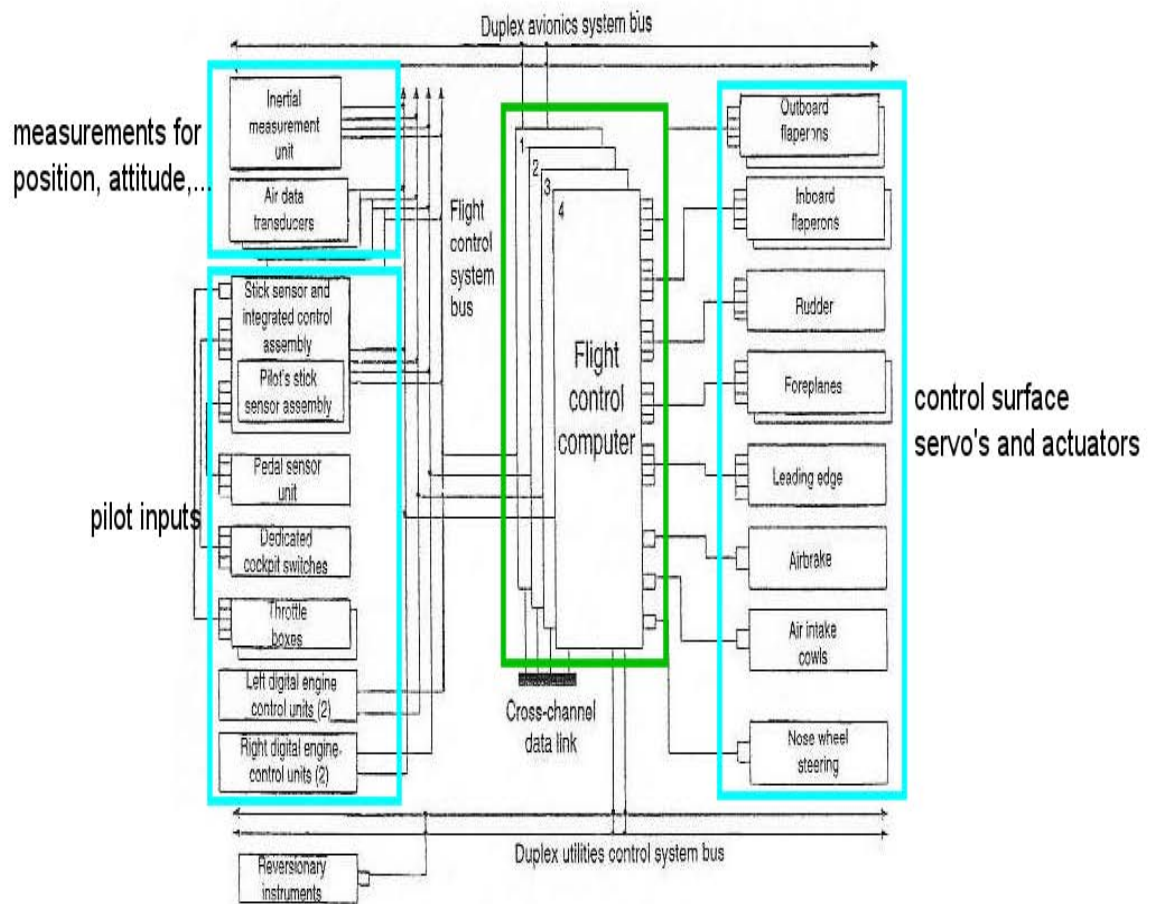


Figure 1.2 Flight Control System Architecture of the Eurofighter Typhoon [4].

Despite the stringent development process, accidents in aerospace systems are not impossible. On the contrary fatal accidents of civil aviation alone are more than 20 every year! Figure 1.3, represents some recent worldwide civil aviation safety statistics [5] (Civil Aviation Authority of the Netherlands, 2008).

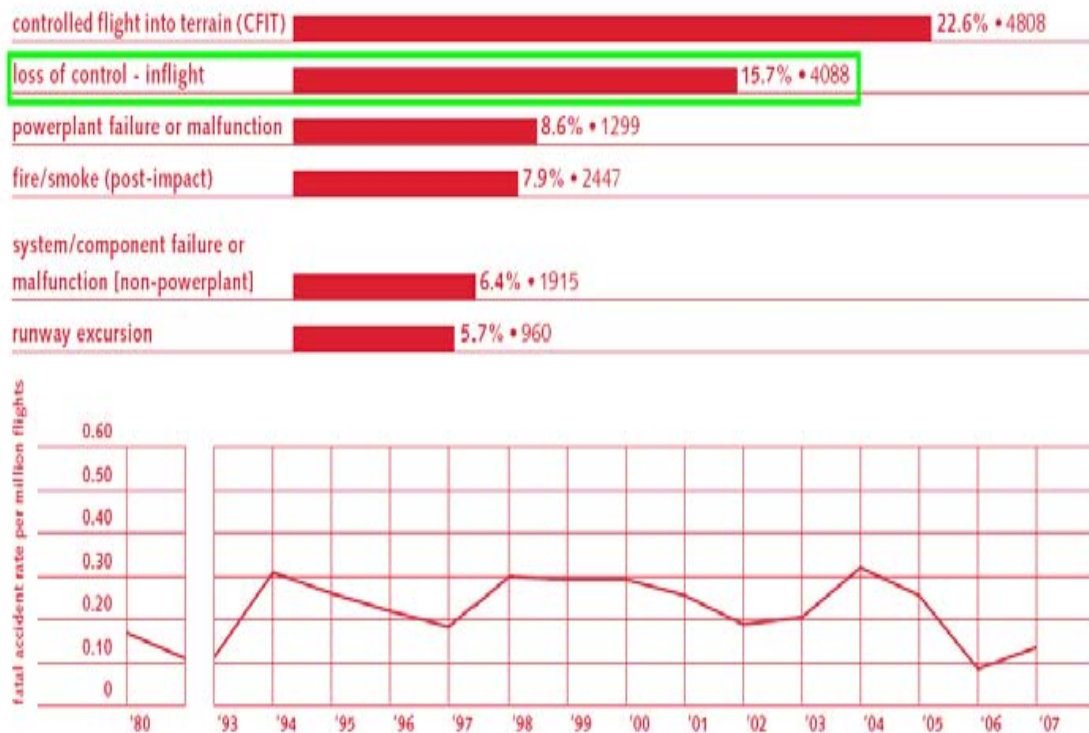


Figure 1.3 Accidents statistics [5].

The above mentioned study indicates two major categories of accidents which can be attributed to a common initial event, “controlled flight into terrain” where an aircraft, despite being fully controllable and under control, hits terrain due to the loss of the situation awareness of the crew, counting for as much as 23% of all the accidents. This percentage is decreasing over the years, thanks to the enormous international attention with respect to crew resource management training and the development and implementation of new systems in the cockpit. The second major category is “loss of control in flight”, which can be attributed to mistakes made by the pilot or a technical malfunctioning. This category counts for 16% of all aircraft accidents and is not decreasing. Loss of control during flight is one of the motivating factors towards fault tolerant control: the idea being, to increase the “fly ability” of the aircraft in the event of faults, failures and airframe damage.

Learning from previous incidents, where pilots successfully landed crippled aircraft – such as Flight 232 in Sioux City, Iowa 1989¹, the Kalita Air freighter in Detroit, Michigan, October 2004 (figure 1.5)² and the DHL freighter incident in Baghdad, November

¹ Flight 232 suffered tail engine failure that caused the total loss of hydraulics [6]

² The freighter shed engine No. 1 but the crew managed to safely land without any casualties

2003 (figure 1.4)³ – it is evident that in many cases, the damaged or faulty aircraft is still “flyable”, controllable and some level of performance still can be achieved, sufficient to allow the pilot to safely land the aircraft. It is thus believed that a significant part of that 16% of accidents could have been prevented.

Furthermore, the existing approaches to fault tolerance with the use of multiple redundant components, imposes a heavy cost on the design and especially the operation and maintenance of modern aircrafts. The current trend to civil aviation is the reduction of weight and the minimization of fuel consumption something that does not favor multiple hardware redundancy. New methods to achieve fault tolerance have to be implemented.



Figure 1.4 DHL A300B4 emergency landing after being hit by a missile in Baghdad, 2003.

³ The DHL A300B4 was hit by a missile on its left wing and lost all hydraulics, but landed safely [6]



Figure 1.5 Kalila Air emergency landing after losing one engine, 2004.

1.3 Fault –Tolerant Flight Control for UAVs

The increasing need of avoiding the exposure of humans to “dull, dirty and dangerous” missions in conjunction with the sustained multi-disciplinary technological progress in the past two decades are the main reasons behind the exponential growth rate in the development, deployment and operation of unmanned autonomous systems (UAS) [7], [9].

Unmanned aerial vehicles (UAVs) have been around and in service since the 1990s and are going to be routinely used for a wide range of tasks such as:

- Sea border searches from the air
- Search and rescue
- Border patrols, homeland security, law enforcement, monitoring of drug trafficking
- Monitoring and control of road traffic and transportation
- Crop yield prediction, drought monitoring, spraying of pesticides
- Inspection of power lines, bridges and barrages
- Observation of oil and gas pipelines
- Forest monitoring, fire detection, firefighting
- Relaying and broadcasting of mobile communication
- Tactical reconnaissance and operational support
- Environmental and climate research: monitoring of air quality, meteorological studies and predictions

New generations of UAVs will play increasingly important role in future military and civil operations. They will be designed to

achieve their mission not only with increased efficiency, but also with more safety and security. However there are a number of significant challenges associated with the development of an advanced control system for these vehicles [8].

First, because the UAVs will be exploited to perform tasks that would otherwise risk the safety of flight crews of manned aircraft, there is an increased probability of damage to the vehicle resulting from extreme operating conditions, hostile actions, etc. This underscores the need for a reliable system design that can accommodate significant changes in system behavior from a wide variety of sources. Flight control and power/propulsion failures are dominant failure modes according to accumulated data as shown in figure 1.6 [7].

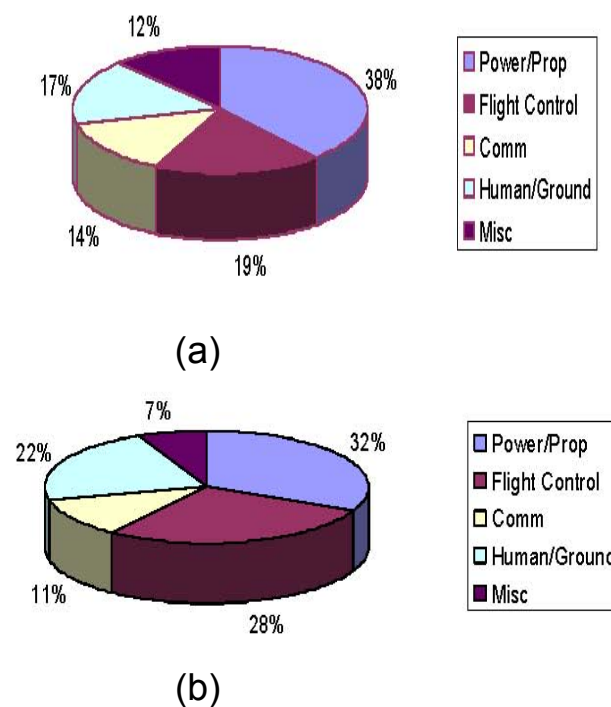


Figure 1.6 Failure sources for UAVs: (a) US military based on 194000 flight hours and (b) IAI military based on 100000 flight hours.

Furthermore the requirement that the UAV must operate in close proximity to humans further emphasizes the need for a reliable system design. As mentioned in [7], there is a trend to incorporate future UAVs into national and international airspace. This goal will impose strict requirements to reliability similar to those of current civil and military aircraft. However in UAVs there are additional cost, weight and payload constraints that negate the application of current techniques that rely on multiple hardware

redundancy. As shown in figure 1.7 [7], these constraints lead to higher mishap rates for smaller (and cheaper) UAVs, while for bigger ones, reliability is comparable to that of modern military aircraft. This can be attributed to the lower cost-lower performance equipment (sensors/actuators) used in lower cost configurations.

Moreover, the lack of a pilot in the cockpit makes very difficult the handling of UAVs in case of failures. Situation awareness is not possible or severely reduced and the response of the ground crew is usually delayed and/or mistaken. This makes essential the existence of reliable on-board fault detection and identification system as well as a fast and efficient fault tolerant control system that could maintain the UAV in flight and report the failures to the ground crew allowing decision making.

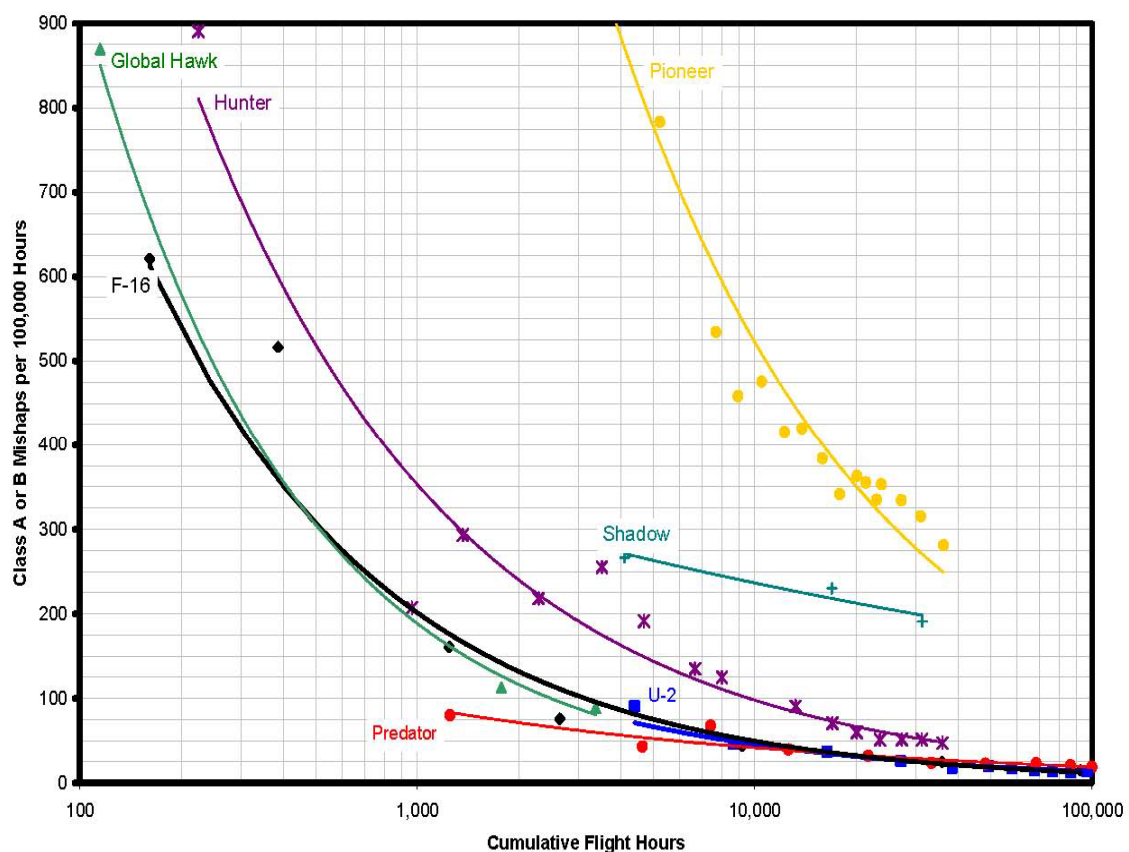


Figure 1.7 Mishap Rate Comparisons [7].

Second, because many UAV systems are expected to cost less than manned systems, it is unlikely that developers will have the resources to collect extensive wind-tunnel and flight test data of the caliber typically found during manned flight vehicle development. Thus, the model available for UAV development will necessarily contain larger uncertainties, which compels the control engineer to compromise performance in favor of robustness. The

same requirement of robustness to the model uncertainties are imposed to the fault detection and identification system of the UAV if a model based approach is selected.

Finally, because of the emerging requirements to increase autonomy, the controller must be augmented with a very sophisticated guidance and autopilot design that not only cruises, climbs and changes heading, but is capable of generating trajectories and perform complex and agile maneuvers that would normally be performed by a pilot, without the risk of losing control of the vehicle. The resulting controller should be capable of adapting to new conditions and restrictions such as actuator and sensor faults and structural damages.

1.4 Purpose of the thesis

The purpose of this thesis is the development of a fault tolerant flight control and guidance system for a non-linear fixed wing air vehicle.

The design and testing is going to be performed in Matlab/Simulink. A simulation environment for the simulation and testing of fault tolerant flight control systems will be created. This environment will be based on Flight Dynamics and Control Toolbox (FDC) [11] and will include a 6-DOF nonlinear UAV model developed at ETH Zurich [10] (Figure 1.8).

Our goal is:

- To design a reliable and efficient fault detection system capable of isolating faults and providing fault estimates and evaluate its performance and complexity.
- The fault detection and isolation system will be able to detect and isolate all types of actuator faults (stuck surfaces and loss of control effectiveness) and will treat every control surface individually (contrary to the current literature where loss of effectiveness is treated on pairs of effectors only).
- To use existing redundancy of the aircraft model to accommodate any sensor or actuator failure by designing a fault tolerant controller. The faults that will be considered are focused on actuator failures (both stuck or floating actuators and loss of control

- To assess the performance of the hall system and the added reliability.

The development will focus on:

- Robustness of the Fault Detection and Identification (FDI) system and the controller to modeling uncertainties
- The limitations in hardware/payload due to weight/costs restrictions. Because of the above, no extra sensor or actuator is going to be implemented to increase survivability
- The coupling of the controller/FDI systems to a real time model identification module capable of providing local updates to the aerodynamic model data if the robustness to the specific parameter is found to be critical
- Effective accommodation of every single actuator failure
- Effective accommodation of structural failures of a limited extend



Figure 1.8 Aerobatic UAV developed by ETH Zurich [10].

Chapter 2 – UAV Simulation Model

2.1 Nonlinear Model of the UAV

This chapter presents the axes, the frames and the non-linear model of the aircrafts used in this thesis. The axes are typically used in almost every aerospace application. The derivation of the model is based on the rigid body equations of motion and can be considered valid for aircraft as well as fixed wing UAV aircraft as long as flexible modes are neglected. A more detailed description of the derivation of the equations can be found in standard textbooks on aerodynamics and flight control [12], [13]. The detailed modeling of the aerodynamic and propulsion forces can be found in the Appendix and section 2.5.2 respectively.

2.2 Definition of the Frames

2.2.1 Navigation Frame

The navigation frame is considered an “inertial frame of reference” (it does not move) and is attached to the earth’s local tangent plane. Its orientation is North-East-Down (x_n, y_n, z_n). When the airplane is on the ground before taking off, the origin of the navigation frame, O_n , is initialized by the position of the airplane’s center of mass.

2.2.2 Body Frame

The body frame is right-handed orthogonal coordinate frame (x_b, y_b, z_b), attached to the aircraft body and moves with it. The origin O_b is located at the aircraft’s center of the mass. The positive x axis points forward along the aircraft’s longitudinal axis, the positive y axis is directed along the right wing and the positive z axis is normal to the x and y axes pointing downwards.

2.2.3 Euler Angles

In order to relate the various vectors expressed in the two reference frames described above, three Euler angles are widely used. As shown in Figs. 2.1 and 2.2, the navigation coordinate frame is first transformed into the intermediate frame 1 via a

rotation about the z_n axis by the angle ψ , which defines the aircraft's heading. This is followed by a rotation about the new y_1 axis by an angle θ (*pitch angle*), which defines the aircraft's elevation. Finally, the aircraft bank angle (roll angle), ϕ , defines the rotation about the new x_2 axis. Figure 2.3 shows a 3D representation of the Euler angles describing the orientation of the body frame with respect to the navigation frame.

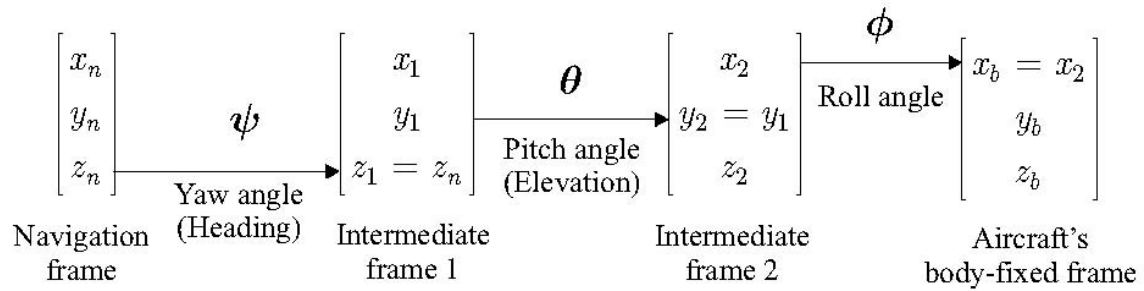


Figure 2.1 Euler angles

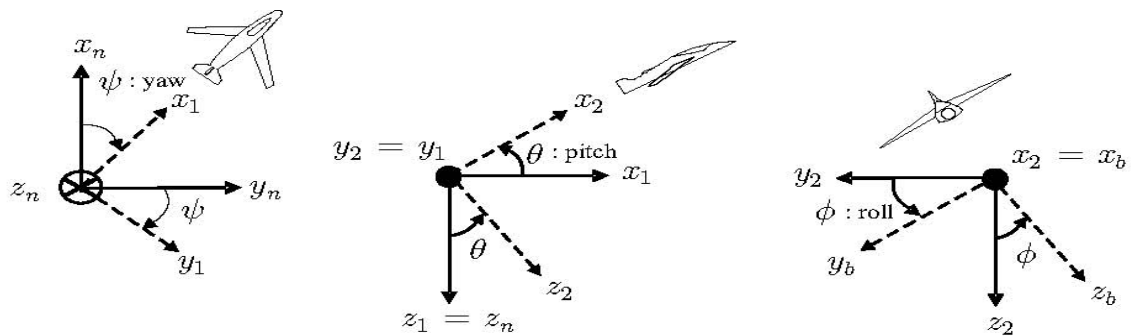


Figure 2.2 Euler angles and frame transformations

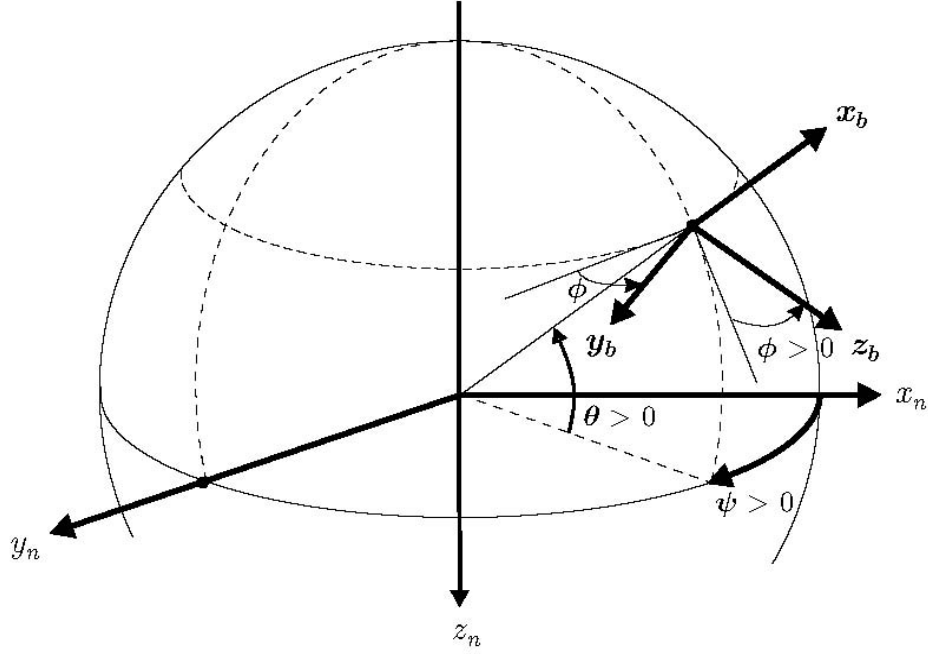


Figure 2.3 3D representations of the Euler angles

The attitude transformation matrix (also called direction cosine matrix) is necessary to transform vectors and point coordinates from the aircraft's body-fixed frame (b) to the navigation frame (n) and vice versa. The direction cosine matrix C_b^n transforms the vector \mathbf{A} expressed in the navigation frame \mathbf{A}^n into a vector expressed in the aircraft's body-fixed frame \mathbf{A}^b as follows:

$$\mathbf{A}^b = C_b^n \mathbf{A}^n \quad (2.1)$$

With

$$C_b^n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

In a similar fashion, the direction cosine matrix C_n^b that transforms the vector \mathbf{A} expressed in the body-fixed frame \mathbf{A}^b into a vector expressed in the navigation frame \mathbf{A}^n , is:

$$C_n^b = (C_b^n)^{-1} = (C_b^n)^T \quad (2.3)$$

2.2.4 Quaternion representation

Due to certain limitations of the Euler angles representation, such as the ambiguity at 90° angles, quaternion representation is often used, especially in highly maneuvering aircraft and spacecraft systems.

Transformation between representations is possible and computationally simple. In [12] it is shown how elements of the quaternion can be expressed in terms of Euler angles and vice versa. The attitude transformation matrix C_b^n can be expressed with a quaternion representation as follows:

$$C_b^n = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & 1-2(q_1^2+q_3^2) & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & 1-2(q_1^2+q_2^2) \end{bmatrix} \quad (2.4)$$

2.2.5 Wind frame

The aerodynamic forces are created by the airflow acting on the airframe. The air flow is described by the airspeed vector \mathbf{V}_T . Its norm is $|\mathbf{V}_T|$ and its direction relative to the airframe is defined by two angles, namely the angle of attack α and the sideslip angle β .

As shown in figure 2.4, the angle of attack α is the angle between the projection of the airspeed vector \mathbf{V}_T onto the (x_b, z_b) plane and the x_b axis. The sideslip angle β is the angle between the projection of the airspeed vector \mathbf{V}_T onto the (x_b, z_b) plane and the airspeed vector itself. The wind axes coordinate system is such that the x_w axis points along the airspeed vector \mathbf{V}_T .

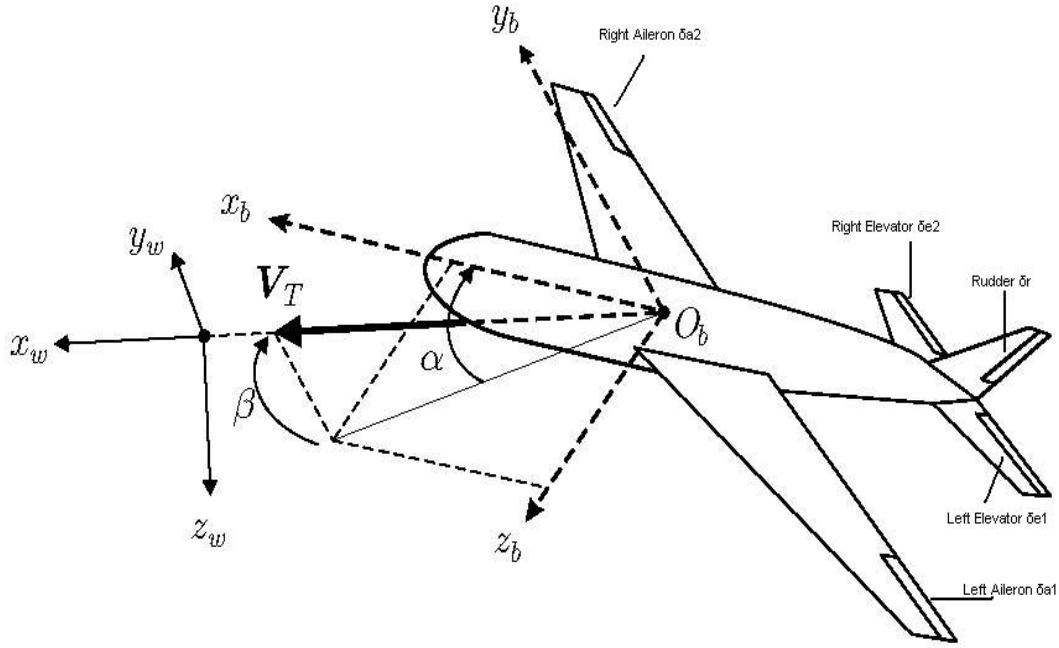


Figure 2.4 Angle of attack and sideslip angle definition

The rotation matrix C_b^w is necessary to transform vectors and point coordinates from the body fixed frame (b) to the wind frame (w) and vice versa according to the following formula:

$$A^w = C_b^w A^b \text{ or } A^b = (C_b^w)^T A^w = C_w^b A^w \quad (2.5)$$

with

$$C_b^w = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix} \quad (2.6)$$

2.3 Wind Disturbance

The unsteady nature of the atmosphere, affects the motion of the airplane and its flight path in relation to the ground. Wind disturbance can seriously impair the fault detection system since it introduces unknown (and unwanted) input to the system. That is why in our design and evaluation of fault tolerant control system the presence of wind will be included.

Wind disturbance can be modeled as a mix of a deterministic and a stochastic component [11]. The deterministic component can be created based on a constant wind speed and wind direction with the addition of a vertical wind velocity component. The change of horizontal wind speed with height has to be taken into account. The modeling of this component can be based on simple deterministic equations.

The stochastic component of the wind is known as turbulence. It is often regarded as a “random” process, although the evolution of turbulent flows are governed by the general Navier-Stokes equations (a set of deterministic, nonlinear coupled, partial differential equations). For simulation purposes it would be practical to model atmospheric turbulence as white noise passing through a linear, rational, “forming filter”, as shown in figure 2.5.



Figure 2.5 Modeling atmospheric turbulence as filtered white noise

Several wind models have been included into FDC Toolbox both for the deterministic component of the wind and the turbulent part. These, as well as a deeper description of wind and turbulence modeling can be found in [11].

2.4 Rigid Body Equations of Motion

The aircraft equations of motion can be derived from Newton’s laws, which state the connection between force and motion. These equations are valid for all aircraft as long as they can be considered as rigid bodies and their flexible modes are neglected. Their derivation can be found in a lot of books about aerodynamics and control [12], [13] and they will be briefly stated in this section.

To build the nonlinear state-space model we start with the basic force and moment equations that describe the change in translational and rotational velocities (i.e. translational and rotational accelerations)

$$\begin{aligned}
\dot{V} &= \frac{1}{m} (F_x \cos \alpha \cos \beta + F_y \sin \beta + F_z \sin \alpha \cos \beta) \\
\dot{\alpha} &= \frac{1}{V \cos \beta} \left\{ \frac{1}{m} (-F_x \sin \alpha + F_z \cos \alpha) \right\} + q - (p \cos \alpha + r \sin \alpha) \tan \beta \\
\dot{\beta} &= \frac{1}{V} \left\{ \frac{1}{m} (-F_x \cos \alpha \sin \beta + F_y \cos \beta - F_z \sin \alpha \sin \beta) \right\} + p \sin \alpha - r \cos \alpha \quad (2.7) \\
\dot{p} &= P_{pp} p^2 + P_{pq} pq + P_{pr} pr + P_{qq} q^2 + P_{qr} qr + P_{rr} r^2 + P_l L + P_m M + P_n N + \dot{p}' \\
\dot{q} &= Q_{pp} p^2 + Q_{pq} pq + Q_{pr} pr + Q_{qq} q^2 + Q_{qr} qr + Q_{rr} r^2 + Q_l L + Q_m M + Q_n N + \dot{q}' \\
\dot{r} &= R_{pp} p^2 + R_{pq} pq + R_{pr} pr + R_{qq} q^2 + R_{qr} qr + R_{rr} r^2 + R_l L + R_m M + R_n N + \dot{r}'
\end{aligned}$$

The variables V (true airspeed), α (angle of attack), β (sideslip angle), p (yaw rate), q (pitch rate) and r (roll rate), which represent the linear and angular velocities of the aircraft, can be regarded as the state variables for this model. Sometimes the body-axes velocities components u , v , w are used instead as state variables. However the use of V , α , β is more convenient and more appropriate as they represent quantities that are usually measured by sensors on most aircraft. F_x , F_y , F_z and L , M , N are the total forces and moments on the aircraft body. These are the result of thrust, aerodynamic forces and wind forces. The coefficients of the last three equations (inertia coefficients) are summarized in table 2.1 while figure 2.6 is a graphical representation of the external forces and moments and the linear and rotational velocity components of the airplane in relation to the body-fixed reference frame.

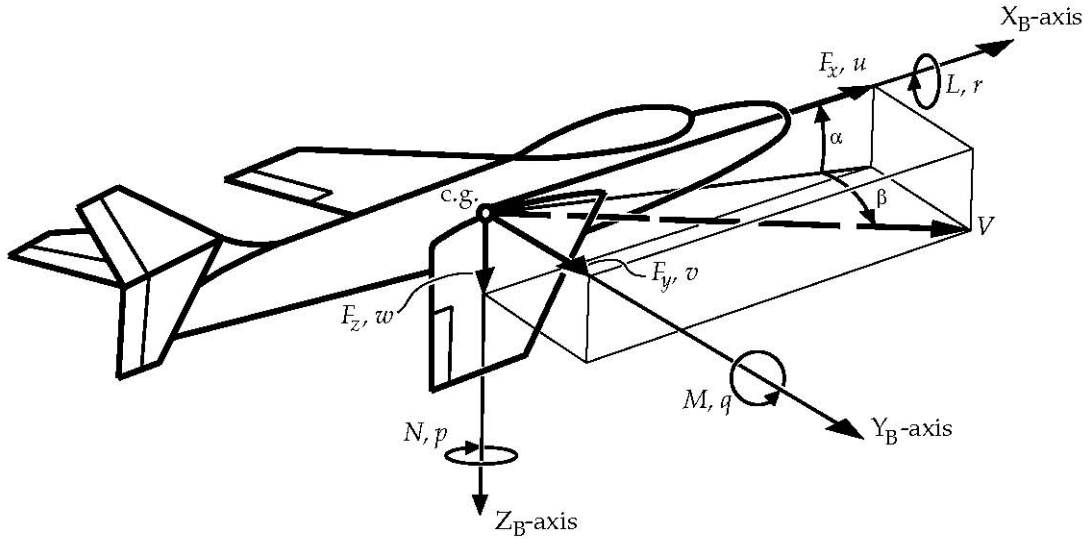


Figure 2.6 Orientation of the linear and angular velocity components, external forces and moments, angle of attack and sideslip angle in relation to the body-fixed reference frame of the aircraft.

The attitude of the airplane and its altitude are needed to determine the gravitational, aerodynamic and propulsive forces and moments. This means that the model needs to be extended with the equations of the Euler angles and the altitude. The aircraft's horizontal coordinates are not needed to solve the equations of motion but they are included for practical purposes. This yields an additional six state equations:

$$\begin{aligned}
 \dot{\psi} &= \frac{q \sin \varphi + r \cos \varphi}{\cos \theta} \\
 \dot{\theta} &= q \cos \varphi - r \sin \varphi \\
 \dot{\varphi} &= p + (q \sin \varphi + r \cos \varphi) \tan \theta = p + \dot{\psi} \sin \theta \\
 \dot{x}_e &= \{u_e \cos \theta + (v_e \sin \varphi + w_e \cos \varphi) \sin \theta\} \cos \psi - (v_e \cos \varphi - w_e \sin \varphi) \sin \psi \\
 \dot{y}_e &= \{u_e \cos \theta + (v_e \sin \varphi + w_e \cos \varphi) \sin \theta\} \sin \psi + (v_e \cos \varphi - w_e \sin \varphi) \cos \psi \\
 \dot{H} &= u_e \sin \theta - (v_e \sin \varphi + w_e \cos \varphi) \cos \theta
 \end{aligned} \tag{2.8}$$

with the new state variables $\psi, \theta, \varphi, x_e, y_e, H$. These twelve state variables are combined in the state vector \mathbf{x} .

$$\mathbf{x} = [V \ \alpha \ \beta \ p \ q \ r \ \psi \ \theta \ \varphi \ x_e \ y_e \ H]^T \tag{2.9}$$

And the resulting equations are combined in a single vector equation:

$$\dot{\mathbf{x}} = f'(\mathbf{x}, \mathbf{F}_{tot}(t), \mathbf{M}_{tot}(t)) \quad (2.10)$$

symbol	definition
$ I $	$I_{xx}I_{yy}I_{zz} - 2J_{xy}J_{xz}J_{yz} - I_{xx}J_{yz}^2 - I_{yy}J_{xz}^2 - I_{zz}J_{xy}^2$
I_1	$I_{yy}I_{zz} - J_{yz}^2$
I_2	$J_{xy}I_{zz} + J_{yz}J_{xz}$
I_3	$J_{xy}J_{yz} + I_{yy}J_{xz}$
I_4	$I_{xx}I_{zz} - J_{xz}^2$
I_5	$I_{xx}J_{yz} + J_{xy}J_{xz}$
I_6	$I_{xx}I_{yy} - J_{xy}^2$
P_l	$I_1 / I $
P_m	$I_2 / I $
P_n	$I_3 / I $
P_{pp}	$-(J_{xz}I_2 - J_{xy}I_3) / I $
P_{pq}	$(J_{xz}I_1 - J_{yz}I_2 - (I_{yy} - I_{xx})I_3) / I $
P_{pr}	$-(J_{xy}I_1 + (I_{xx} - I_{zz})I_2 - J_{yz}I_3) / I $
P_{qq}	$(J_{yz}I_1 - J_{xy}I_3) / I $
P_{qr}	$-((I_{zz} - I_{yy})I_1 - J_{xy}I_2 + J_{xz}I_3) / I $
P_{rr}	$-(J_{yz}I_1 - J_{xz}I_2) / I $
Q_l	$I_2 / I $
Q_m	$I_4 / I $
Q_n	$I_5 / I $
Q_{pp}	$-(J_{xz}I_4 - J_{xy}I_5) / I $
Q_{pq}	$(J_{xz}I_2 - J_{yz}I_4 - (I_{yy} - I_{xx})I_5) / I $
Q_{pr}	$-(J_{xy}I_2 + (I_{xx} - I_{zz})I_4 - J_{yz}I_5) / I $
Q_{qq}	$(J_{yz}I_2 - J_{xy}I_5) / I $
Q_{qr}	$-((I_{zz} - I_{yy})I_2 - J_{xy}I_4 + J_{xz}I_5) / I $
Q_{rr}	$-(J_{yz}I_2 - J_{xz}I_4) / I $
R_l	$I_3 / I $
R_m	$I_5 / I $
R_n	$I_6 / I $
R_{pp}	$-(J_{xz}I_5 - J_{xy}I_6) / I $
R_{pq}	$(J_{xz}I_3 - J_{yz}I_5 - (I_{yy} - I_{xx})I_6) / I $
R_{pr}	$-(J_{xy}I_3 + (I_{xx} - I_{zz})I_5 - J_{yz}I_6) / I $
R_{qq}	$(J_{yz}I_3 - J_{xy}I_6) / I $
R_{qr}	$-((I_{zz} - I_{yy})I_3 - J_{xy}I_5 + J_{xz}I_6) / I $
R_{rr}	$-(J_{yz}I_3 - J_{xz}I_5) / I $

Table 2.1 Definition of Inertia Coefficients

Equation (2.10) is a set of nonlinear equations that seem easy to manipulate, however this is not true. The total forces and moments, apart from being time dependent, are also state dependent leading to a strong coupling between the equations. What's more, in some cases this forces and moments depend also on the time-derivative of the state vector ($\dot{\mathbf{x}}$), which causes the vector equation to become implicit. Of course this depends on the way forces and moments can be described, something that is aircraft depended.

2.5 Model of the Aircraft total forces and moments

2.5.1 Model of the Aerodynamics forces and moments

The forces and moments acting on a complete aircraft are defined in terms of dimensionless aerodynamic coefficients [12], [14]. Usually these forces and moments are defined in terms of wind-axes components. This way we have the following coefficients:

$$\begin{aligned}
 \text{drag}, D &= \bar{q} S C_D \\
 \text{lift}, L &= \bar{q} S C_L \\
 \text{crosswind _ force}, C &= \bar{q} S C_C \\
 \text{rolling _ moment}, l_w &= \bar{q} S b C_l \\
 \text{pitching _ moment}, m_w &= \bar{q} S \bar{c} C_m \\
 \text{yawing _ moment}, l_w &= \bar{q} S b C_n
 \end{aligned} \tag{2.11}$$

The aerodynamic coefficients that are included in the above equations, are in practice specified as functions of the aerodynamic angles (α, β), Mach and altitude. In addition, control surface deflections δ_s and propulsion system effects cause changes in the coefficients. Consequently we write the dependence of an aerodynamic coefficient as:

$$C(\) = C(\)(\alpha, \beta, M, h, \delta_s, T_c) \tag{2.12}$$

Equation (2.12) implies a complicated functional dependence that has to be modeled as a “look-up table” in a computer. The vast majority of aircraft however have flight envelopes restricted to small angles of attack and/or low Mach numbers. For these aircraft, the functional dependence will be simpler and any given

coefficient might be broken down into a sum of simpler terms, with linearity assumed in some terms. This procedure is known as “component build-up”. Usually every aerodynamic coefficient consists of a “baseline” component plus small increments for control surfaces, gear, etc. These small increments are called aerodynamic derivatives.

The aerodynamic coefficients can be determined by the use of CFD computer codes (e.g Advanced Aircraft Analysis version 2.5) or a combination of empirical data and theory built into a computer program such as the Stability and Control Datcom. The input data to these programs include a geometrical description of the aircraft.

The aerodynamic coefficients can be estimated in a wind tunnel using an aircraft scale-model. This is the most widely used method to experimentally derive the coefficients. The second important method is the measurement of these coefficients through flight tests.

The UAV model [10] is a relatively simple model where every coefficient is composed of a baseline component and additional terms linear to state variables like α , β and their powers (up to the second). Analytical presentation of the modeling of the aerodynamic forces and moments can be found in Appendix A.

2.5.2 Model of the thrust (engine) forces and moments

The engine force (thrust) for every aircraft is modeled in a different way. For the simple UAV model used in this thesis it is assumed that the engine can produce a force only along the aircraft body’s longitudinal axes. No moments are thus created by the engine force. The force generated by the engine is dependent on the propeller slipstream and the angle of attack, while the coefficients defining the force are constant over the entire flight envelope.

The thrust force is computed as follows:

$$F_T = \rho n^2 D^4 C_{F_T}(J)$$

Where ρ is the air density, n the engine speed, D the propeller diameter and $C_{F_T}(J)$ is the dimensionless thrust coefficient that is expressed as a function of the ratio $J = \frac{V_T}{D\pi n}$.

The thrust coefficient is expressed as:

$$C_{F_T}(J) = C_{F_{T1}} + C_{F_{T2}}J + C_{F_{T3}}J^2$$

2.6 Control Vector and Actuators

As can be seen from the above modeling of forces and moments, these are influenced by the control inputs to the system. Control inputs are the surface deflections δ_s and the throttle setting (or thrust) for the engine. On a typical aircraft, there are five control surfaces (five actuators), two elevators, two ailerons and a rudder (figure 2.4). The elevators and ailerons are moving together (symmetrically and asymmetrically) making control law calculation easier.

So $u = [\delta a \ \delta e \ \delta r \ F_T]$, where, $\delta a, \delta e, \delta r$ is the aileron, elevator and rudder deflection scaled such that their range is the same ($\delta_e, \delta_a, \delta_r \in [-1, 1]$) and $F_T \in [0, 100]$ is the engine thrust.

However as will be stated in the next chapter, simulating individual surface failures requires the effect of each surface to be taken into account. Also a very easy way to increase redundancy in the aircraft actuation system without adding hardware is the individual movement of control surfaces. In this way only the controller complexity is increasing.

In order to take into account the individual contribution of every surface, we need to evaluate the effect of symmetrical and asymmetrical movement of ailerons and elevators. This modeling can be incorporated into the way aerodynamic forces are calculated. This means that equation (2-12) will be changed so that every actuator can contribute individually to the forces/moments generated. The control vector will be:

$$u = [\delta a1 \ \delta a2 \ \delta e1 \ \delta e2 \ \delta r \ F_T] \quad (2.13)$$

The control inputs are not directly set by the controller. Actuation devices are used to apply the control surface deflections. The dynamic properties of these actuators have to be taken into account during simulation and modeling. The dynamics of the actuators present nonlinear behaviors and dead zones, however for controller design it is common to approximate them [12] as:

$$\begin{aligned} G_{rudder}(s) &= \frac{20.2}{s + 20.2} \\ G_{aileron}(s) &= \frac{20.2}{s + 20.2} \\ G_{elevator}(s) &= \frac{10}{s + 10} \end{aligned} \quad (2.14)$$

As can be noticed, actuator dynamics are considerably fast (poles at -20.2 and -10) so the transfer functions of the actuators can often be approximated to 1 without losing much accuracy (of course this depends also on the system poles as well).

2.7 Sensors

In order to measure the state of the aircraft, several sensors are used. Body-axes linear accelerations can be measured with accelerometers. These measurements are considered highly reliable because of their excellent linearity and small bias error. However there are serious issues involving sensor location since measurements must be corrected to the aircraft center of gravity and they can pick up structural response and engine vibrations. The integration of these quantities can lead to the position estimation (x_e , y_e , H) although typically additional sensors are used (barometric altitude, GPS) and sensor fusion algorithms (usually Extended Kalman Filters although other more complicated algorithms like Particle Filters can be used). True airspeed (V) can be measured by air data sensors (pitot tubes) and angle of attack (α) can be measured by angle of attack sensors (flow vanes mounted on the aircraft). All these sensors are available in low weight/low cost due to MEMS technology. Although they suffer from low (relatively) accuracy and higher noise than sensors used in full scale aircraft, their accuracy is adequate. Furthermore it has been shown that their accuracy can be enhanced by the use of fusion algorithms and linear acceleration measurements. The efficiency of these methods and the ability to be run even in very small RC models has been proved [9],[15].

The only state that cannot be directly measured is sideslip angle β . The quantity that can indeed be measured with sensors similar to those of angle of attack, is the flank angle β_f which is the angle between the x body axis and the projection of true airspeed on the Oxy plane. It is easy to prove that $\beta = \tan^{-1}(\tan \beta_f \cos \alpha)$. In the case of small angles β_f , α , (a realistic assumption for a UAV which is usually not being flown aggressively) the above equation reduces to $\beta \simeq \beta_f$.

Aircraft angular velocity components (p, q, r) are usually measured using rate gyros attached to the aircraft and aligned with the body axis. These sensors are among the most reliable and accurate of the aircraft instrumentation. Angular accelerations

$(\dot{p}, \dot{q}, \dot{r})$ on the other hand, can be measured but the sensors available have relatively high noise level and/or lags. That's why they are not usually included in aircraft instrumentation systems. However as sensor technology evolve, this situation may change. Finally, integrating gyros or magnetometers can be used to calculate the Euler angles (φ, θ, ψ) .

As stated in [16], the dynamic characteristics of the sensors are provided by the manufacturer or can be estimated from dynamic tests in the laboratory. However, because the natural frequencies of sensors in aircraft and UAV instrumentation systems are usually very high relative to the frequencies associated with the quantities being measured, the sensor dynamics can be approximated by a small time delay or simply neglected. In this case, the sensors transfer functions can be assumed to be equal to 1. This approach was used in the current thesis.

The sensors are usually influenced by noise. This noise is assumed to be Gaussian zero mean white noise corresponding to typical specifications of low cost sensors. For the turn rate sensors the standard deviation is assumed equal to $\sigma_{p,q,r} = 5 \text{ deg/sec} = 0.0873 \text{ rad/sec}$, which corresponds to a noise covariance of $\Sigma_{p,q,r} = 0.0076 \times I_3 [\text{rad}^2 / \text{sec}^2]$. The airspeed sensor noise has a standard deviation of $\sigma_{v_t} = 1 \text{ m/s}$ ($\Sigma_{v_t} = 1 \text{ m}^2 / \text{sec}^2$). For the airflow sensors, the noise standard deviation is $\sigma_{a,b} = 2 \text{ deg} = 0.0349 \text{ rad}$ ($\Sigma_{a,b} = 0.0012 \times I_2 [\text{rad}^2]$).

Chapter 3 - Modeling of Faults on an Aircraft

3.1 Fault-Failure

According to the definitions in chapter 1, a fault corresponds to an abnormal behavior of the system which may not affect the overall functioning of the system but may eventually lead to a failure. For example, consider the temperature of an engine. If this temperature exceeds a certain accepted limit, say $100\text{ }^{\circ}\text{C}$, there is a fault in the system. Although this excessive temperature does not prevent the engine from working properly for a while, it may eventually damage components of the engine and possibly lead to its breaking down. In this thesis, the term fault is used to describe any abnormal behavior.

Faults/failures are events that take place in different parts of the controlled system. In the Fault Tolerant Control Systems (FTCS) literature faults are classified according to their location of occurrence as (figure 3.1): Actuator, sensor and component faults. Further, with respect to the way they are modeled, they are classified as additive and multiplicative. Additive faults are suitable to represent component faults in the system while sensor and actuator faults are in practice most often multiplicative in nature. Faults are also classified according to their time characteristics as abrupt, incipient or intermittent. Abrupt faults occur instantaneously often as a result of a hardware damage. Usually they are very severe as they affect the performance and/or the stability of the controlled system and as such they require prompt reaction by the FTCS. Incipient faults represent slow in time parametric failures, often as a result of aging. They are more difficult to detect due to their slow characteristics but are also less severe. Finally intermittent faults appear and disappear repeatedly, for instance due to partially damaged wiring.

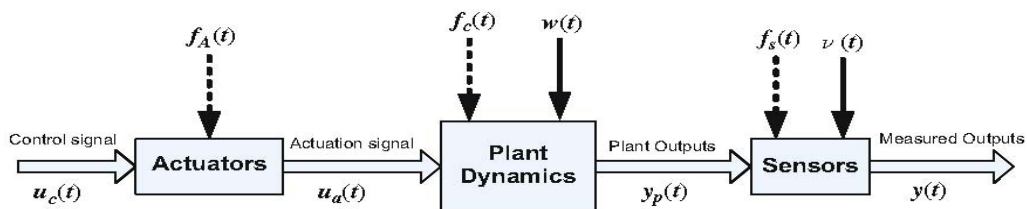


Figure 3.1 Classification of faults according to location

3.2 Faults and Failures on an Aircraft System

3.2.1 Sensor Faults/Failures

Sensor faults are less critical than actuator faults because even though they can seriously affect control system performance they do not change the dynamics of the system. They do not influence the controllability of the system although of course they affect observability.

Typical sensor faults in aircraft systems are described in figure 3.2. The faults can be classified as:

Bias is a constant offset/error between the actual and measured signals.

Sensor drift is a condition whereby the measurement errors increase over time (and might be due to loss of sensitivity of the sensor).

Loss of accuracy occurs when the measurements never reflect the true values of the states. In that case, the standard deviation of the measurements is increased. This fault can be modeled as an increase of noise added to the measurements.

Freezing of sensor signals indicate that a sensor provides a constant value instead of the true value.

Finally calibration error is a wrong representation of the actual physical meaning of the states from the electrical or electronic signals that come out from the sensor unit itself.

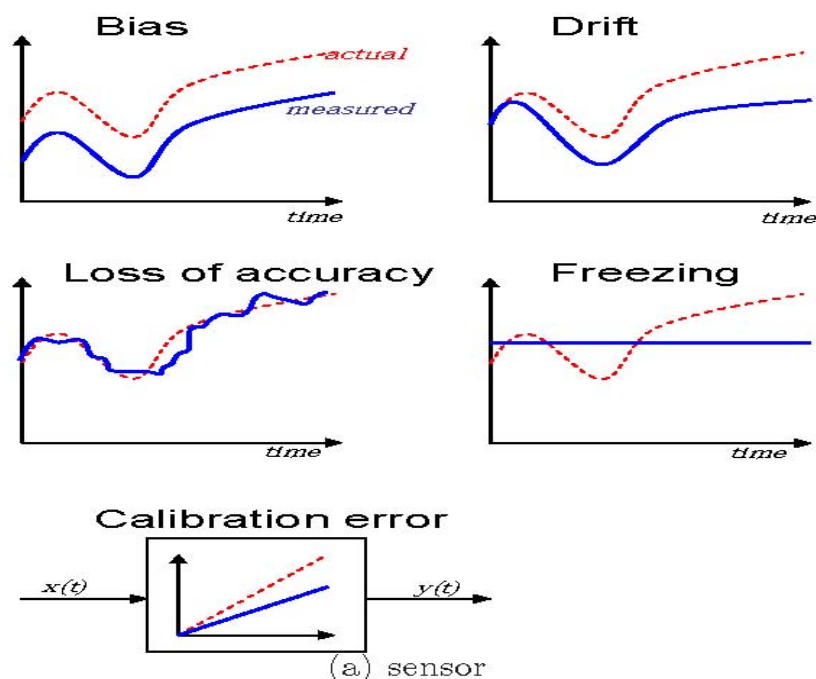


Figure 3.2 Types of Sensors faults/failures

The mathematical representation of the above sensor faults is as follows [17]:

$$y_i(t) = \begin{cases} x_i(t) & \text{Absence_of_faults} \\ x_i(t) + b_i & \dot{b}_i(t) = 0, b_i(t_{Fi}) \neq 0 (\text{Bias}) \\ x_i(t) + b_i(t) & |b_i(t_{Fi})| = c_i t, \forall t \geq t_{Fi} (\text{Drift}) \\ x_i(t) + b_i(t) & |b_i(t_{Fi})| \leq \bar{b}_i, \dot{b}_i(t) \in L^\infty, \forall t \geq t_{Fi} (\text{Loss_of_accuracy}) \\ x_i(t_{Fi}) & \forall t \geq t_{Fi} (\text{Freezing}) \\ k_i(t)x_i & \forall t \geq t_{Fi} (\text{CalibrationError}) \end{cases}$$

Where t_{Fi} represents the time of failure, $k_i(t)$ is the effectiveness and b_i denotes the accuracy coefficient of the sensor.

Sensor faults/failures can occur due to malfunctions in the components in the sensor unit, loose mounting of the sensors and loss of accuracy due to wear and tear. They are milder than actuator faults and in a manned aircraft the disengagement of autopilot systems is sufficient to overcome their effects. However, an unmanned system is much more vulnerable to such failures due to its dependence on autopilot systems.

3.2.2 Actuator Faults/Failures

In aircraft systems there are a few distinct types of actuator failures, the three most common are shown in figure 3.3 [17].

Lock in place Failure (LIP) is a failure condition when an actuator becomes stuck and immovable. This might be caused by a mechanical jam, due to lack of lubrication for example. This type of failure occurs in documented incidents like flight 96 (DC-10, Windsor, Ontario, 1972) [6] (where the rudder jammed with an offset). In these cases, no aerodynamic change happens to the aircraft, although the asymmetrical movement of the control surfaces induces unwanted forces and moments that need to be compensated for.

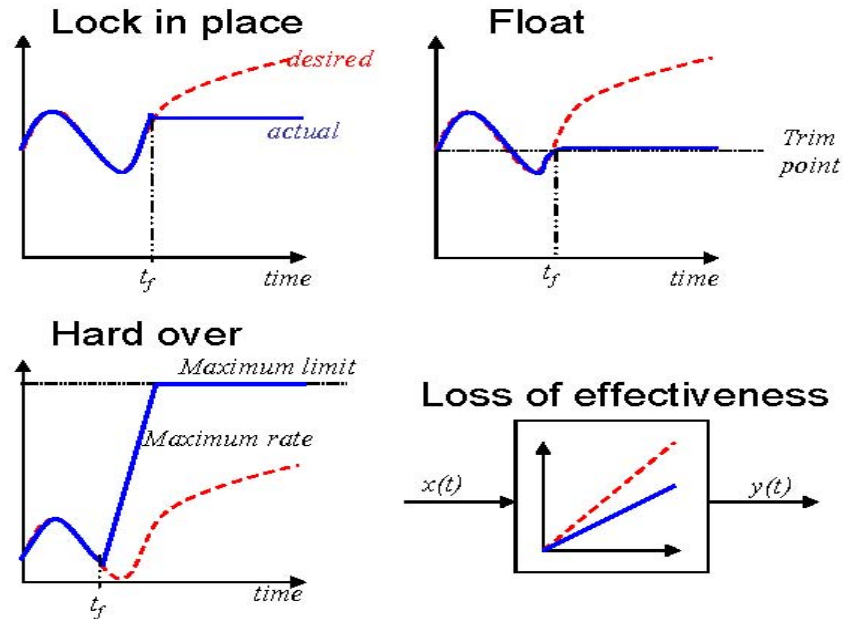


Figure 3.3 Types of Actuator faults/failures

Float failure is a failure condition when the control surface moves freely without providing any moment to the aircraft. An example of a float failure is the loss of hydraulic fluid in the elevator's actuator causing it to move freely in the direction of angle of attack and therefore cannot produce any effective moment in the pitch axis. This type of failure occurred in flight DHL A300B4 (A300, Baghdad, Irak, 2003) [6] (where a total loss of hydraulics occurred).

Runaway/hardover (HOF) is the most catastrophic type of failure. A runaway control surface will move to its maximum rate limit until it reaches its maximum position limit or its blowdown limit. For example, a rudder runaway can occur when there is an electronic component failure which causes a (wrong) large signal to be sent to the actuators causing the rudder to be deflected at its maximum rate to its maximum deflection at low speed (or its blowdown limit at high speeds). This type of failure occurred in flight 85 (B-747, Anchorage, Alaska, 2002).

Loss of effectiveness (LOE) is the reduction of the actuator's efficiency (gain) by some factor. This loss of effectiveness is the most widely studied case in the literature. It can be assumed to occur if a part of the control surface is deformed or broken, in such a way that only negligible change occurs in the aerodynamic characteristics of the aircraft. For example a small loss (10%) in the elevator actuator's surface will not move the center of gravity of the aircraft or modify its aerodynamic coefficients other than the coefficients that are directly related to the control surface.

Different types of actuator failures can be mathematically represented by:

$$u_a^i(t) = \begin{cases} u_c^i(t) & \text{Absence_of_faults} \\ k_i(t)u_c^i(t) & 0 < \varepsilon_i \leq k_i(t) < 1, \forall t \geq t_{Fi} (LOE) \\ 0 & \forall t \geq t_{Fi} (Float) \\ u_c^i(t_{Fi}) & \forall t \geq t_{Fi} (LIP) \\ u_{i\min} \vee u_{i\max} & \forall t \geq t_{Fi} (HOF) \end{cases} \quad (2.15)$$

Where $u_a^i(t)$ denotes the actuation signal (or actuator output) from the i th actuator, $u_c^i(t)$ is the control command signal (or actuator input) to the i th actuator, t_{Fi} denotes the time of fault occurrence on the i th actuator and $k_i(t)$ is the actuator effectiveness coefficient of the i th actuator. $u_{i\min}$ and $u_{i\max}$ are the lower and upper limits on the actuation level. We can represent the above cases with the following mathematical model:

$$u_a^i(t) = \delta_i k_i u_c^i(t) + (1 - \delta_i) \bar{u}_i \quad (2.16)$$

Where $\delta_i=1, k_i=1$ in the absence of failures, $\delta_i=1, k_i=0$ in the presence of LOE and $\delta_i=0$ in other types of faults with \bar{u}_i being the position at which the actuator locked.

Equations (2.15) and (2.16) were used to inject failures in the Simulink model.

Actuator faults are really critical for the operation of any controlled system. Even though they don't affect system dynamics of the controlled system itself, they can significantly affect the dynamics of the closed loop system, and may even affect the controllability of the system. Figure 3.4 presents a simple example with a partial 50% actuator fault that results in instability of the closed-loop system. In this example a system with transfer function $S(s)=1/(s-1)$ is controlled by a PI controller with transfer function $C(s)=1.5+5/s$, so that a sinusoidal reference signal is tracked under normal operating conditions. At time $t=20$ sec, a 50% loss of control effectiveness is introduced and as a result the closed-loop system stability is lost. This example makes clear that even «seemingly simple» faults can significantly degrade the performance and can even destabilize the system.

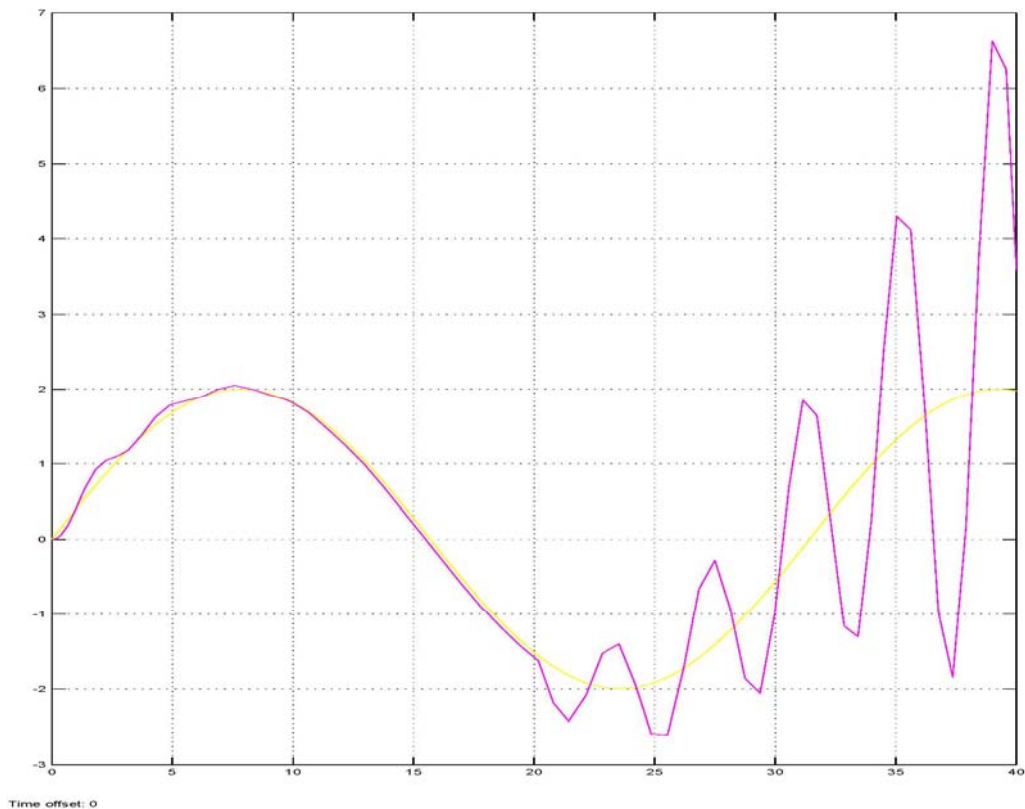
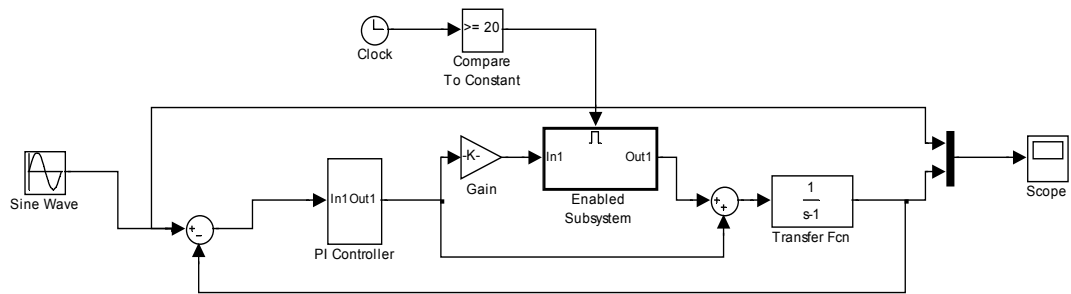


Figure 3.4 After an Actuator fault the system may become unstable if no reconfiguration takes place.

3.2.3 Structural Faults/Failures

Another category of faults present in aircraft systems is the component or structural faults. These are caused by a structural damage of the wings, airframe or control surfaces. Structural damage may change the operating conditions of the aircraft (from its nominal conditions) due to changes in the aerodynamic coefficients or a change in the center of gravity. These types of

failures have been modeled in the FTCS literature in terms of linear systems as changes to the system and control matrices A , B . Mathematically, this can be represented as:

$$\dot{x}(t) = (A + \Delta A)x(t) + (B + \Delta B)u(t) + \xi(x, u, t)$$

It is really difficult to model the aerodynamic changes of an aircraft under structural damage in the general case. An attempt to model these changes in the case of control surface damage was made in [18]. It was assumed that the physical destruction of the control surface imposed a quantitative alteration in the aerodynamic modeling. That means that the forces and moments generated by the control surface after the failure differ from those before the failure by a proportional factor (\bar{s}_d) affecting the efficiency parameter.

Another way of modeling aerodynamic changes was made by Smaili et al. [19] in an attempt to reproduce through simulation the failure scenario of flight's 1862 accident. The reconstructed aerodynamic effects were added as contributions to the baseline aerodynamic coefficient equations of the validated undamaged aircraft model. The reconstruction methodology allowed an iterative adjustment of the initial aerodynamic estimates, in an a priori model structure, that accounts for the overall effect of aircraft structural damage to obtain a match with the collected data from the Flight Data Recorder.

A more recent approach was proposed in [20], where the general equations of motion of a structural damaged aircraft were derived. The model would be validated through flight tests. It should be emphasized at this point that both before and after the failure the movement of the aircraft can be described by the rigid body model of section 2.4 after the transient effects (due to mass change etc) die out. The difference is that the two models will have different masses, moments of inertia and a different center of gravity. Of course, the aerodynamic forces and moments acting on the two models will be different also. To derive a general model from which each of the two models can be derived is an extremely difficult task.

3.3 Focus of the thesis

In this thesis we will focus on actuator rather than sensor failures since these are considered much more critical to handle. Sensors are much cheaper and lighter than actuators and the possibility of including hardware redundancy (i.e multiple IMUs or visual sensors and integrate them through a data fusion algorithm) is greater. The failures cover every possible actuator failure scenario.

Control surfaces are treated separately. This means that both stuck actuator and loss of control effectiveness failure will be assumed to affect a single control surface. This scenario is much more realistic than those encountered in the literature where both surfaces are assumed stuck or having lost their effectiveness. The loss of a part of the left aileron for example will definitely lead to reduced rolling moment capability for the same differential deflection of the two ailerons however it will also induce some pitching moment although the ailerons will deflect symmetrically. In order to do that separate modelling of each control surface effectiveness should be available.

Chapter 4 - Review on Fault Tolerant Control (FTC) and Fault Detection and Identification (FDI)

As was stated earlier, the purpose of FTC is to maintain the system under control even in the presence of faults. FTC is generally divided into two classes: passive and active [21] (see figure 4.1).

Passive FTCS are based on robust controller design techniques and aim at synthesizing one (robust) controller that makes the closed-loop system insensitive to certain faults. This approach requires no online detection of the faults, and is therefore computationally more attractive. However its applicability is restricted because in order to achieve such robustness a very restricted subset of faults can be considered while the increased robustness is only possible at the expense of decreased nominal performance.

The active approach to the design of FTCS is based on controller redesign or selection/mixing of pre-designed controllers. This technique requires a Fault Detection and Isolation scheme that has the task to detect localize and estimate the magnitude of the faults that occur in the system. The structure of an active FDI-based FTCS is presented on figure 4.2 [22].

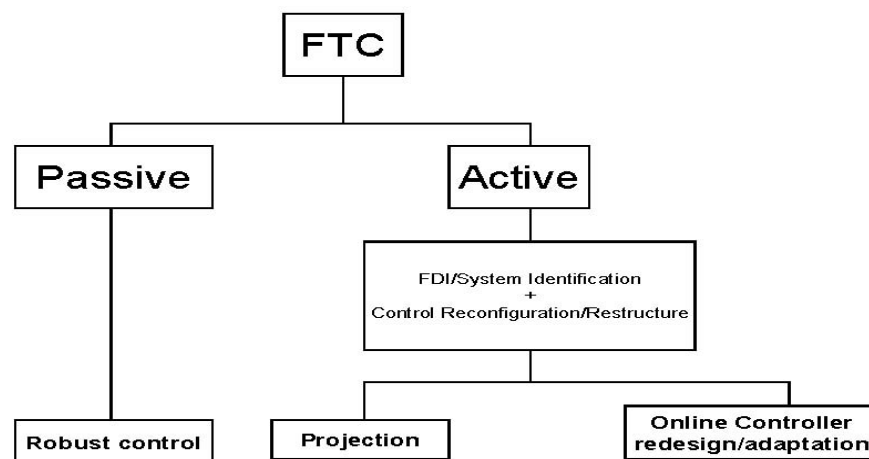


Figure 4.1 Classification of FTC [21]

Depending on the way the post-fault controller is formed, the active FTC methods are further subdivided into projection based methods and on-line redesign methods. The projection based methods rely on a controller selection from a set of off-line pre-designed controllers. Usually each controller of the set is designed for a particular fault situation and they are switched on according

to the fault diagnosed by the FDI/FDD module. The on-line redesign methods involve on-line computation of the controller parameters, referred to as reconfigurable control, or recalculation of both the structure and the parameters of the controller, called restructurable control. The on-line redesign methods are superior, with respect to post-failure performance, to the passive methods and the off-line projection based methods. However, they are the computationally most expensive methods as they often boil down to on-line optimization.

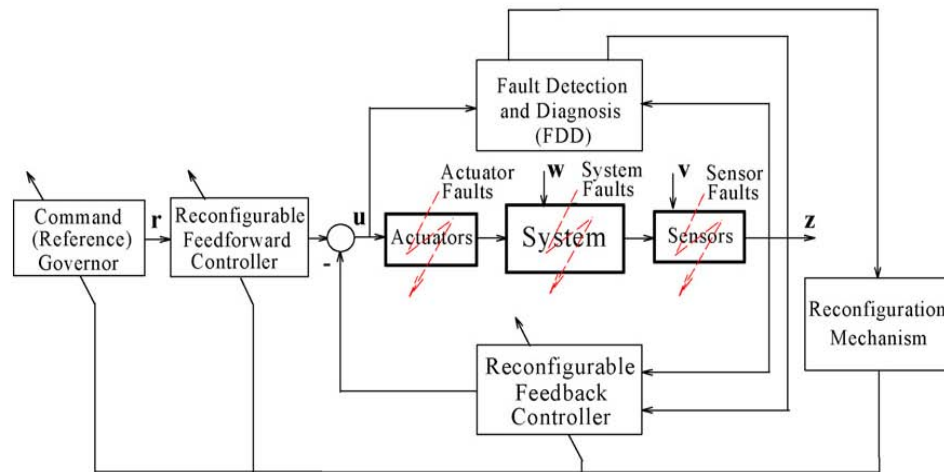


Figure 4.2 A general structure of Active Fault Tolerant Controller (AFTC) [22]

It is evident from the above figure that there is a close relationship between FTC and FDI (sometimes referred to as Fault Detection and Diagnosis (FDD)). At the same time there is a close relationship between FTC, FDI and robust control as was identified in [21]. FTC can be regarded as a complex combination of these three major research fields (see figure 4.3).

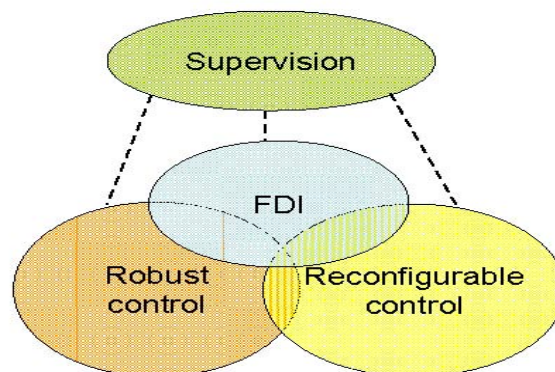


Figure 4.3 Areas of fault tolerant control research [21]

4.1 Fault Detection and Identification

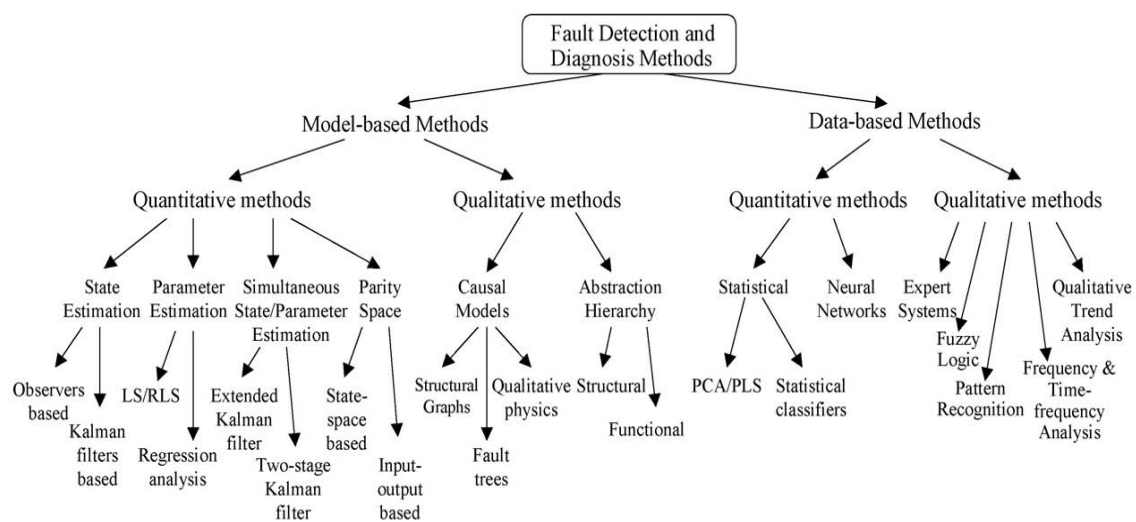
As was stated earlier, in active FTC, FDI plays a vital role to provide information on faults/failures in the system and to enable reconfiguration to take place. Therefore the main function of FDI is to detect a fault or failure and to find its location and possibly its magnitude so that corrective action can be made to eliminate or minimize the effect on the overall system performance.

The FDI process is composed of three steps according to their «depth» [1]:

- fault detection: determination of the existence of faults in the system
- fault isolation: determination of the kind and location of the fault
- fault identification: determination of the size and time-variant behavior of the fault.

These steps are functions of deferent complexity (and difficulty) which increases with the number of the step. Thus detecting that a fault is present is the easier task while isolating the fault and estimating its magnitude is a rather difficult task. This explains the limited literature that involves the topics of fault isolation and identification with respect to that of fault detection.

The main classification of FDI techniques in the literature is that between model based and Data based methods [23]. An extensive review of the different methods can be found in the same source and can be seen in figure 4.4.



Note: LS/RLS: Least Squares/Recursive Least Squares; PCA: Principal Component Analysis; PLS: Partial Least Squares.

Figure 4.4 Classification of FDI methods [23]

Model free (or data based) approaches to FDI are applicable when no explicit dynamical model of the system is available. The system knowledge boils down to real-time measurements, possibly completed by process history. With such data two main strategies could be adopted [24]. In a sense both aim at interpolating the new measured point based on the available data. The first strategy is classification. It involves building classes from the database either in a supervised way (i.e. with the help of an expert) or in an unsupervised manner (i.e., collecting elements of the database that are close to one another (clustering)). A classifier is then trained with respect to these classes to perform the classification of the newly measured variables as representative of a healthy or faulty behavior. The second strategy is model building. It builds a statistical model that uses the redundancy of the process history in order to predict the values of the new variables and generate residuals by comparing predictions to measured values. A lot of methods have been proposed in the literature like neural networks, trend analysis, kernel machines and Support Vector Machines, Principal Component analysis (PCA) etc. They can overcome the lack of an analytical model which is a major problem to complex industrial systems as well as non-linearity in the system to be monitored but their main disadvantage is that in order to detect and isolate faults the process history has to include the faults that we seek to isolate.

In the case of a fixed wing UAV an analytical model of the system is available. The dynamics are quite well studied due to their resemblance to aircrafts. That means that a model-based method can be adopted limiting the need to collect extensive process history data (which are difficult or even impossible to collect for every faulty situation).

4.1.1 Model Based FDI

When the physics of the process is well known, it becomes possible to use an explicit knowledge-based dynamical model. Fault detection then amounts to checking whether the behavior of the monitored system is inconsistent with that of the model. This is done by exploiting the structure of the model and the existing input-output relationships (analytical redundancy). With respect to the model used, model-based methods are further subdivided to qualitative and quantitative. Qualitative model based methods are applied whenever the model of the system is available but the confidence in its parameters and quantitative outputs are very low.

The most widely used methods for FDI in aircraft systems are quantitative model based methods. In this case, a reliable (and possibly reduced) model of the system is used to reconstruct the process/system's behavior online, which associated with the concept of hardware redundancy is called software redundancy or analytical redundancy concept. Similar to the hardware redundancy schemes, in the framework of analytical redundancy the system model will run in parallel to the system and be driven by the same inputs. It is reasonable to expect that the reconstructed system's variables delivered by the model will well follow the corresponding real system variables in the fault free operating states and show an evident derivation by a fault in the system. In order to receive this information, a comparison of the measured system variables (output signals) with their estimates delivered by the model will then be made. The difference between the measured and estimated variables is called residual.

The process of a model based fault diagnosis system can be separated into two stages. The first stage involves the process of creating the estimates of the system outputs and building the difference between the real outputs and their estimates and is called residual generation. The second part involves the post-filtering of the residuals to extract the information about the presence of the faults, their location and possibly their magnitude and is called residual evaluation. The complete process of model based diagnosis is shown in figure 4.5.

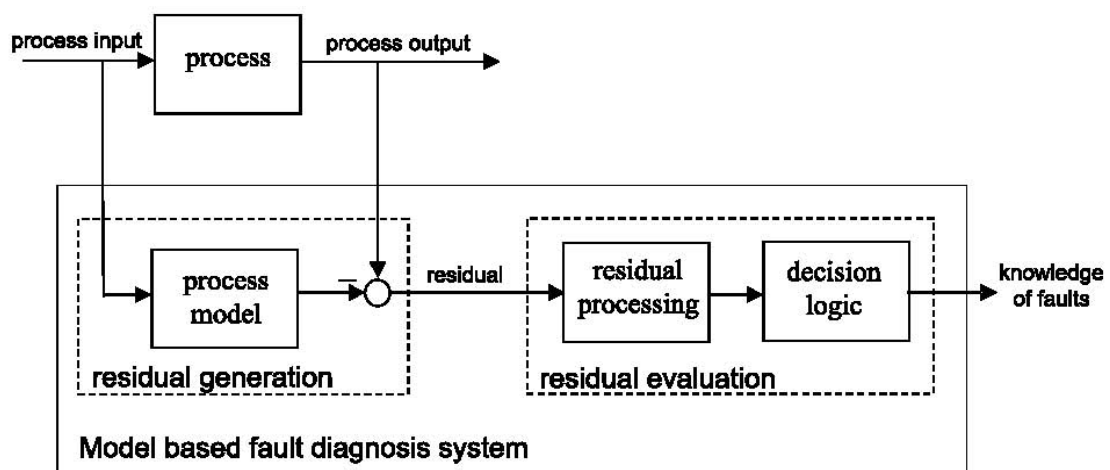


Figure 4.5 Schematic description of model based fault diagnosis scheme [25]

Quantitative model based methods can be further divided to observer-based, parity space and parameter identification based methods.

4.1.1.1 Parameter Estimation based methods

These techniques are suitable when the faults considered have a direct effect on some characteristic constant of the system itself. The nominal value of the parameter vector is supposed known and FDI boils down to estimating on-line the value of the parameters to generate residuals. The on-line estimation is a demanding task and usually models that are linear to their parameters are considered. In the case of aircraft fault detection the parameters identified are usually the aerodynamic coefficients which change based on the operating conditions. These coefficients are usually pre-estimated offline through wind tunnel and flight tests before being used for controller design. However during faults/failures (especially structural damage) no accurate pre-estimate is available and therefore these coefficients need to be obtained online. Recursive least squares (RLS) and Modified Sequential Least Squares (MSLS) algorithms have been used and successfully flight tested [8], [26], [27]. Other researchers proposed the use of a frequency domain method based on discrete time fast fourier transform [28]. Extended Kalman Filtering (EKF) is another option for on-line parameter estimation [29]. A comparison of different parameter estimation techniques within a fault tolerant control system was conducted in the framework of Intelligent Flight Control System (IFCS) F-15 program [30]. More recently a two step method for estimating the model of a damaged aircraft on-line for FDI and control reconfiguration was proposed by researches of TU Delf [31]. This method splits the identification procedure into two consecutive steps: a non-linear state estimation step where an EKF can be utilized and a linear, in the aerodynamic parameters, identification step which is solved by a recursive least squares algorithm. The main advantage of the above methods is the fact that the parameter estimates obtained can be directly used to update the control law via an adaptive control technique. However the task of identifying all the aerodynamic parameters of the aircraft can be very challenging due to cross coupling and insufficient excitation.

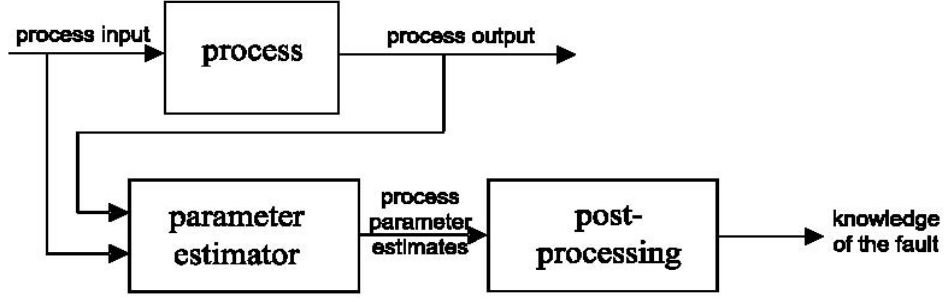


Figure 4.6 Schematic description of the parameter identification scheme [25]

4.1.1.2 Parity Equation based Methods

The basic idea of parity relations approach is to «provide a proper check of the parity (consistency) of the measurements of the monitored system» [32]. The following description adopted by [25] illustrates the basic concept. Consider a system with the state space description:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + E_d d(k) + E_f f(k) \\ y(k) &= Cx(k) + Du(k) + F_d d(k) + F_f f(k) \end{aligned} \quad (4-1, 4-2)$$

Combining together the above relations from the time instant $k-s$ up to k we end up to the following relation which is called parity relation and describes the input-output relationship in dependence of the past state variable $x(k-s)$, the disturbance d and the faults f :

$$\underbrace{\begin{bmatrix} y(k-s) \\ y(k-s+1) \\ \vdots \\ y(k) \end{bmatrix}}_{y_s(k)} = H_{o,s} x(k-s) + H_{u,s} \underbrace{\begin{bmatrix} x(k-s) \\ x(k-s+1) \\ \vdots \\ x(k) \end{bmatrix}}_{u_s(k)} + H_{f,s} \underbrace{\begin{bmatrix} f(k-s) \\ f(k-s+1) \\ \vdots \\ f(k) \end{bmatrix}}_{f_s(k)} + H_{d,s} \underbrace{\begin{bmatrix} d(k-s) \\ d(k-s+1) \\ \vdots \\ d(k) \end{bmatrix}}_{d_s(k)} \quad (4-3)$$

Where,

$$\begin{aligned} H_{o,s} &= \begin{bmatrix} C \\ CA \\ \vdots \\ CA^s \end{bmatrix}, \quad H_{u,s} = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}B & \dots & CB & D \end{bmatrix}, \\ H_{f,s} &= \begin{bmatrix} F_f & 0 & \dots & 0 \\ CE_f & F_f & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}E_f & \dots & CE_f & F_f \end{bmatrix} \quad \text{and} \quad H_{d,s} = \begin{bmatrix} F_d & 0 & \dots & 0 \\ CE_d & F_d & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s-1}E_d & \dots & CE_d & F_d \end{bmatrix} \end{aligned}$$

The parity relation based residual generator is constructed by finding the vector (or matrix) V such that:

$$VH_{s,o} = 0$$

and expressing the residual as:

$$r(k) = V(y_s(k) - H_{u,s}u_s(k))$$

The dynamics of the residual in the fault free case is then expressed as:

$$r(k) = V(y_s(k) - H_{u,s}u_s(k)) = VH_{s,o}x(k-s) = 0$$

In the presence of faults and disturbances the dynamics of the residual are:

$$r(k) = V(H_{f,s}f_s(k) + H_{d,s}d_s(k)) \neq 0$$

The parity space residual generation design procedure is simple and straight forward since it involves algebraic manipulation of the system and not the implementation of advanced control techniques. The method can be extended to provide isolation of both actuator and sensor faults. However as mentioned in [25] provides less design flexibility compared to that of observer based FDI. In fact it has been proved that parity space methods lead to certain types of observer structures and therefore are structurally equivalent to the observer based ones though the design procedures differ.

The most common approach of parity space approach methods in the aerospace field is based on the redundancy available in Inertial Measurement Units (IMUs) [33],[34]. Another approach which belongs to the parity space methods is the so-called Polynomial method (PM) ([35],[36]). It is strongly dependent on the use of an input-output polynomial description of the system under diagnosis. An important aspect of the PM residual generator design concerns the decoupling properties of the disturbance. This decoupling is obtained by means of suitable coordinate exchange of the monitored input-output system. In [36] the methods effectiveness is tested on a non-linear aircraft simulator model that takes into account wind gusts.

4.1.1.3 Observer based Methods

The basic idea underlying observer based (or filter based, in the stochastic case) approaches to fault detection is to obtain the estimates of certain measured or unmeasured signals. Then in the most usual case, the estimates of the measured signals are compared to their originals, i.e., the difference between the original signal and its estimate is used to form a residual signal $z(k) = y(k) - \hat{y}(k)$ (figure 4.7).

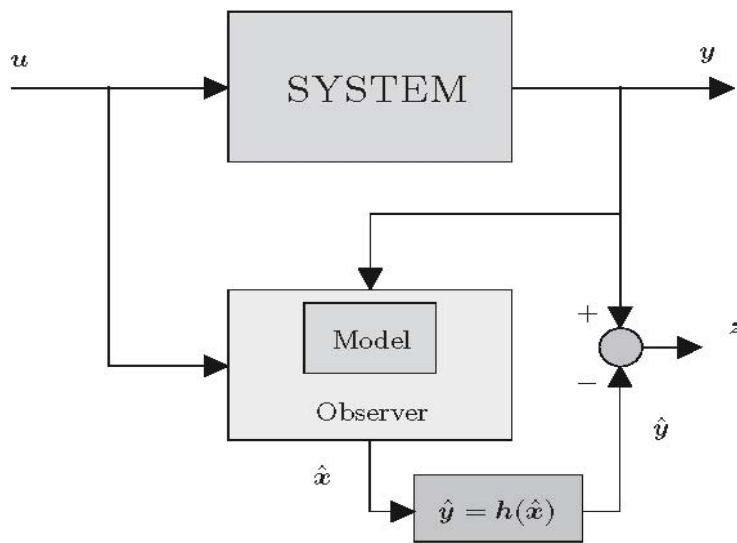


Figure 4.7 Principle of Observer-based residual generation [37]

To tackle this problem, many different observers (or filters) can be employed, e.g., Luenberger observers, Kalman filters, etc. From the above discussion it is clear that the main objective is the estimation of system outputs while the estimation of the full state vector is unnecessary. Since reduced order observers can be employed, state estimation is significantly facilitated. On the other hand, to provide an additional freedom to achieve the required diagnostic performance, the observer order is usually larger than the minimum one. The wide acceptance of observer based fault detection schemes is caused by the still increasing popularity of state-space models as well as the wide usage of observers in modern control theory and applications. Due to such conditions the theory of observers (or filters) seems to be well developed especially for linear systems. This has made a good background for the development of observer-based FDI schemes.

As was proved in [25], any residual generator can be considered as an extension of an output observer based residual generator. They consist of two parts: an output observer and a dynamic system. These two parts may take different functions:

- The output observer builds the core of the residual generator and is used to reconstruct the system behavior so that the original form of residual signal, $z(p) = y(p) - \hat{y}(p)$, provides us with the information about the variation of the system operation from its nominal value,
- The dynamic system $R(p)$, acts in fact as a signal filter and can, by a suitable selection, help us to obtain significant characteristics of faults thus is also called post-filter.

In fact all the observer based residual generation approaches aim in finding a suitable observer gain matrix L and post filter $R(p)$ applying different mathematical and control theoretical tools.

The first kind of observer based residual generators was the Fault Detection Filter (FDF) proposed by Beards and Jones in the early 70's. Another form is the diagnostic observer which has a structure similar to the Luenburger type observer and is designed under deterministic hypotheses. Luenburger observers have been applied to autonomous helicopters for fault detection [38].

Kalman filter

The Kalman filter is a widely used algorithm for state estimation based on indirect, inaccurate and uncertain observations [39]. It assumes linearity in the dynamics and Gaussian noise however its success in the industry is caused by the relative robustness it exhibits to the violations of the above hypotheses. Kalman filters have been proposed for aircraft systems FDI in different configurations as well as for other autonomous vehicles ([40]-[45]).

Disturbance Decoupling Approaches

In the disturbance decoupling approaches, the aim is to generate the fault indicating signals (residuals) so that they behave in an orthogonal space of unknown inputs (disturbances, modeling

errors etc.) whilst maintaining sensitivity to the faults. This scheme is often called unknown input observer (UIO) [32]. An UIO is a Luenburger type observer that delivers a state estimate independent of unknown input d . If $H_{f,s}$, $H_{o,s}$ and $H_{d,s}$ are the matrices defined in section 4.1.1.2 it can be shown [25] that the design of an UIO boils down to the choice of matrix V such that $VH_{f,s} \neq 0$ and $V[H_{o,s} \ H_{d,s}] = 0$.

Several signals can be treated as disturbances. For actuator fault isolation for example, we may consider as disturbances the other inputs to the system except the one being monitored. Also modeling errors can be treated as disturbances as long as their influence on the system is deterministic and known.

Some recent applications of UIOs were reported for the fault diagnosis of a linear model of an unmanned aerial vehicle (helicopter) [46] and the monitoring of gyroscopes in a spacecraft [47]. The concept of unknown input decoupling can be extended in the stochastic case when the system is influenced by zero mean, white noise sequences. The UIO derived in this case is called Unknown Input Filter (UIF) [37]. Such a filter was used for the IMU and thruster diagnosis of the Mars Express spacecraft [48].

Norm based approaches

The disturbance perfect decoupling is often not possible. As mentioned in [25], the restriction for the application of perfect decoupling may be too strong for the practical use of the technique. Indeed the existence condition for perfect decoupling is $rank(G_{yd}(p)) < m$ where m the number of sensors and $G_{yd}(p)$ the transfer function connecting the output and the disturbance (unknown input) implies that there are enough sensors available, something that may not be too realistic. Furthermore if disturbances appear in all directions of the measurement subspace, the decoupling approaches will fail.

The norm based approaches try to resolve this issue in the context of a trade-off between the robustness against the disturbances and the sensitivity to the faults. As a result the residual signal will also be affected by the disturbances. Generally there are three different strategies to attack this problem:

- Make use of knowledge for the disturbance (a typical example is the kalman filter which assumes that the unknown input is white noise)

- Approximate the disturbance to output transfer function with another which satisfies the condition for perfect decoupling. Then the design of an observer for this transfer function is possible
- Designing residual generators under a certain performance index. A lot of performance indexes have been proposed leading to a variety of designs.

Among the above strategies the third is the most widely accepted. One of the first contributions to this method was the one from Hou and Patton [50]. According to this method a structured residual vector r is structured in following general form:

$$r(k) = M_y y(s) + M_u u(s) - L(s) \begin{pmatrix} y(s) \\ u(s) \end{pmatrix}$$

$$u(s) = k(s)y(s)$$

The FDD problem consists then of jointly designing M_y , M_u and $L(s)$ such as the effects that faults have on r are maximized in the H_2 -norm sense, whilst minimizing the influence of unknown inputs and model uncertainties in the H_∞ -norm sense. Applications of the method in the aerospace field are numerous ([51]-[53]). Other performance indexes can be used, like the one proposed by Ding et. al. [25], [54], which tries to prevent conservatism in the design and achieve a design that guaranties a minimum False Alarms Rate (FAR), maximizing the Fault Detection Rate (FDR).

Fault estimation filters

The concept of observers or filters can be extended in the case of fault estimation. The purpose of the above mentioned scheme is not only to detect faults but also to estimate their magnitude. In the case of an actuator loss of effectiveness case for example, the identification of the post-failure effectiveness factor could be essential information for the reconfiguration strategy. The idea of fault estimation is depicted in figure 4.8 and is sometimes referred to as simultaneous state and parameter estimation.

The two step Kalman filter falls in this category [55],[56]. An Extended Kalman filter can also be used adding the fault magnitude as an extra state for estimation.

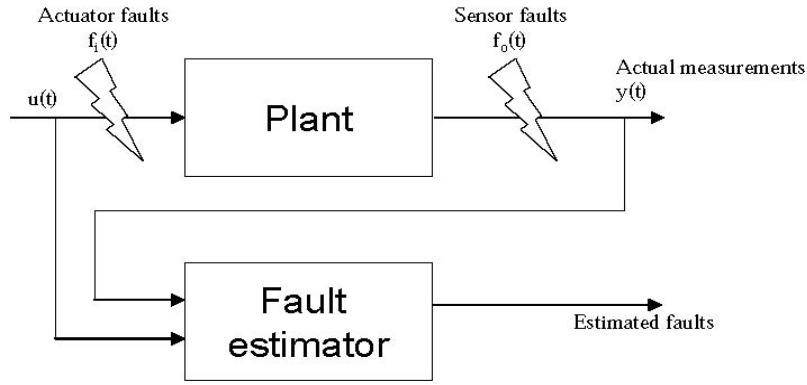


Figure 4.8 Fault estimation based FDI

In the context of robustness to disturbances and modeling errors, in the same manner as norm based observer design, the H_∞ framework can be used to design robust fault estimator filters [57]. As shown in figure 4.9 in the case of a system model following the Linear Fractional Representation (LFT) form, the H_∞ - based fault estimation problem is equivalent to the design problem of a stable filter F such that for all model perturbations $\Delta \in \|\Delta\|_\infty \leq 1$, \hat{f} is an optimal estimate in the H_∞ -norm sense, of the fault signal f . The method was applied for the fault diagnosis of the X-33 and Hopper RLVs [58] [59] as well as the fault detection and isolation of a transport aircraft (Boeing 747-100/200) [60].

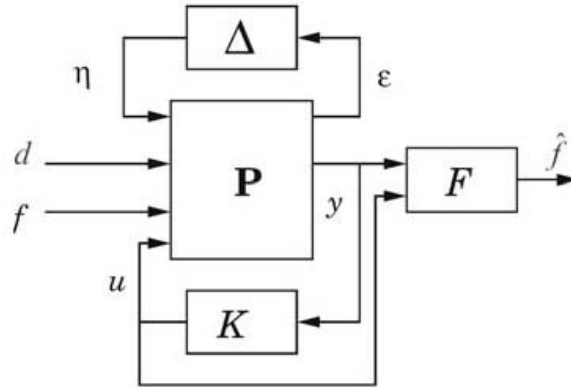


Figure 4.9 The H_∞ -based fault estimation problem [4]

Multiple Model and Interactive Multiple model based FDI

In multiple model based FDI, a bank of models with anticipated faults/failures is created. The outputs of this bank of models can be compared with the actual plant output to create residual error signals [44]. Here the model with the smallest error is the model which best represents the current faults/failures in the system. Therefore the faults can be detected and isolated. The scheme was first implemented using a bank of Kalman filters [40], [42], [44], [45]. A bank of Extended Kalman Filters was proposed in [10]. Other observer methods can be used like a bank of UIOs. The main advantage of multiple model methods is the very fast fault isolation as well as the accurate reconstruction of the state estimate as the probability weighted sum of estimates. These advantages make the method suitable to the FDI of dynamic systems like aircraft where fast fault isolation is essential for the prevention of instability. On the other hand the use of multiple models or observers increases the computational load. Moreover the expected faults should be able to be hypothesized by a reasonable number of filters. Structural failures are very difficult to be addressed due to their great variety.

4.1.1.4 Non linear Systems

The application of fault detection to practical systems is limited by the nonlinear nature of almost every dynamical system. A simple method to deal with this problem is the use of linearization of the system around an operating point. The use of any linear method is then straight forward. The design of several filters for different operating points can be easily accomplished and gain scheduling can be used to cover the full operating envelope of the system. Although this method works well for systems that operate close to the linearization points or systems that are piecewise linear, it cannot handle highly nonlinear plants. The performance of the FDI system can also be degraded by modeling errors and uncertainties induced by the linearization as well as by coupling of inputs not included in the linear model used for FDI design. Moreover, the operation of the system away from the linearization operating points can trigger false alarms. There are cases thus that necessitate the use of nonlinear techniques for FDI design. Most nonlinear techniques constitute direct extensions of the approaches described in sections 4.1.1.1-4.1.1.3 for linear systems.

Parameter Estimation

Similarly as in the case of linear in the parameter systems, the FDI problem boils down to estimating the parameters of the nonlinear model of the system (Fig. 4.6). The system in this case can be generally described by:

$$y_k = g(\phi_k, p_k) + v_k,$$

where ϕ_k may contain the previous or current system input u_k , the previous system or model output and the previous prediction error. The specific approach inherits all the drawbacks and advantages of its linear counterpart. For complex systems however there is an additional difficulty: the function $g(\cdot)$ is nonlinear in the parameters so non-linear parameter estimation techniques should be applied. This may cause serious problems with a fast reaction on faults and convergence to local minima.

Parity Relations

In exactly the same manner as for linear systems, the first step for the application of parity relations methods is to express the state and output equations on a time window $[k-S, k]$. In order to check the consistency however, the state variables have to be eliminated. This elimination step might be quite involved in the general case of nonlinear systems as was pointed out in [61] while the obtained relations contain nonlinear terms and are often implicit in the fault variables which implies limited practical applicability. In many aerospace applications however, the system can be expressed as nonlinear input affine (in the faults), leading to the so-called nonlinear inversion (NLI) methods. One such application was proposed for a nonlinear aircraft (missile) model in [62], where the force equations, which contain only measured (or reliably estimated) state variables and their derivatives, are used to estimate the control surface deflections of the missile. Since the commanded deflections can easily be obtained from the control system, structured residuals can be constructed from the difference between estimated and commanded deflections. The practical use of the above method demands the availability of linear and/or angular acceleration measurements and direct appearance of control surface deflection on the force and/or moments equation. The latter is usual to most aircraft/UAV simplified models, however it is very difficult to obtain the angular acceleration measurements involved from current sensor

technology. Even if angular acceleration sensors are available, their accuracy and cost/weight will be prohibiting compared to the possibility of direct control surface deflection measurement using encoders. Linear acceleration sensors, on the other hand, are widely used but there are a lot of aircraft models where control surface deflections do not appear explicitly in the force equations.

Observers

Model linearization is a straightforward way of extending the applicability of linear observer methods to nonlinear systems. Extended Kalman Filter (EKF) approaches as well as Extended Luenberger observers and Extended Unknown Input Observers (EUIO) [37], are based on this concept. Application of the above observers for aircraft systems were proposed in [10] and [41]. These approaches lead to relatively easy computation however their main drawback is that they work well only when there is no large mismatch between the model linearized around the current state estimate and the nonlinear behavior of the system. It should however be pointed out that EKF has been proved very reliable in the case of state estimation for nonlinear systems as well as in cases when the Gaussian noise assumption is clearly violated and is in use in many current INS/GPS fusion algorithms in use in current aircrafts.

Contrary to the previous extensions of the Kalman filter, the Unscented Kalman Filter (UKF) does not linearize the model. This technique predicts the system behavior by using evaluations of the nonlinear model at a set of points approximating a Gaussian distribution of the state vector [63]. Based on a similar idea but without being based on a Gaussian belief distribution, sequential Monte Carlo methods such as Particle Filtering (PF) are a very promising approach to deal with non-linearity. The basic idea of PF is to approximate the belief Power Density Function (PDF) at each instant with the sum of (a large number of) Dirac functions and to make them evolve at each time instant based on the latest observed data. Each Dirac function used in the approximation is called a particle. In the case of fault detection, the PDF that is approximated is the state of the system as well as the fault mode of operation. The use of PF for fault diagnosis was presented in [64] and [65]. In order for the PF method to deliver reliable estimates, a large number of samples (particles) are required, increasing the computational burden. Also, since failures are a rather rare event, the PF methods suffer from degeneracy

problems and a good re-sampling strategy is needed. What's more, a reliable statistical model of the system should be available.

Another way of handling nonlinear systems FDI, is the design of observers for special nonlinear systems like polynomial or Lipschitz systems [66], or through a nonlinear transformation. This is the non linear geometric approach proposed initially in [67], which aims at finding a state and output coordinates transformation that leads to a new set of observable decoupled residuals. The method was extended to be able to provide an estimate of the fault and was applied on a simplified nonlinear aircraft model in [36], combined with adaptive filtering and particle filtering. The use of this method for an aircraft however, required the transformation of the aircraft nonlinear system to an input affine nonlinear system, something that cannot be done if the aerodynamic model is too complicated.

Many other observer structures have been proposed, from the Sliding Mode Observer (SMO) [68], to nonlinear adaptive observers [36] and high gain observers [70]. SMOs are particularly interesting for FDD purposes since they are also able to reconstruct the fault rather than just detect it through a residual signal and are robust with respect to modeling uncertainties. An application of SMOs to a civil aircraft was proposed in [70] and flight tested in a high fidelity simulator. Recently a unified theory of nonlinear observers is presented [71].

To avoid the complexity of nonlinear observers, some new and unconventional methods have appeared in the literature especially for applications of aircraft systems. Examples of these methods are Linear Parameter Varying (LPV) based FDI and the use of multiple model strategies.

LPV based FDI is motivated by the problem of coping with a wide range of operating conditions. Such an FDI system has inherent performance and stability guarantees for the full operating conditions compared to multiple-model or gain-scheduling based FDI [72]. Of course its application requires additional modeling efforts.

Similar to the LPV based FDI, the idea of multiple model representation approaches is to apprehend the global behavior of the system by a set of local models (linear or affine), each of them characterizing the behavior of the system in a particular zone of operation [73]. Then the nonlinear system can be approximated by a Takagi-Sugeno (T-S) fuzzy model. In this way, one can design one observer for each of the local models and then synthesize a global observer by interpolation of the local

observers. This interpolation is actually obtained through the same activation functions as the T-S fuzzy model. The specific approach allows the implementation of well known linear observer design techniques while can guarantee global convergence. The only drawback is that it sometimes leads to conservative designs.

4.1.1.5 Residual Evaluation

Once the residuals are generated by one of the residual generation methods stated above, the residual evaluation logic is used to detect and isolate any fault occurrence. The residual processing methods can be based on simple residual geometrical analysis or comparison with fixed thresholds [32], [36].

In general, in the absence of faults, the residual signals are approximately zero. In practical situations however, the residual is never zero, even when no faults occur. This is caused by the dependence of the residual to the input of the system, by modeling uncertainties and noise and disturbances that affect the system. A threshold must then be selected suitably larger than the largest magnitude of the residual in the fault free case. It is obvious that the threshold setting is a compromise between the need for fast fault detection (high fault sensitivity) and the false alarm rate.

The simplest and more widely used residual evaluation scheme is the selection of a fixed threshold. This threshold can be selected based on experience or by knowledge of the disturbances acting on the system. In the latter case, systematic threshold computation is possible using the well established robust control theory (norm based residual evaluation) [25].

It is well known that the system input almost always affects the dynamics of the residual generator. From this point of view, the input acts as a disturbance however there is a significant difference: in most cases the input is known exactly. In order to improve the FDI systems performance this knowledge should be integrated into FDI system design and operation. This can be done in the form of an adaptive threshold [75]. The adaptive threshold concept is essential in order to make the FDI system robust to disturbances that cannot be suppressed in the residual generation process. This is the so called passive robust fault detection that can be applied using interval analysis [76].

Another way to treat the residual is by statistical testing. The residual is treated in this case as a stochastic variable with mean and variance. These properties change due to faults and

techniques of change detection such as a likelihood ratio test, generalized likelihood ratio test, Neyman Pearson test, Sequential probability ratio test (SPRT) etc is commonly used [77].

4.1.1.6 On Fault Isolation

Fault isolation is one of the central tasks of a fault diagnosis system, a task that can become, by many practical applications, a real challenge for the system designer. Generally speaking fault isolation is a signal processing process aiming at gaining information about the locations of the faults occurred in the process under consideration. It is evident that fault isolation is necessary when any action needs to be applied to counteract the faults consequences (apart from cutting down the system). In order for fault isolation to be possible, the effects of the different faults on the residuals should be distinguishable. The main strategies to accomplish fault isolation are:

Directional Residuals

In this approach, the residual generation problem is addressed within a geometric framework. The idea is to design a directional residual vector that lies in a fixed and fault-specific direction (or subspace) in the residual space in response to that particular fault. The Fault Detection Filter, proposed by Beard and Jones since the 70's, is one of the pioneering methods and has actually inspired the directional residual concept, while a recent work on the same framework is [67].

Although directional residuals are simpler to implement and can also provide more reliable fault isolation under ideal conditions, it is really difficult to make them robust against various sources of uncertainties, especially modeling errors and system disturbances. Also after the design objectives have been met for fault isolation, no more design freedom is left for other goals to be accomplished by the residuals (i.e. speed of the response).

Structured Residuals

In this case, which is the most common, each residual is designed to be sensitive to a number of faults, while remaining insensitive to the remaining ones. The design procedure consists of two steps: the first step is to specify the sensitivity and insensitivity relationships between residuals and faults according to the assigned isolation task and the second is to design a set of residual generators according to the desired sensitivity and insensitivity relationships [32]. The structured residuals can be designed in two conceptually different ways, namely dedicated residual set and generalized residual set. The two schemes are shown in figure 4.10 for an example of isolating three faults [f_1 , f_2 , f_3].

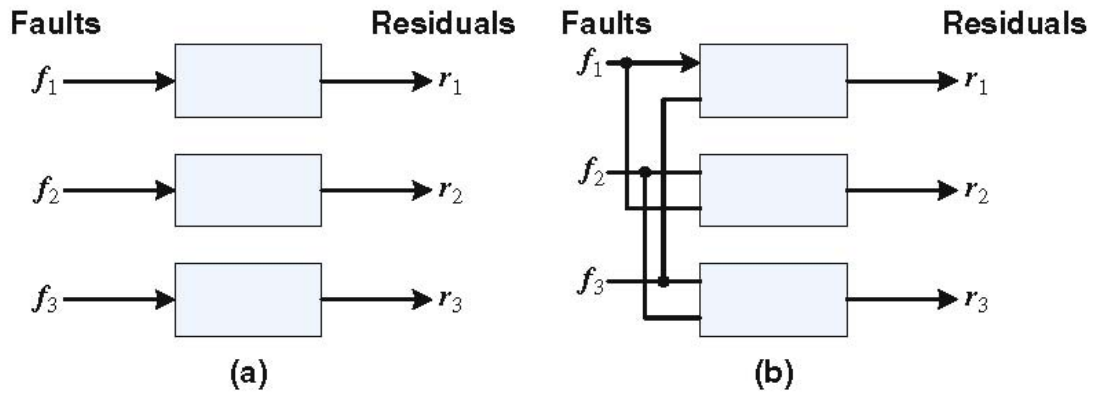


Figure 4.10 Two schemes of structured residual sets for the isolation of three faults: (a) Dedicated scheme and (b) Generalized residual scheme.

In the dedicated scheme every residual is designed to be sensitive to one and only fault and a simple logic can be used about the appearance of a specific fault. Various fault isolation techniques have been developed in the literature under the dedicated scheme like the dedicated observer scheme (DOS) [74]. Another very important group of fault isolation methods that essentially fall under the dedicated scheme are the multiple model (MM) approaches. Such approaches have been applied to aircraft FDI [42]-[45].

In the generalized scheme, each residual is sensitive to all faults but one. This means that a more complicated logic must be applied for fault isolation. If a bank of observers is used for generation of all residuals in the generalized residual set the structure is known as Generalized Observer Scheme (GOS). The

GOS-based FDI is more robust than DOS with respect to parameter uncertainties and measurement noise. This is mainly due to the fact that more than one output (and thus more information) is fed to each observer.

4.1.1.7 On Fault Identification

Despite its undeniable importance, model based fault identification has received less attention from the research community as compared to model based FDI. This is especially true for nonlinear systems. Nevertheless, knowledge of the faults severity is essential if any action of reconfiguration is to be taken other than shutting down the process or switching to a different component.

One way to deal with the problem is to treat the fault magnitude as a parameter and estimate it using a suitable parameter estimation technique. This approach was proposed by Isermann [78] and developed for linear systems. More recently, Tan and Edwards [79], applied the concept of “equivalent output estimation error injection” to reconstruct faults for linear and a class of nonlinear systems using sliding mode observers. One may also use multiple-model approach for fault identification where every model in the bank corresponds to a different fault severity, however this will introduce an inevitable quantization error in fault estimation. Zhang and Jiang [80] have also developed a two-stage adaptive Kalman filter for simultaneous state and fault parameter estimation which is applicable to identification of only actuator (not component) faults.

Criteria\ Method	Parity Space	Parameter Estimation	Observer Single (FDF)	Multiple Observers	Multiple Kalman Filters	Two Stage Kalman Filter	Extended Kalman Filter	LPV & Multiple Model representation	Sliding Mode Observers	Interval Observers (Set Membership Estimation)
Sensor Fault	✓	*	✓	✓	✓	✓	✓	✓	✓	✓
Actuator Fault	+	✓	+	+	✓	✓	✓	✓	✓	✓
Structure Fault	+	✓	+	+	✓	✓	✓	✓	✓	✓
Speed of Detection	✓	*	✓	✓	✓	✓	✓	✓	✓	*
Isolability	✓	✓	*	✓	✓	✓	✓	✓	✓	✓
Identifiability	*	✓	x	x	*	✓	✓	*	✓	✓
Suitability for FTC	x	✓	x	+	*	✓	✓	✓	✓	✓
Nonlinear Systems	*	+	x	+	+	✓	✓	✓	*	✓
Robustness	✓	+	-	*	*	+	+	*	✓	✓
Low Conservatism	✓	✓	✓	✓	✓	✓	✓	x	✓	*
Computational Complexity	✓	✓	✓	*	*	✓	*	✓	✓	*

Note: (✓) favorable, (*) less favorable, (x) not favorable, (+) applicable but with limitations, (-) not applicable.

Table 4.1 Comparison of different FDI methods and their properties (partially taken from [22]).

4.2 Fault Tolerant Control (FTC)

Fault tolerant control is an active field of research and there are numerous publications in the literature referring to this subject ([4] and [22] are excellent surveys). Aircrafts, as safety critical systems, are among the most cited applications of FTC. Over the last 20 years, several research projects have explored the application of FTC on aircraft systems. The earliest results on FTC for aircraft were accomplished during the Self-Repairing Flight Control Systems (SRFCS) program sponsored by the US Air Force Wright Research and Development Center in 1984. The program led to successful flight tests on F-15 aircraft performed by NASA in 1989 and 1990 [84]. The Propulsion Control Aircraft (PCA) was also developed by NASA Dryden Center following the Sioux City accident and was successfully flight tested on several aircrafts in 1990s. Within the 1999-2004 Intelligent Flight Control System (IFCS) F-15 program, sponsored by NASA Dryden, pre-trained and on-line learning neural networks were flight tested on the NASA IFCS F-15 testbed. Other programs like the RESTORE program conducted by USAF, made significant advances towards the application of FTC for a tailless aircraft. A good survey paper for these programs is [85]. Recently, the GARTEUR Flight Mechanics Action Group FM-AG(16) on Fault Tolerant Flight Control, brought together well known universities and industries for the exploration of the applicability of FTC on civil aircrafts. The proposed methods were evaluated on SIMONA high fidelity flight simulator available at Delf Technical University [4].

An up to date classification of FTC methods (both passive and active) along with some of their applications on aircraft or fixed wing UAV systems is shown in figure 4.11.

Despite the above mentioned programs, FTC is for the most part still in academic notion and there are very few controllers implemented on physical systems (none on civil aircraft). The main reason for this resides in the fact that FTC involves an authority restriction for the pilot something that both the industry and the public are reluctant to accept. What's more, FTC methods are not mature enough to prove their applicability. Low cost and risk systems like UAVs, are a good candidate for FTC systems since the absence of a pilot makes the need for such systems even greater and their implementation does not involve increased risk for human lives.

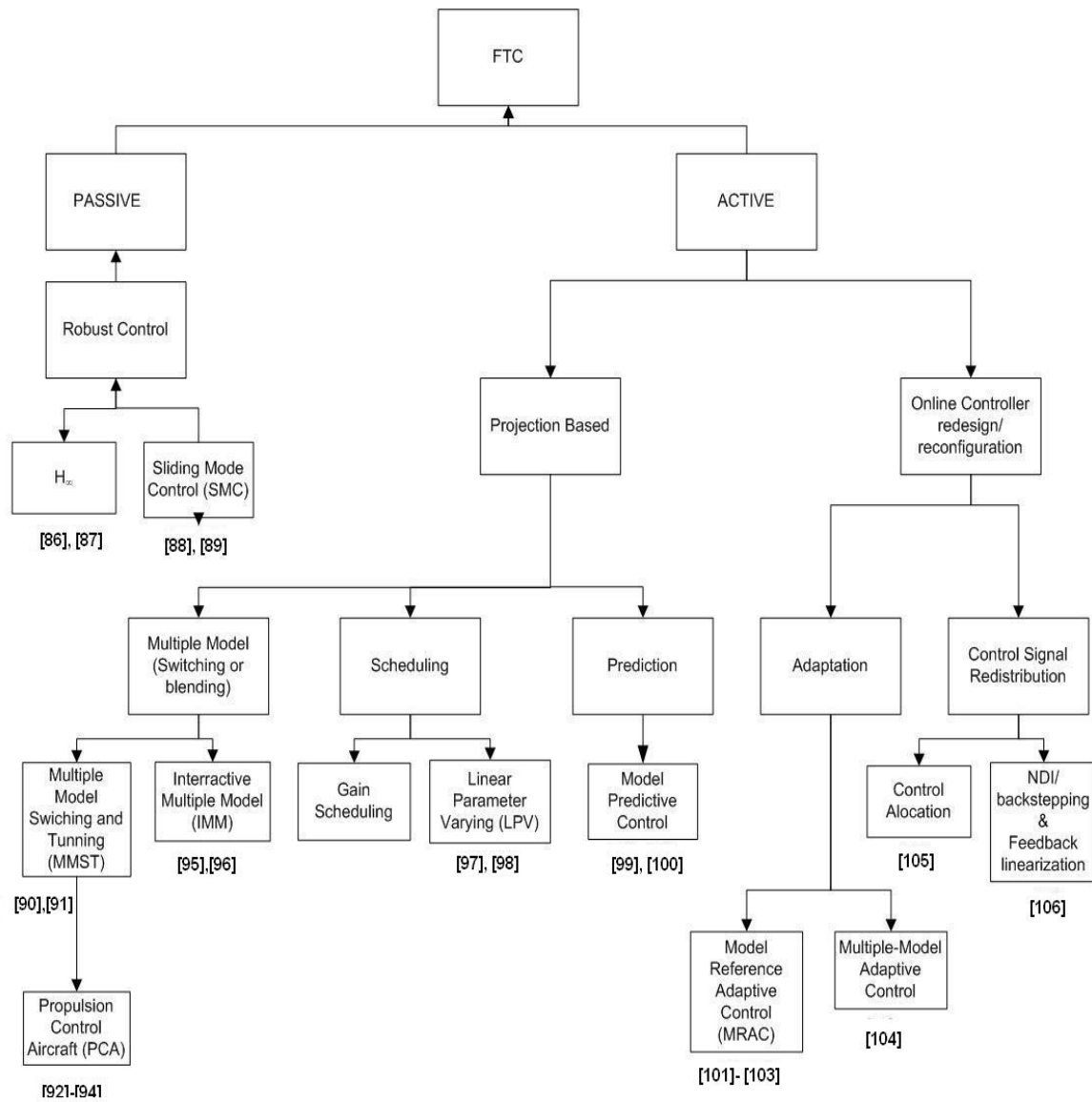


Figure 4.11 Classification of Fault-Tolerant Flight Control Methods with recent examples

The bibliographic review shows a trend from the well-known but difficult to apply on real (non-linear) systems linear control methods, to non-linear techniques for FTC. This is mainly caused by the fact that in real life every system has non-linearities and even though if it can be treated as piecewise linear, the introduction of faults lead to highly non-linear behavior. This is particularly true for aircraft systems. They have been traditionally treated as linear systems around equilibrium points and the widely applied method for flight control design in the industry is still gain scheduling. However even in the presence of simple faults (like a stuck actuator), the linearity assumption breaks and strong cross-coupling terms appear. In this thesis a non-linear technique for

FTC was implemented and its merits and disadvantages were examined when used in conjunction with an FDI system.

4.3 Discussion on FDI-FTC method for UAVs

Features of the existing quantitative model based approaches to FDI are summarized in Table 4.1 (partially taken from [22]). It should be noted that not many comparative studies are published in the literature. A UAV is a nonlinear system although its missions (primarily surveillance) allow it to remain close to steady state for long times. This means that linear techniques could be applied. One approach is the design of several FDI systems in several operating conditions and then the interpolation of the designs to cover the full flight envelope. This is actually the industrial practice for almost every aircraft system to date for control system design. The drawback of this approach is the risk of increased false alarms between transitions and during maneuvers that take the aircraft away from the design operating points. Also the design operating points must be chosen carefully and a gain scheduling policy should be found by intuition. The main advantage on the other hand is the use of well developed linear techniques as well as the possibility of introducing robustness in the FDI design by the use of norm optimization methods (i.e H_∞). Such an approach could be selected if the UAV would be utilized for non-aggressive maneuvering.

Another promising approach is the design of LPV or multiple model observers to cover the full flight envelope in the FDI design. In this case convergence of the FDI scheme is guaranteed and no gain scheduling is needed but conservativeness is inevitably introduced in the design. Furthermore, additional modeling efforts are required in order to obtain an LPV or multiple model representation of the system.

The use of a linearization method like Extended Kalman Filtering (EKF) or Extended Unknown Input Observers (EUIO) is another approach. Provided no big mismatch exist between nonlinear and linearized dynamics such techniques could provide an alternative solution to the need for gain scheduling. A drawback is that any disturbances should be modeled in order to apply EUIOs while EKF assumes white Gaussian noise in the model. Both approaches suffer from un-modeled dynamics and parameter uncertainties.

As far as robustness is concerned, any UAV will experience external disturbances due to air gusts and turbulence. As we will see, these disturbances enter the model in a state dependent manner constant for a specific operating condition. Also there will always be uncertainty in the aerodynamic parameters. These parameters are identified in a wind tunnel or in-flight for every aerospace vehicle as we saw in Chapter 2, however for a UAV (due mainly to reduced funding) no detailed modeling is attempted. It is logical to assume therefore that parameter uncertainties that are bounded exist in the UAV model. These uncertainties can cause significant problems in the FDI module and should be taken into account.

From the point of view of fault tolerant control requirements it is obvious that the speed of detection is of vital importance due to fact that the aircraft is a very fast dynamic system that cannot (unfortunately) be shut down. Missed detection is unacceptable and thus the FDI systems sensitivity even of small faults or of faults that do not affect the present operating condition (close to steady state) should be high since these faults can become lethal in a different maneuver. On the other hand some false alarms may be acceptable as long as reconfiguration is fast and the performance of the aircraft is not degraded too much. In order to design a simple and reliable fault tolerant control scheme fault isolation is essential while fault identification is useful. Finally the computational burden should be acceptable since due to reduced payload the computational power onboard is limited. Furthermore these resources will have to be split to other function like the FTC algorithm itself. Summarizing an FDI scheme for a UAV should have the following properties:

- (a) Promptness of detection
- (b) Sensitivity to small or slowly developing faults
- (c) No missed fault detection
- (d) Low false alarm rate
- (e) Accurate fault isolation and preferably identification
- (f) Robustness
- (g) Low computational burden

The design of a FTC system has to be based on the information provided by the FDI module. Although a passive approach on FTC can be selected so that an FDI module is not needed, the faults that can be treated by such an approach are limited and a compromise exists between the faults that can be

tolerated and the performance of the control system. An active FTC system is usually a viable approach. In this case, the overall performance of the system should be assessed since the performance of the fault tolerant controller depends on the promptness and the accuracy of the information it receives from the FDI system. A joint design approach, based on a stochastic analysis of the performance of FDI was proposed in [107] for linear systems. Another issue is the reduction in acceptable performance after a failure, an issue not well-studied so far [108]-[110]. The treatment of the initial period during which a fault is detected but not isolated (or identified) is another important issue that deserves further research [111]. Summarizing an FTC scheme for a UAV should have the following properties:

- (a) Non-linear system applicability
- (b) False alarm handling
- (c) Robustness to initial low information period immediately after the fault
- (d) Robustness to system miss-modeling and disturbances
- (e) Low computational burden
- (f) Possibility of being added to the nominal controller of the system (any flight control system that will have to be re-designed from scratch during the fault's occurrence will probably be treated by skepticism, since a simple flight control system design is an iterative procedure that demands a lot of engineering judgment).

A comparison of FTC methods as evaluated in the GARTEUR AM(16) Action Group is provided in [4] and presented in Table 4-2 below.

Method	Failures		Robust	Adaptive	Fault Model		Constraints	Model Type	
	Actuator	Structural			FDI	Assumed		Linear	Nonlinear
Multiple Model Switching and Tuning (MMST)		•		•	•			•	
Interacting Multiple Model (IMM)		•		•	•		○	•	
Propulsion Controlled Aircraft (PCA)	•		○			•		•	•
Control Allocation (CA)*	•				•		○	•	
Feedback Linearization	•	•		•	•				•
Sliding Mode Control (SMC)*	○ ¹	•	• ²				•		•
Eigenstructure Assignment (EA)		•				•		•	
Pseudo Inverse Method (PIM)		•				•		•	
Model Reference Adaptive Control (MRAC)*		•		•	•			•	○
Model Predictive Control (MPC)*	•	•	○	○	•	•	•	•	•

Comparison of reconfigurable control methods

* Evaluated in this Action Group

1: Can handle partial loss of effectiveness of actuators, but not complete loss

2: Assumes robust control can handle all forms of structural failures

Table 4.2. Comparison of FTC techniques and their properties [4]

Chapter 5 - Nonlinear Fault Detection and Identification System based on a two step method.

The UAV system is a nonlinear system. Although there are missions during which the behavior of a UAV can be considered linear and be kept close to some predefined operating condition, there are situations where high maneuverability is desired. Also there are phases of the flight (i.e. during take-off and landing) when the nonlinear behavior dominates. An FDI system based on the interpolation (gain scheduling) of linear FDI routines could experience high false alarm rates. Moreover the gain scheduling is a tedious task involving a lot of engineering judgment.

In this thesis a nonlinear FDI technique is used to detect, isolate and estimate actuator failures. This is a two step procedure: In the first step, a multiple model adaptive estimation method based on Extended Kalman Filters is used to detect stuck actuators and to provide filtered estimates of the angular rates of the UAV. Possible biases in the measured quantities can be corrected in this step. As long as no stuck failure is detected, the filtered angular rates along with the commanded deflections from the control system are used to provide estimates of the aerodynamic derivatives including the effectiveness of the control surfaces using a linear parameter estimation method.

5.1 Multiple Model Adaptive Estimation (MMAE)

As mentioned in chapter 4, in MMAE method, a bank of Kalman Filters (KFs) are used running in parallel, each of which is matching a particular fault status of the system. A hypothesis testing algorithm uses the residuals from each Kalman Filter to assign a conditional probability to each fault hypothesis. The use of Kalman Filters requires a linear system. What's more a Kalman Filter can be used to monitor a specific failure (a specific value of control effectiveness for example or a particular angle of stuck surface). This leads to an enormous number of KFs required in order to span the range of possible fault scenarios, which is limited by the computational load.

These limitations can be avoided by the use of Extended Kalman Filters (EKF). One such filter can monitor the health status of one actuator and also provide an estimate of the stuck failure at the same time. One additional EKF is required to

represent the no-fault scenario. This arrangement is presented in figure 5.1.

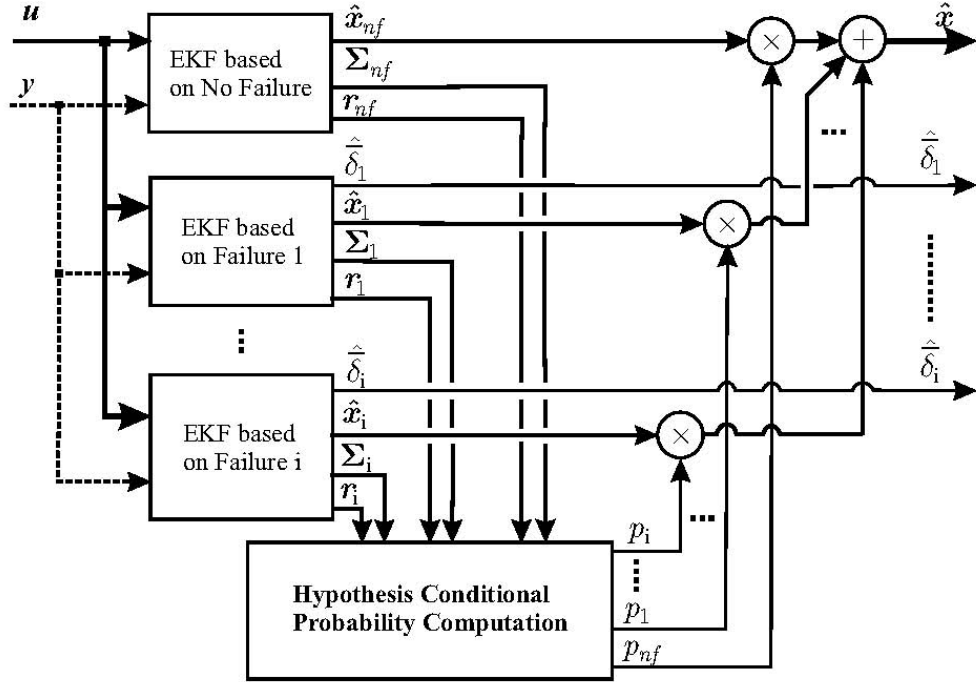


Figure 5.1 MMAE with Extended Kalman Filtering. Each Filter monitors its assigned actuator.

The above arrangement implies that the use of six EKFs running in parallel is adequate for the monitoring of the actuators since there are five control surfaces (two ailerons, two elevators and a rudder). Although the computational load of the above scheme is important its implementation is feasible due to the increased power of modern multi-thread computers and the possibility to implement these filters in different computer boards running in parallel.

5.2 Design of the Extended Kalman Filters

The EKFs are designed based on a set of continuous differential equations that describe the plant under consideration:

$$\dot{x} = f(x, u) + w \quad (5-1)$$

$$y = h(x) + v \quad (5-2)$$

where x is the state vector, u the input vector, y the output vector, $f(x, u)$ the set of nonlinear functions of the state and control, h a set of (possible) nonlinear functions of the state, w is the random zero mean Gaussian process noise vector with covariance matrix Q

($E\{ww^T\}$) and v the random zero mean Gaussian measurement noise vector with covariance matrix R ($E\{vv^T\}$).

In our case the state vector is $x = [p \ q \ r \ a \ b]^T$ and the control vector is $u = [\delta a1 \ \delta a2 \ \delta e1 \ \delta e2 \ \delta r]^T$. The set of nonlinear state equations are those described in equation (2-7) while it is assumed that the states are measured directly (something that is a logical assumption for even the smallest UAVs containing a simple pitot tube and an IMU unit). The dynamics of the measurement sensors are neglected.

According to standard textbooks [39], [81], EKFs are similar to the linear Kalman Filters with the state and measurement equations being linearized along the estimated trajectory. The implementation equations are the same, however because the system and measurement equations are nonlinear, a first-order approximation is used in the continuous Riccati equations for the systems dynamics matrix F and the measurement matrix H . The matrices are related to the nonlinear system and measurement equations according to:

$$F(k) = \left. \frac{\partial f(x, u)}{\partial x} \right|_{x=\hat{x}_k, u=u_k} \quad (5-3)$$

$$H(k) = \left. \frac{\partial h(x)}{\partial x} \right|_{x=\hat{x}_k} \quad (5-4)$$

The discrete transition matrix is approximated as $\phi(k) \approx I + F(k)T_s$, where T_s is the sampling time.

The method can be considered as consisting of two steps. The state propagation step and the measurement update. The computation steps are:

A. State Propagation Step:

1. The state is propagated forward according to the state equations. This can be accomplished either using the transition matrix or by integrating the actual nonlinear differential equations forward at each sampling interval. The latter is more accurate and a simple Euler integration technique is used in this thesis:

$$\bar{x}_k = \hat{x}_{k-1} + \dot{\hat{x}}_{k-1}T_s, \quad \dot{\hat{x}}_{k-1} = f(\hat{x}_{k-1}, u_{k-1}) \quad (5-5)$$

2. The state error covariance matrix is propagated forward using the discrete transition matrix and the process noise covariance matrix:

$$\bar{P}_k = \phi_{k-1} \hat{P}_{k-1} \phi_{k-1}^T + Q_{k-1} \quad (5-6)$$

Where $Q_{k-1} = \int_0^{T_s} \phi(\tau) Q \phi(\tau)^T d\tau = G_k Q G_k^T, G_k = T_s \left. \frac{\partial f(x, u)}{\partial u} \right|_{V_{T_k}}$ (5-7)

B. Measurement Update Step:

1. The Kalman Gain matrix is computed:

$$K_k = \bar{P}_k H_k^T [H_k \bar{P}_k H_k^T + R_k]^{-1} \quad (5-8)$$

2. The Kalman Gain is used to correct the state and the state error covariance matrix:

$$\hat{x}_k = \bar{x}_k + K_k [y_k - h(\bar{x}_k)] \quad (5-9)$$

$$\hat{P}_k = [I - K_k H_k] \bar{P}_k \quad (5-10)$$

To avoid numerical problems during filter operation, equation (5-10) was replaced by the Joseph form for the state error covariance measurement update:

$$\hat{P}_k = [I - K_k H_k] \bar{P}_k [I - K_k H_k]^T + K_k R_k K_k^T \quad (5-11)$$

The filter representing the no fault scenario is fed by all inputs and outputs. The matrices F, H and G can be evaluated analytically from the equations (2-7) and (5-3), (5-4) and (5-7). The other filters that monitor one actuator each, need some modification. In order to estimate the deflection of the failed actuator, this deflection is going to augment the state of these filters. Therefore the state vector for each filter i, is:

$$z_i = \begin{bmatrix} x \\ \bar{\delta}_i \end{bmatrix}, \text{ where } \bar{\delta}_i \text{ is the faulty control signal}$$

caused by the jammed or floating actuator. The augmentation of the state vector leads to the following state space equations for each filter:

$$\begin{aligned} z_i(k+1) &= f_{z_i}(z_i(k), \delta(k)) + w_k \\ y_i(k) &= h(z_i(k)) + v_k \end{aligned} \quad (5-12)$$

where

$$f_{z_i}(z_i(k), \delta(k)) = \begin{bmatrix} f(z_i(k), \delta(k)) \\ \bar{\delta}_i(k) \end{bmatrix} \quad (5-13)$$

The matrices F, H and G are obtained by differentiation so that the linearized system evaluated at each sampling time can be written as:

$$\begin{aligned} \begin{bmatrix} x(k+1) \\ \bar{\delta}_i(k+1) \end{bmatrix} &= \begin{bmatrix} F(k) & G^i(k) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x(k) \\ \bar{\delta}_i(k) \end{bmatrix} + \begin{bmatrix} G^{(0,i)}(k) \\ 0 \end{bmatrix} \delta(k) \\ y(k) &= [H \quad 0] \begin{bmatrix} x(k) \\ \bar{\delta}_i(k) \end{bmatrix} \end{aligned} \quad (5-14)$$

The matrices $G^i(k)$, $G^{(0,i)}(k)$ are evaluated from the input matrix G (equation 5-7). $G^i(k)$ represents the i th column of G while $G^{(0,i)}(k)$ is matrix G with its i th column set to zero. By $\delta(k)$ we represent the input vector u . A critical design parameter for the EKFs is the selection of the process and measurement noise covariance matrices Q, R. These were selected by a trial and error procedure as follows: $Q = 0.002 \times I_5$ and $R = \text{diag}[0.1 \times I_3 \quad 0.02 \times I_2]$.

The EKFs are implemented as Matlab function blocks in the simulation and the source code for each filter is included in the CD that accompanies the thesis.

5.3 Design of the Hypothesis Conditional Probability Computation Module

When Kalman filters are used for MMAE, the residuals and the state error covariance matrices from the filters can be used to assign a conditional probability to each fault scenario. These probabilities will be used for fault detection. Moreover the computed probabilities can be used to estimate the state vector of the system according to the formula:

$$\hat{x}(k) = \sum_i \hat{x}_i(k) p_i(k) \quad (5-15)$$

In the above formula $\hat{x}_i(k)$ is the state estimate computed by the EKF that assumes the fault scenario i and $p_i(k)$ is the probability assigned in the specific fault scenario.

The on-line computation of the probability $p_i(k)$ is possible by the Bayes' law, and can be done analytically assuming that the probability densities are Gaussian functions, or the residuals of the EKF are Gaussian distributed. As long as the linearization procedure of the EKF is efficiently representing the system and the dynamics of flight are not rapidly changing this assumption is a logical one. It can be shown that the fault probability can be evaluated at each time step k as:

$$p_i(k) = \frac{p[y = y_k | (fault = i, Y_{k-1})] p_i(k-1)}{\sum_{j=0}^N p[y = y_k | (fault = j, Y_{k-1})] p_j(k-1)} \quad (5-16)$$

Where

$\sum_{j=0}^N p[y = y_k | (fault = j, Y_{k-1})] p_j(k-1)$ is the sum of all scenarios probabilities such that the fault probabilities add up to one, $p_i(k-1)$ is the probability of the fault scenario i at the previous time step and $p[y = y_k | (fault = i, Y_{k-1})]$ can be shown to be given by:

$$p[y = y_k | (fault = i, Y_{k-1})] = \frac{1}{(2\pi)^{m/2} \Sigma_i(k)^{1/2}} e^{-r_i(k)^T \Sigma_i(k)^{-1} r_i(k) / 2} \quad (5-17)$$

In the equation (5-17), m is the number of measurements, $r_i(k)$ and $\Sigma_i(k)$ are the residual and the residuals covariance matrix calculated at time step k from the i th EKF.

By examining the probabilities we can determine the “health status” of the system. An actuator fault is declared valid if the corresponding fault probability exceeds 80% for a certain amount of time. A fault can be declared removed when the corresponding fault probability drops below 5% for a certain amount of time. This method that uses probabilities for fault isolation is sometimes called a Bayes classifier [1].

The prior probabilities used by the recursive algorithm can be chosen equal or according to MTBF (Mean Time Between Failures) data available for the actuators. In practice the probabilities should not be allowed to reach zero as they will stay to zero ever after, so a lower bound for each probability is set to 0.001.

The method was tested in a simple scenario: The UAV is flying close to a trim condition (straight and level flight) and the faulty actuator is stuck with a deflection close to the trim value, making the detection of a fault difficult. The simulation is performed in open loop with no controller included. The fault is injected at the

left aileron whose deflection is fixed at time equal to 35s to 0.02 or 0.5 degrees (from a trim value of 0) The results of the simulation are presented in figure 5.2:

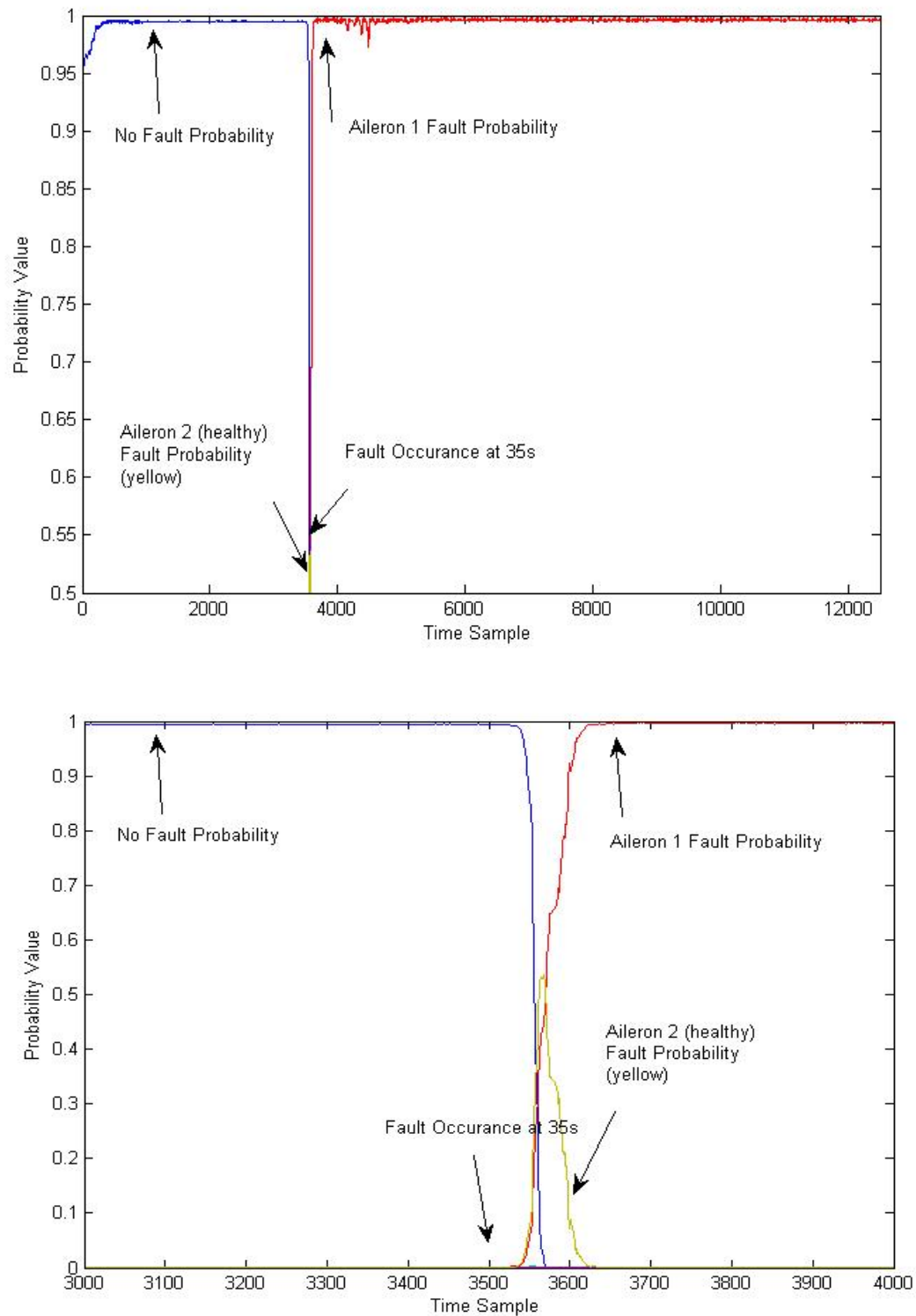


Figure 5.2. Aileron 1 fault detection

We can see that the detection is almost immediate (1s) although there is a false probability rising for the healthy right aileron which quickly drops to zero. In figure 5.3 we can see the estimation of the faulty aileron deflection. The estimation is valid after the declaration of the fault and we can see that the estimated deflection closely follows the true faulty one immediately after the failure.

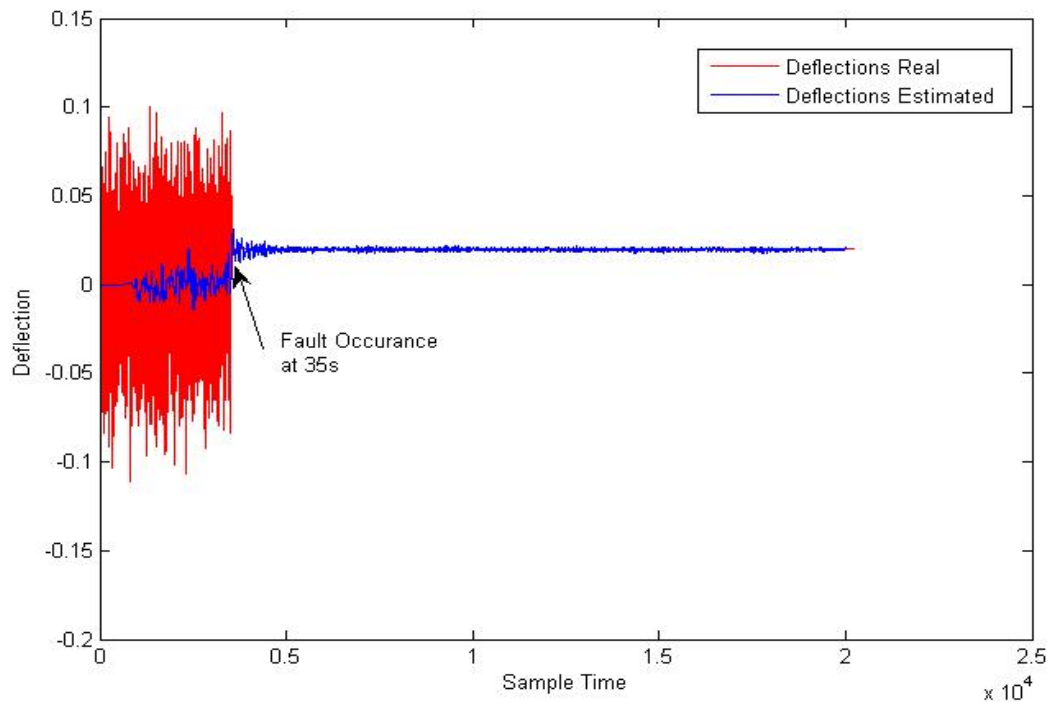


Figure 5.3. Aileron 1 faulty deflection estimation
(Stuck failure at 0.02)

In the case of a floating actuator, the results are similar as can be seen in figure 5.4 for the left aileron, however in this case the detection time is longer (approximately 5s), caused by the low excitation of the system due to the failure in the specific flight condition. When a control surface is floating its contribution is zero and so the commanded faulty surface deflection is equal to zero. The MMAE method can efficiently isolate and identify a floating actuator. The correct estimation of the failure is demonstrated in figure 5.5. Similar results were obtained for the other control surfaces and were omitted for brevity. It should be emphasized that for the particular flight condition, the time of detection in the case of the floating aileron actuator (5s), was the worst observed, since floating failures in other control surfaces (elevator, rudder) was much faster (1.2s-3s).

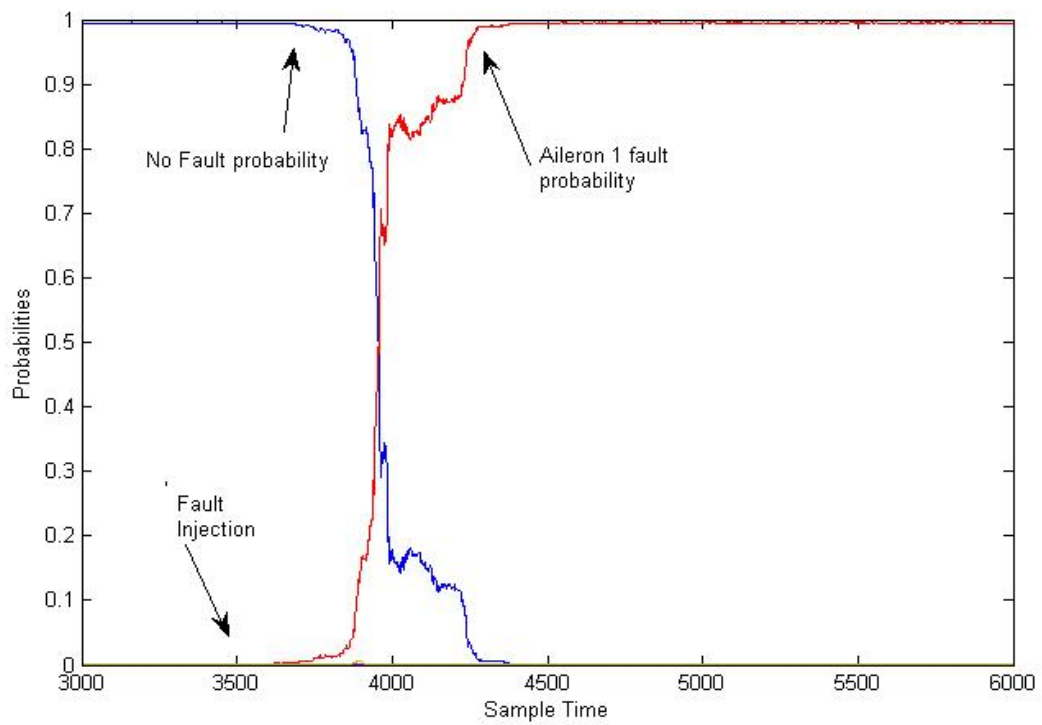


Figure 5.4. Aileron 1 floating failure detection

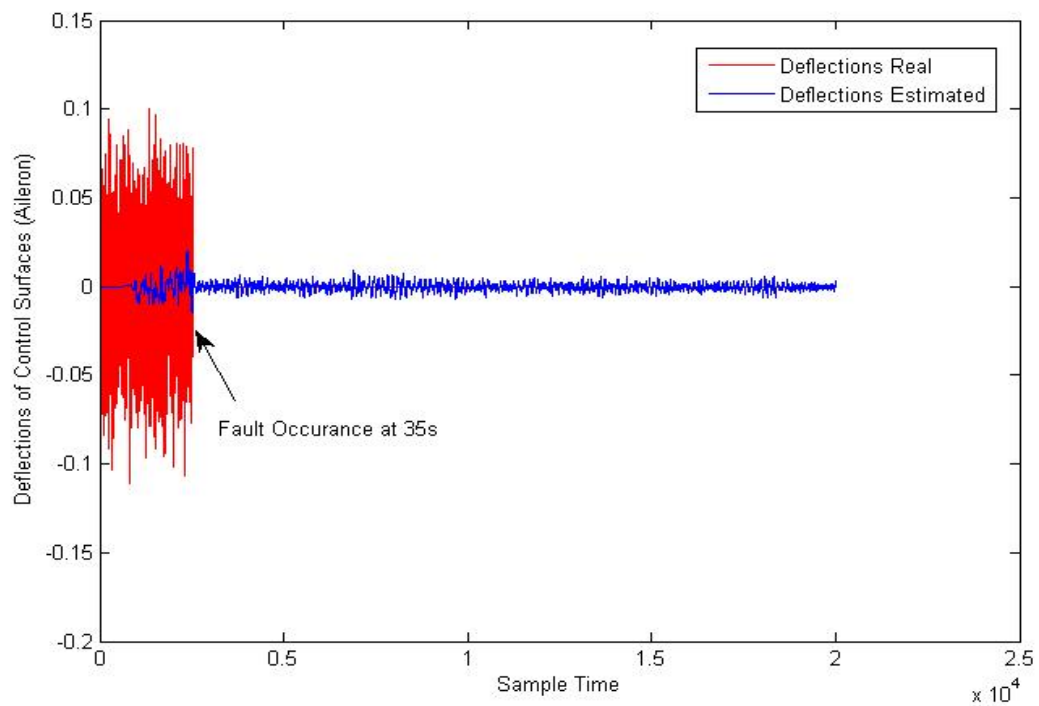


Figure 5.5. Aileron 1 faulty deflection estimation (Floating failure)

The introduced system is capable of detecting and identifying a stuck or floating failure in the actuators however in a UAV, especially when it is operated in a harsh environment, there is a possibility of a structural failure. The identification of major structural failure in a control surface (i.e loss of a part of the surface) can be modeled as a loss of control effectiveness of the specific surface. In the literature usually linear time invariant systems are treated where the loss of control effectiveness is treated as a loss parameter included in the control matrix B multiplied with the control deflection (i.e δa where δa is the combined symmetrical movement of the ailerons). This approach is limited by the fact that a structural damage is likely to occur in a non-uniform manner for a pair of control surfaces (i.e the left aileron can be damaged only or the damages can be different for the two surfaces). In this case, apart from the reduction in control surface effectiveness, the symmetrical movement of the control surfaces can lead to the induction of moments on the other axes as well. For example a loss of a part of the left elevator will lead to a loss of pitch control effectiveness but due to the different lift produced by the two elevators its symmetrical deflection will induce a roll moment as well.

In order to identify such failures the MMAE method introduced so far should be extended. An obvious approach would have been to include more EKF's to represent this kind of failures. The failure can be modeled as a loss factor multiplying the surface effectiveness aerodynamic derivative in the force and moments equations of the UAV. However, a control surface damage is in practice a much more complicated case [18], [19], [20] and [31] and apart from the aerodynamic derivative of the control surface itself, other aerodynamic derivatives are affected as well. It is felt that in order to detect, isolate and identify this kind of failures a more general approach should be used.

5.4 Parameter Estimation method for control surface damage Fault Detection and Isolation

Since a structural damage affects the forces and moments modeling, a parameter estimation method could be used to identify these changes. A major structural damage can alter the way these forces and moments can be modeled. What's more, it alters the center of gravity of the aircraft as well as the matrix of inertia. In this case, a complete re-identification of the system with unknown structure is extremely difficult especially in the limited time the control system has in order to prevent a potential crash. In the case of control surface damage, when a part of a control surface is missing, it is logical to assume that the structure of the aerodynamic model does not change. Also it is unlikely that the other properties of the aircraft will be affected. There are only changes in the aerodynamic derivatives (mainly those of the control surface but also other derivatives concerning aircraft states might be affected as well) and these changes could be identified.

Parameter identification is commonly applied to aircraft especially during wind tunnel development and flight testing [16]. The most commonly used method is the maximum likelihood parameter estimation which is not applicable on-line as it iterates through all the data gathered (batch method) and is a nonlinear parameter estimation technique. One of the few methods that can be implemented in real time is the so-called filtering error method developed at DLR [29]. This is a joint state and parameter estimation algorithm which is very complex. Other algorithms are the EKF for both state and parameter estimation, which is easy to implement but due to the correlation of the parameter and state estimates the accuracy of the former can be decreased especially if the number of the identified parameters becomes large. The two-step method [4] can be used to decouple the state and parameter estimation. In the first step the data from all on-board sensors is used to estimate accurately the state as well as biases in the measurements in a nonlinear state estimation problem. An EKF or other fusion algorithm like particle filtering can be used in this stage. In the second step the states are used to identify the aerodynamic parameters in a linear parameter estimation problem. A least squares algorithm like Recursive Least Squares (RLS), Exponentially Weighted Least Squares (WLS) or Sequential Least Squares (SLS) can be used in this step. Alternatively a frequency based method can be used [28].

In our case the state of the system (in fact mainly the angular rates are of interest) can be estimated by the MMAE algorithm presented in previous sections. The deflections of the control surfaces are not measured directly but can be supplied by the controller. Assuming that no stuck failure is identified these commands can be safely assumed to be the actual deflections of the actuators. In practice however the dynamics of the actuators should be taken under consideration as well. If these dynamics are fast enough they can be neglected. The equations used from the parameter estimation algorithm are those of the moment coefficients:

$$\begin{aligned}
C_m &= \frac{1}{\bar{q}S\bar{c}} \left[I_y \dot{q} + (I_x - I_z)pr + I_{xz}(p^2 - r^2) \right] = C_{m_\alpha} \alpha + C_{m_{\dot{q}}} \dot{q} + \boxed{C_{m_{\delta a}} \delta a} + C_{m_{\delta e}} \delta e + C_{m_i} \\
C_l &= \frac{1}{\bar{q}Sb} \left[I_x \dot{p} - I_{xz}(pq + \dot{r}) + (I_z - I_y)qr \right] = C_{l_\beta} \beta + C_{l_{\dot{r}}} \dot{r} + C_{l_{\dot{p}}} \dot{p} + C_{l_{\delta a}} \delta a + \boxed{C_{l_{\delta e}} \delta e + C_{l_{\delta r}} \delta r} \\
C_n &= \frac{1}{\bar{q}Sb} \left[I_z \dot{r} - I_{xz}(\dot{p} - qr) + (I_y - I_x)pq \right] = C_{n_\beta} \beta + C_{n_{\dot{r}}} \dot{r} + C_{n_{\dot{p}}} \dot{p} + \boxed{C_{n_{\delta a}} \delta a}
\end{aligned} \tag{5-18}$$

The main difficulty using equations (5-18) is the accurate estimation of the angular rate derivatives. Angular acceleration sensors are not included in the standard aircraft instrumentation system and are very expensive. They are characterized by a relatively high noise levels and/or lags. This is why numerical differentiation of the angular rate measurements is usually used. Differentiation however amplifies the noise in the measurements and thus very accurate rate sensors should be used for the angular rates. A standard deviation of 0.001deg/s that is common to current high accuracy aerospace sensors permits this procedure [4]. In a case of a small UAV however, the standard deviation of the noise in low accuracy turn rate sensors is a number of orders higher (5 deg/s) and the differentiation of the measurements is not applicable directly. This problem can be solved by the use of higher accuracy sensors for the turn rates and the filtering of the measurements from the MMAE scheme. In the following figures we can see the effects of the filter to the turn rate measurements. It can be seen that the estimated values are much closer to the true values and the noise in the measurements is eliminated. As outlined in [16], the angular accelerations are obtained by smoothed numerical differentiation of the turn rates. An algorithm for effective and accurate calculation of the derivatives was used contained in the SIDPAC package (chapter 11 of [16]).

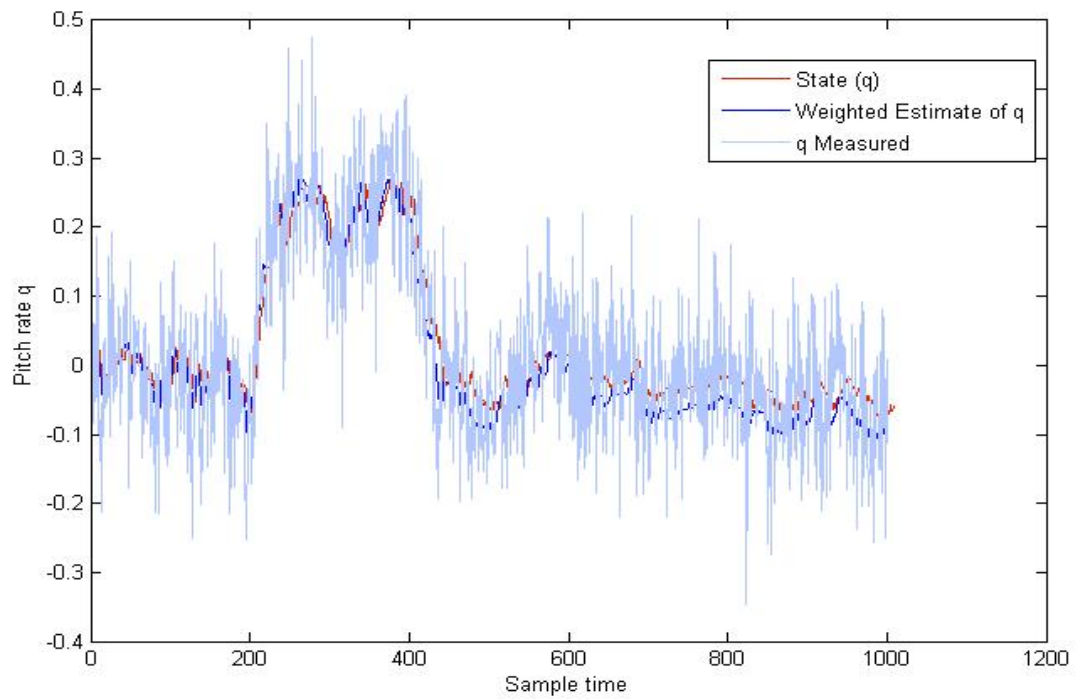


Figure 5.6. Pitch rate estimation compared with measured and true values

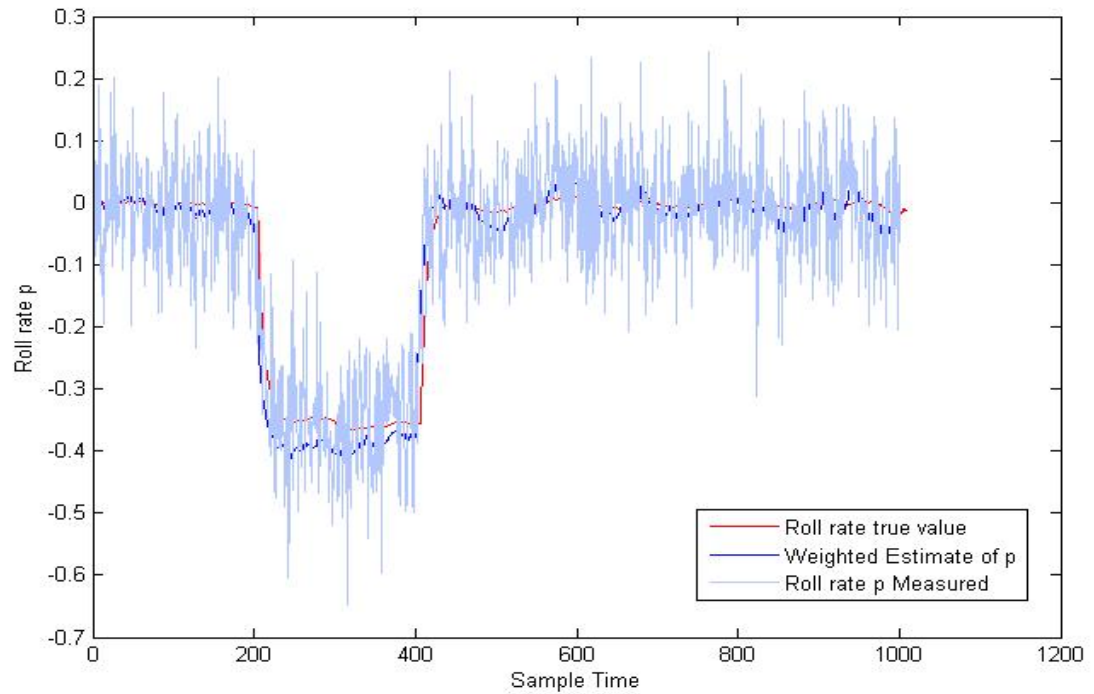


Figure 5.7. Roll rate estimation compared with measured and true values

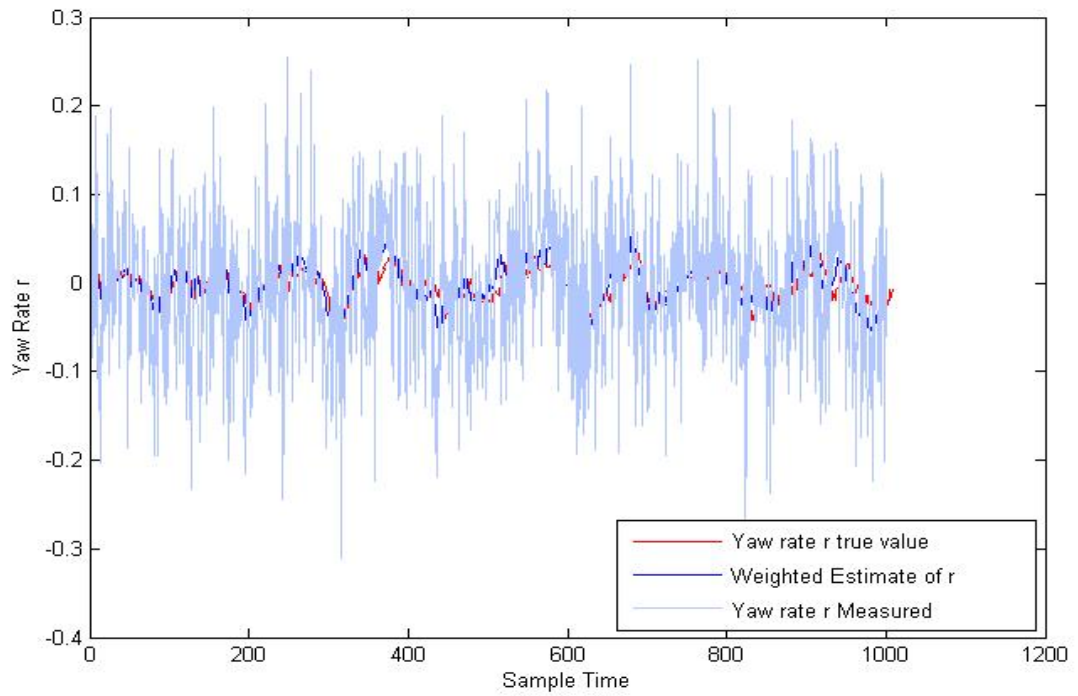


Figure 5.8. Yaw rate estimation compared with measured and true values

After the angular accelerations are computed, a linear regression technique, like the recursive least squares algorithm (RLS), can be used to estimate the non-dimensional aerodynamic coefficients. A separate linear regression problem is formulated and solved independently for the three equations of (5-18).

In order to identify a structural failure in the control surfaces a change in the computed coefficients with respect to their nominal values should be detected. There are two strategies to accomplish that. One is to rely on a weighting factor λ , in the recursive least squares procedure (Exponentially Weighted Least Squares) and the other is to incorporate a trigger for re-identification. The former has the disadvantage that during long periods of stationary flight with no control inputs, like cruise, the model is likely to become unstable due to the lack of significant excitations. A remedy to this problem could be the use of the Modified Sequential Least Squares (MSLS) algorithm [16], [82], which uses regularization terms and can guarantee that regardless of the amount of excitation the model parameters drift will be constrained. This is an important issue since cruise flight conditions constitute the largest part of a typical flight profile. This is why the trigger to re-identification was chosen.

Since a trigger is going to be used for the re-identification, a measure should be chosen that characterizes the quality of the

model. In [41], the authors describe a procedure to use the innovation (the difference between the model prediction and the actual behavior of the aircraft) as a measure for the quality of the model. However the absolute value of the innovation does not only depend on the model quality, but also on the noise in the input channels, which makes it unsuitable for quality determination. Instead, the whiteness of the innovation could be used as a quality measure, since a perfect model would have a residual comparable to the noise present in the input signals. The residual (innovation) of the estimated aerodynamic model can be calculated as follows:

$$\Delta(k) = z(k) - X(k)\hat{g}_{RLS}(k) \quad (5-19)$$

where $\Delta(k)$ the innovation at time step k , $z(k)$ the aircraft states measurements, $X(k)$ is the data (regressor) matrix and $\hat{g}_{RLS}(k)$ the vector of estimated parameters.

Several criteria for the whiteness of the innovation calculated by (5-19) can be used, like the autocorrelation criterion and the innovation average value. Also if the characteristic of the noise was known, the covariance matrix could be used as well [41]. The average of the innovation however is more general and simple computationally so it was chosen as a measure of model quality. The computation of the average value of the innovation $\bar{\Delta}(k)$ is performed by using the relation:

$$\bar{\Delta}(k) = \frac{1}{N_{av}} \sum_{i=0}^{N_{av}} \Delta(k-i) \quad (5-20)$$

The average is taken over a period of time (number of samples N_{av}) and this number is a design parameter that must be tuned carefully to avoid false alarms. Also for the triggering of the re-identification a threshold must be chosen to indicate the deviation of the innovation average from zero. This threshold should also be carefully chosen based on several flights with and without a failure. It should however be stressed that the trigger itself won't produce a false alarm but will just start the re-identification procedure. A value of 100 was selected (1s) for N_{av} to ensure that the innovation average will be close to zero (whiteness of the residuals). If a failure is present the whiteness criterion will detect that the model quality is poor and therefore a re-identification will be triggered.

Since there are three dimensionless moments and each has a separate innovation channel, the reconfiguration can be focused on the specific parameter or parameters that triggered the re-

identification. This prevents unnecessary destabilizing of the aircraft model parts that are used in the control system.

It is important to understand that for the fault detection and identification procedure, the absolute value of the estimates has less significance than its change compared to the initial value. The main advantage of the technique is the physical insight it provides since a good understanding of aircraft aerodynamics during failures can enhance the identification process. This understanding can be enriched by wind tunnel tests and the accumulated knowledge can be incorporated into the fault identification procedure by a fuzzy decision system which uses the values of the deviations of the parameters from their initial value to declare a structural failure. In this way the cross coupling introduced by the failures could be exploited in order to identify the failure.

In this thesis simple structural failure of the control surfaces is studied but more complicated failures could be handled as well. Structural failures at the control surfaces affect especially the control effectiveness parameters $(C_{(Y,l,n)\delta a}, C_{(X,Z,m)\delta e}, C_{(Y,l,n)\delta r})$ but other minor changes in aerodynamics $(C_{(Y,l,n)0/\beta/p/r}, C_{(X,Z,m)0/\alpha/q})$ are observed as well. Also cross couplings are likely to develop like $C_{l_{\delta e}}$ for example in case of elevator structural failure. It is important to realize that in order for the identification to work as expected the modeling of the forces and moments must be representative to the one after failure. The identification algorithm does not try to estimate separately all the control surfaces effectiveness coefficients but it takes under consideration the conventional way the surfaces are controlled:

$$\begin{aligned}\delta a1 &= -\delta a2 \\ \delta e1 &= \delta e2\end{aligned}\tag{5-21}$$

It also takes under consideration the different cross-coupling terms that are injected into the equations (5-18). Under normal conditions these terms are equal (or close) to zero. Here only cross-coupling terms related to the control surfaces were added but more thorough modeling based on wind tunnel tests can reveal that more terms should be added. In any case the identification procedure will not change.

The above procedure was again tested in open loop simulation of the non-linear aircraft model. The scenario assumes a straight and level flight maneuver with the left elevator losing one third of its total surface. The aerodynamic control surface coefficient of the left elevator for the roll and pitch moments is decreased by one third. The fault is injected at time 35s by

changing the aerodynamic coefficients of the model in the simulation. In particular, it is assumed that due to the failure there is a loss of effectiveness of the left elevator to contribute to the roll ($C_{l_{\delta e1}}$) and pitch moments $C_{m_{\delta e1}}$ (30%) accompanied by a change in aerodynamic coefficients for pitch rate C_{m_q} (20%) and the baseline pitch term C_{m_0} (10%). The results are shown in figure 5.9. The whiteness criterion is shown to be able to identify that a failure model is present at the pitch axis very quickly (less than 350ms) depending on the threshold chosen. Also the MMAE filter does not flag an alarm for this kind of failure (figure 5.10). Finally in the absence of a failure the whiteness criterion seems to be insensitive to the excitation of the control surfaces as shown in figure 5.11 for a significant excitation of all surfaces.

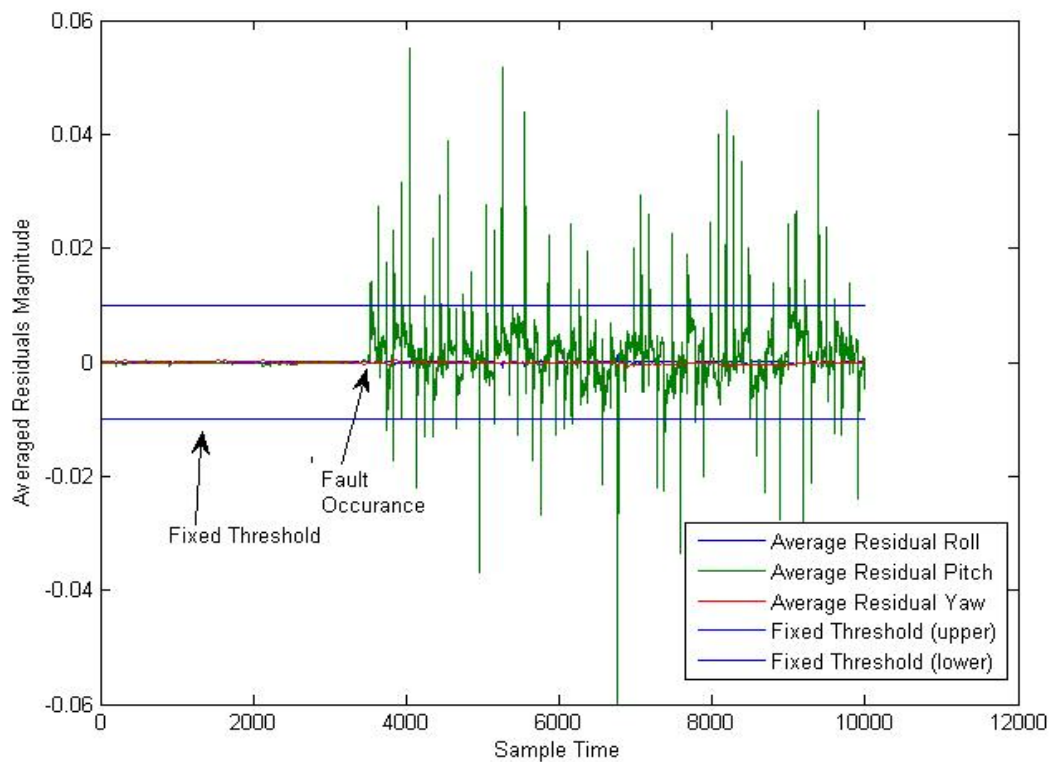


Figure 5.9. Average Residuals computed during an elevator structural failure at $t=35s$ (3500 time sample)

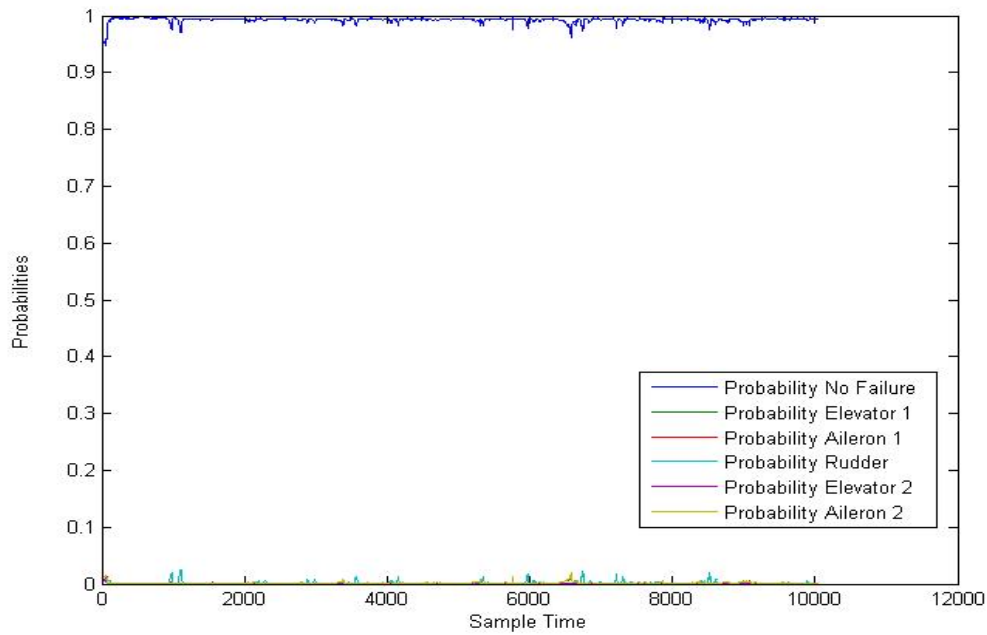


Figure 5.10. Failure Probabilities computed by the MMAE filter during the elevator structural failure at $t=35s$ (3500 time sample)

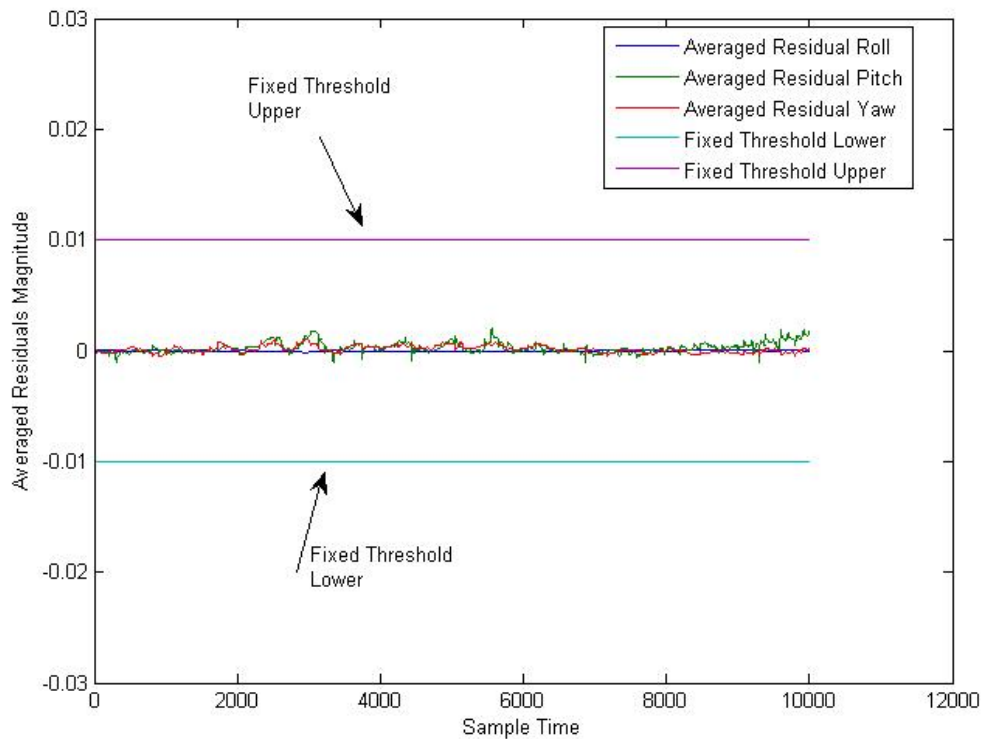


Figure 5.11. Average Residuals computed during no failure conditions with significant excitation of control surfaces

After the re-identification trigger is issued, the linear regression parameter estimation is initiated. The convergence of the parameters is fast, especially that of the elevator effectiveness

which converges only after three time samples (0.03 sec). All the parameters are efficiently and accurately estimated as shown in figure 5.12 for the case of pitch moment coefficients.

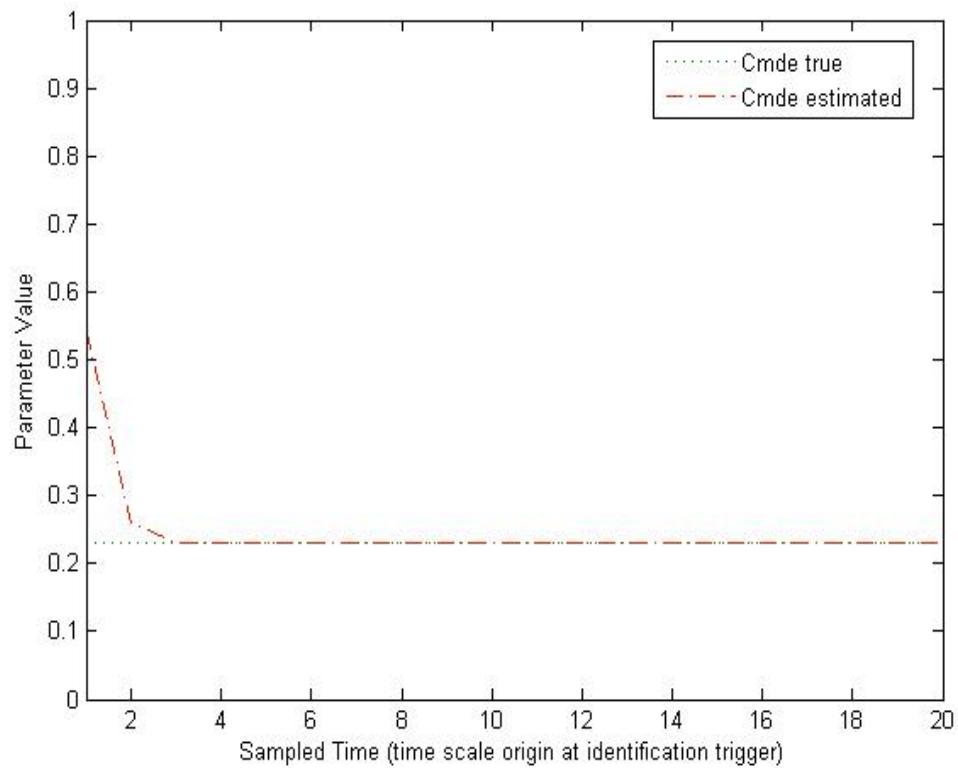


Figure 5.12 a. Estimation of elevator effectiveness coefficient

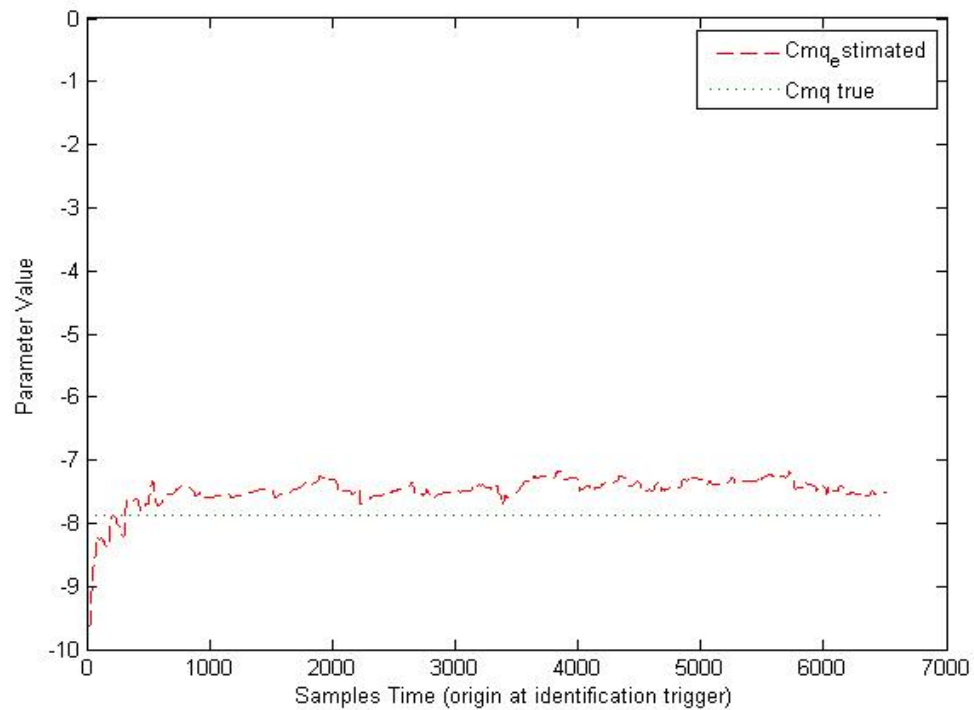


Figure 5.12 b. Estimation of pitch rate coefficient

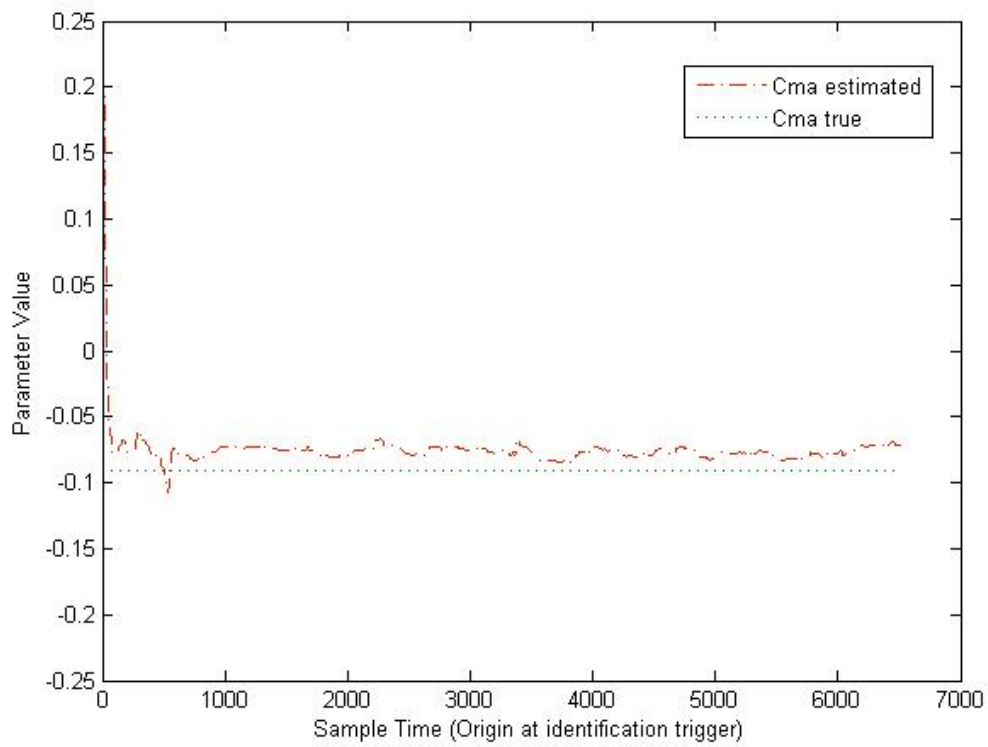


Figure 5.12 c. Estimation of angle of attack coefficient

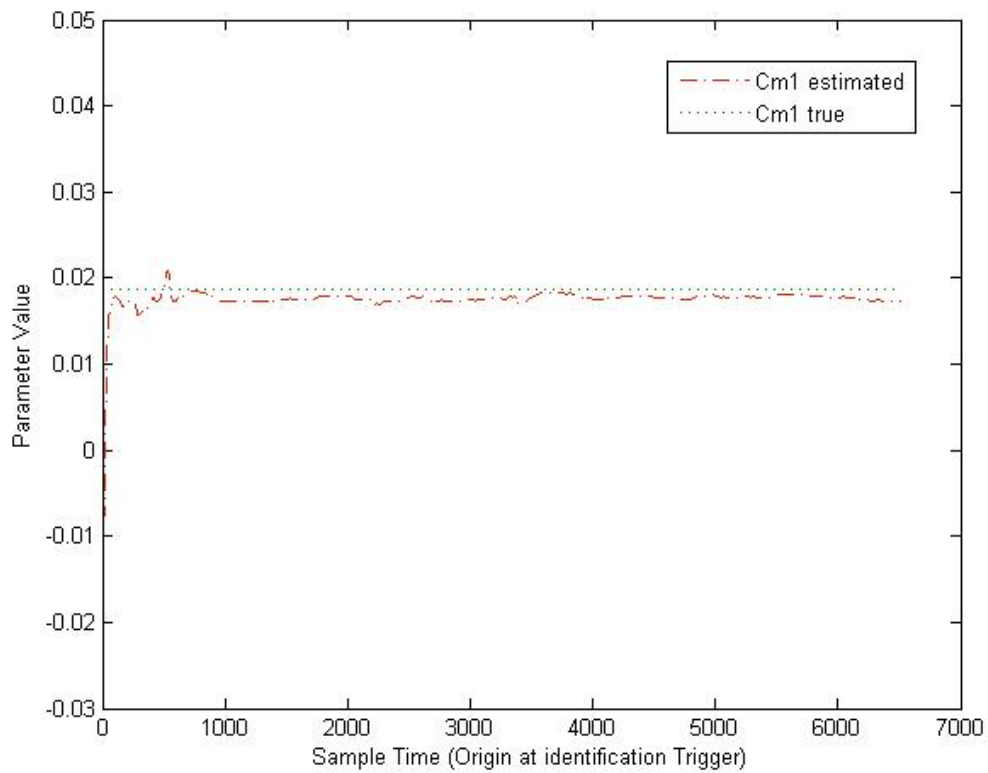


Figure 5.12 d. Estimation of bias coefficient

It is clear that the reduction in control effectiveness can be accurately estimated as a 30% difference in the pitching moment capability of the aircraft. However since no individual surface effectiveness was identified, we cannot determine which elevator is damaged. A way to identify the failure would be to estimate the elevator effectiveness to produce roll. However, this coefficient is very small and it is difficult to identify its contribution accurately and fast. A more efficient way to handle this problem is to perform a linear regression on rolling moment estimating the sign of the elevators contribution on rolling moment. If it is negative, the damaged surface is the right one and if positive the left one. In the following figure, we can compare the estimates of elevator contribution on rolling moment for a right (blue) and a left (red) elevator failure. It is clear that an identification decision can be reached within 3 time samples (0.03 sec) from the identification trigger.

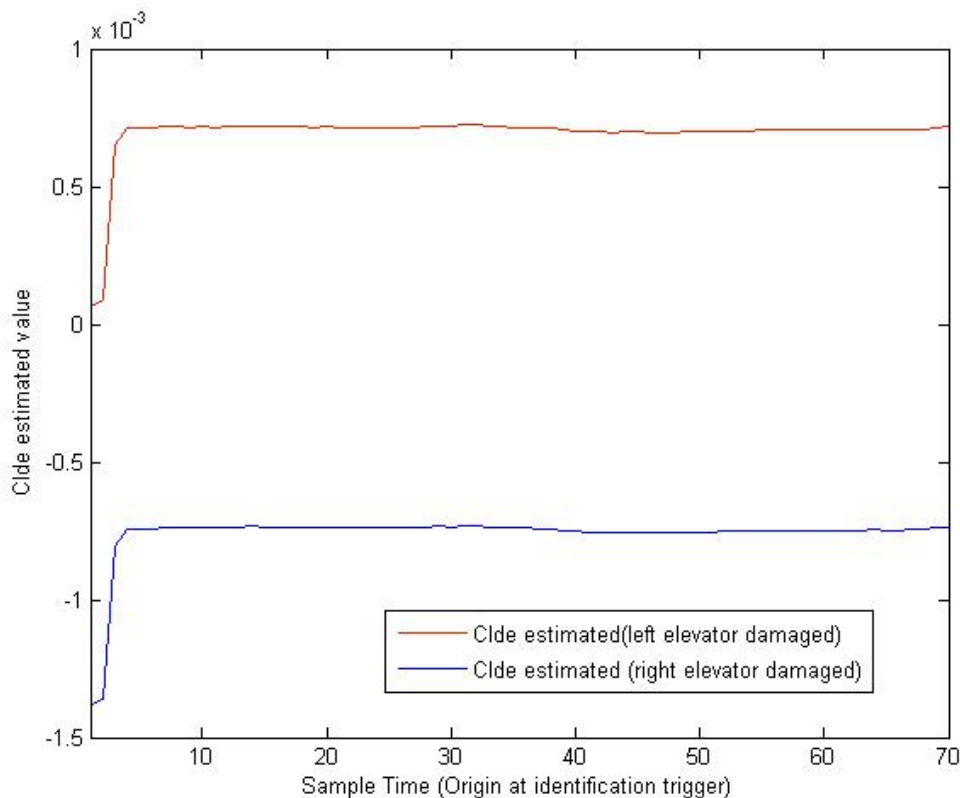


Figure 5.13 Estimation of elevator effectiveness for rolling moment (Cl_{de}) coefficient in the case of a left (red) and right (blue) elevator failures.

The above method can be used if damage is expected to occur on one surface at a time. In a different situation, in order to efficiently identify the failures on both surfaces, the regression problem for the rolling moment coefficient can be solved using the difference of the two elevators deflections. In order to do so, independent excitation of the elevators should be used. The resulting estimated coefficient expresses the combined capability of the elevators to produce rolling moment. Assuming that the failure has the same effect on pitching and rolling moment capability, the magnitude of the fault on both surfaces can be identified. However in this thesis we assume that the most common situation would be the damage of one control surface only.

The estimation of moment coefficients in real time is a stochastic procedure that can be affected by noise or insufficient excitation, the parameter estimation module can be used in conjunction with a fuzzy logic based inference system, which will be able to use the engineer's expertise for decision making purposes. Such a module was designed in this case to identify the elevator failures but could be easily extended to handle more complicated failures.

The fuzzy inference system has four inputs and two outputs. In order to identify an elevator failure, the estimated parameters $C_{m_{\delta_e}}$, C_{m_q} , C_{m_l} and $C_{l_{\delta_e}}$ are the inputs to the system. These are fuzzified taking under consideration the uncertainty in the estimation procedure. An example is shown in figure 5.14 for the parameter $C_{m_{\delta_e}}$. Figure 5.15 represents the fuzzification of the parameter $C_{l_{\delta_e}}$. The fuzzy rules have the form: "IF $C_{m_{\delta_e}}$ is S and $C_{l_{\delta_e}}$ is Negative then FaultEstimation is MF and FailedSurface is Left". This system can handle noise and can use wind tunnel test data to encode the failures symptoms on aerodynamic derivatives.

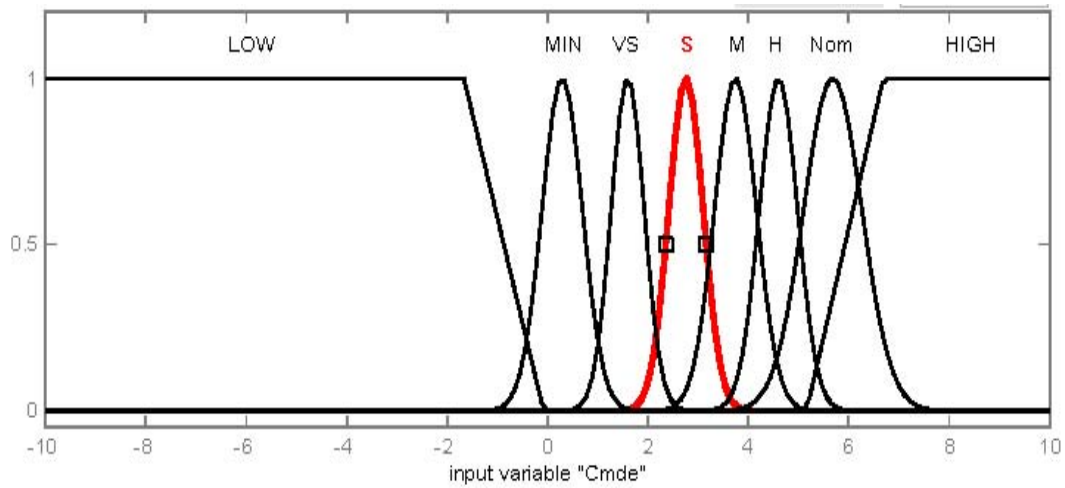


Figure 5.14 Fuzzification of input variable Cmde

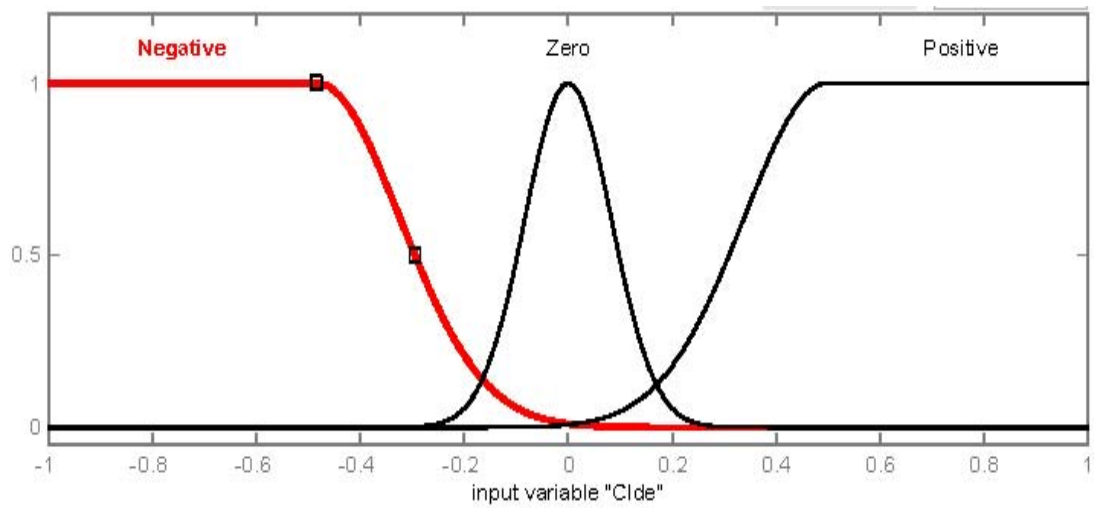


Figure 5.15 Fuzzification of input variable Clde

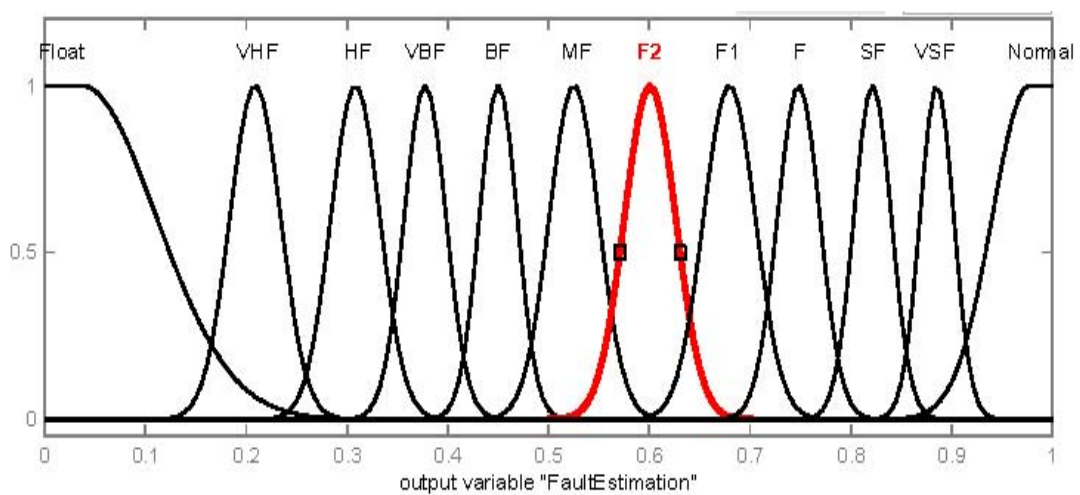


Figure 5.16 Fuzzification of output variable FaultEstimation

5.5 The need for a supervision system

It is evident that the model quality will be influenced not only by structural failures but also by any stuck or floating actuators. This “correlation” of the failures is caused by the lack of control surfaces position measurements. Thus any stuck actuator failure could be identified by the whiteness criterion and a re-identification could be triggered. This would be undesirable since the inaccuracy of the control surface measurements could cause faulty model identification. This problem can be handled by delaying the trigger for the identification an amount of time capable for the MMAE filter to identify the stuck surface failure. Unfortunately this means that the structural failures will have a longer delay of detection by at least 1 sec. In the figure 5.17 we can see how sensitive the whiteness criterion is in the case of a stuck failure. The detection is almost immediate (almost 350 ms). This sensitivity is similar to the sensitivity in structural failures. This means that the whiteness criterion can be used by a supervision system as an early fault detection mechanism, before an identification of the fault is possible.

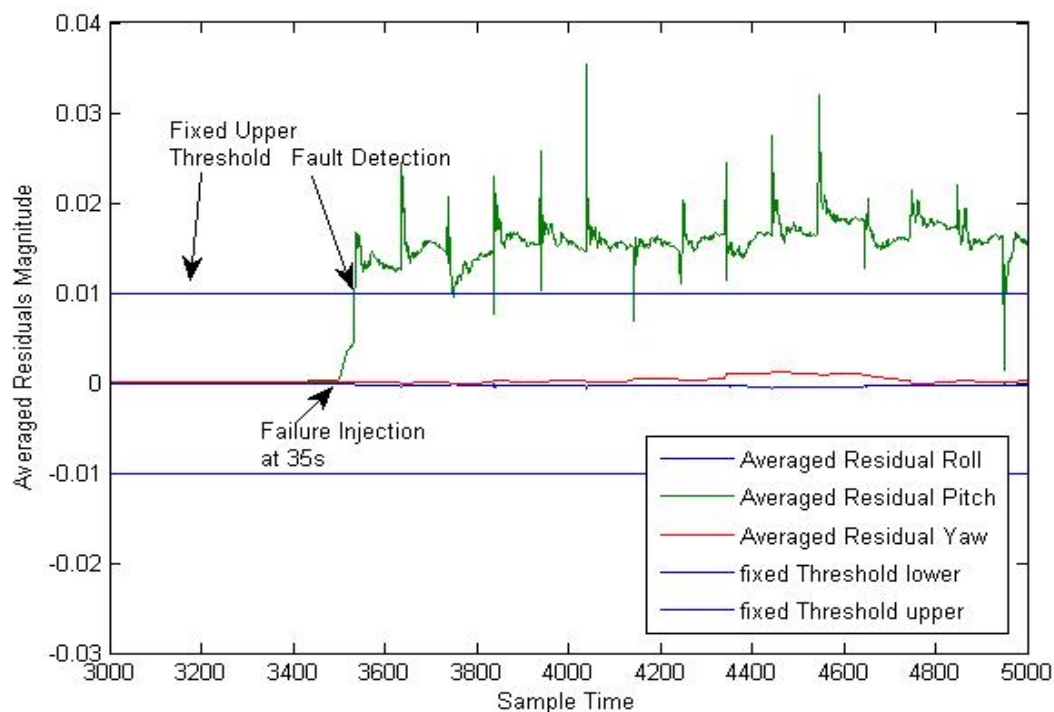


Figure 5.17. Average Residuals computed during a stuck elevator at last position failure conditions

The supervision system will provide command limiting to the navigation system to prevent instability for the period of time required by the FDI system to identify the failure and the control system to be reconfigured. If necessary it will also issue commands to make the identification of the fault easier. This is critical since minimum excitation is necessary both for the MMAE filter and the parameter estimation procedure. This supervision system is fed by signals from the MMAE filter and the whiteness criterion detection subsystem and is responsible for the fault information management.

After the early detection of the fault, the supervision system waits for 1 second for a fault isolation of a stuck or floating failure from the MMAE filter. In figure 5.18 we can see the response of the MMAE filter to the above stuck elevator failure. We can see that the correct control surface failure is identified in 1s. Also the magnitude of the fault is estimated correctly at the same time (Figure 5.19). The supervision system inhibits the trigger for re-identification until the MMAE filter identifies the fault.

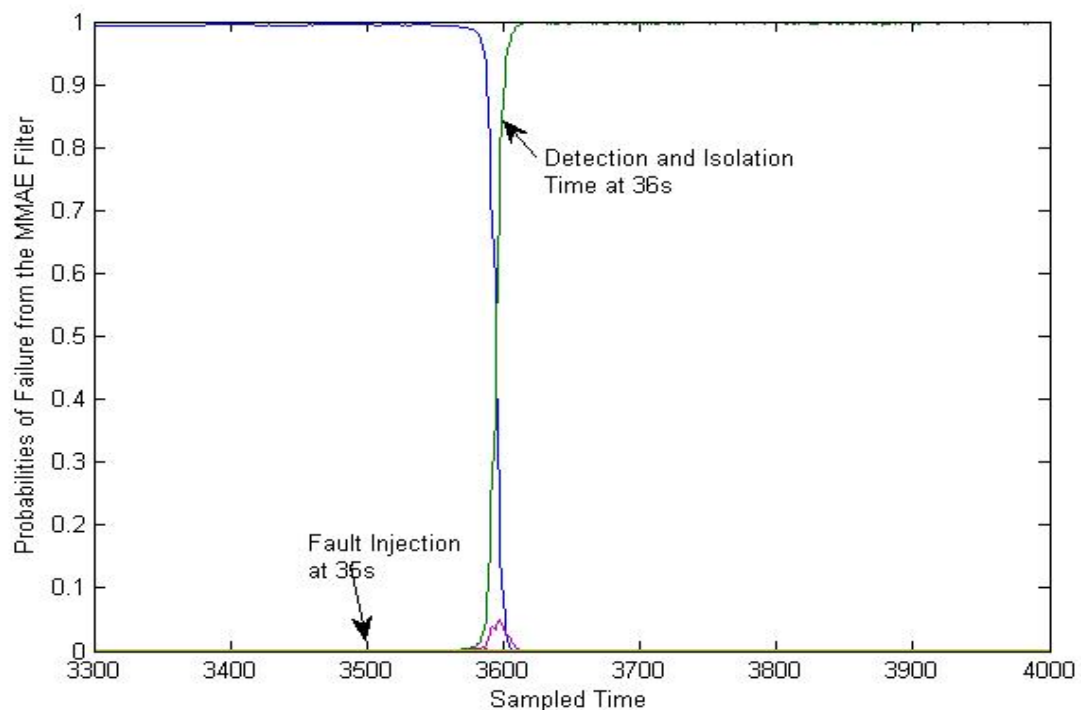


Figure 5.18. MMAE Filter probabilities computed during a stuck elevator at last position failure conditions

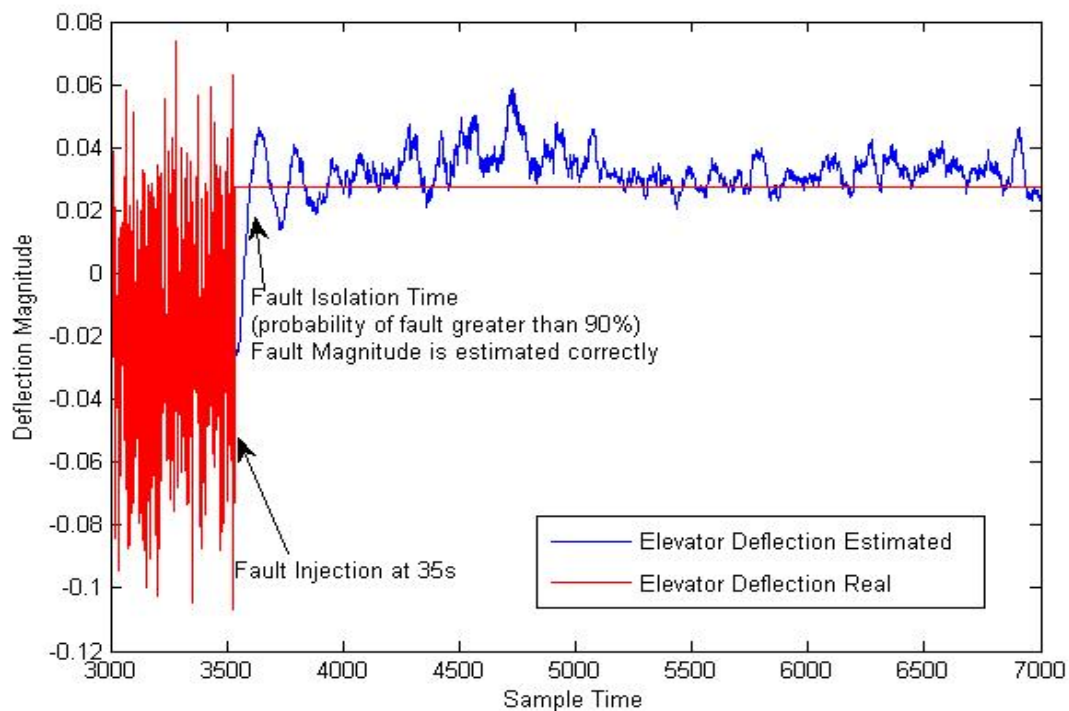


Figure 5.19. MMAE Filter left elevator deflection estimation against real deflection, computed during a stuck elevator at last position failure conditions

In the case of a structural failure of a control surface, the supervision system waits 1s limiting the commands from the pilot or the navigation system and issuing commands for identification. After 1s it checks the probabilities calculated by the MMAE filter. If these probabilities do not exceed 80%, a re-identification is triggered. If on the other hand there is a fault probability that does exceed 80%, the estimated deflection is checked and if the estimated value is constant, a stuck surface failure is declared.

On the other hand a loss of control effectiveness could be identified by the MMAE. This can happen if an EKF efficiently tracks an input in such a way that the structural damage can be “explained” by the failure scenario. In this case however, it is easy to reject the false alarm by efficient excitation and by comparing the commanded deflection with the estimated one. If the estimated deflection changes in a way similar to the command, then no stuck or floating actuator has happened. In order to handle these cases, a more complicated logic should be implemented. The simulations performed did not reveal such a case, that is why not such logic was implemented.

The overall fault detection system is shown in Figure 5.21. The supervision logic is presented in Figure 5.20. We can see that the system is hierarchical: The whiteness criterion is first flagged in

the case of a fault (within 350ms from failure occurrence). It is this signal that enables control surface excitation for parameter estimation (MMAE filter is benefited by that also). A null space injection policy can be used [83] or any other excitation method for active fault detection. The early detection of a fault can also be used so that extreme maneuvers can be inhibited by a governor. The supervision module initiates a 1 second timer and if a stuck surface failure is not detected by the MMAE filter, a re-identification is triggered. The system can identify a stuck surface deflection within one second after occurrence, along with a valid estimate of the stuck surface deflection. In the case of a damaged surface, due to the delayed trigger, a minimum of 1.03 seconds is required for failure detection and isolation. At the same time a valid estimate of the damaged surface effectiveness coefficient can be obtained. The other parameters of the model require slightly longer estimation times (i.e reliable C_{m_q} and C_{m_i} estimates are available at 2 seconds after failure since their estimation requires one second since identification trigger).

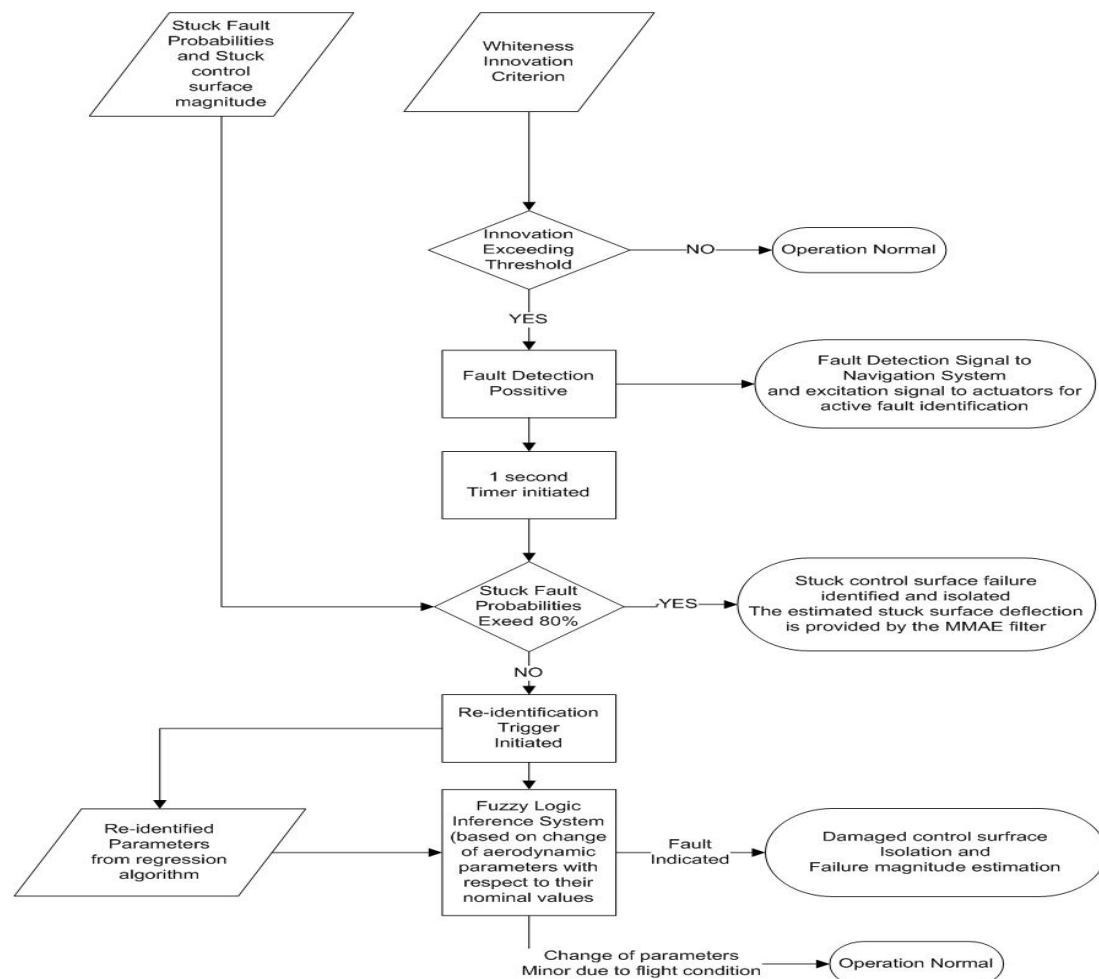


Figure 5.20. Fault detection logic implemented in the supervision module

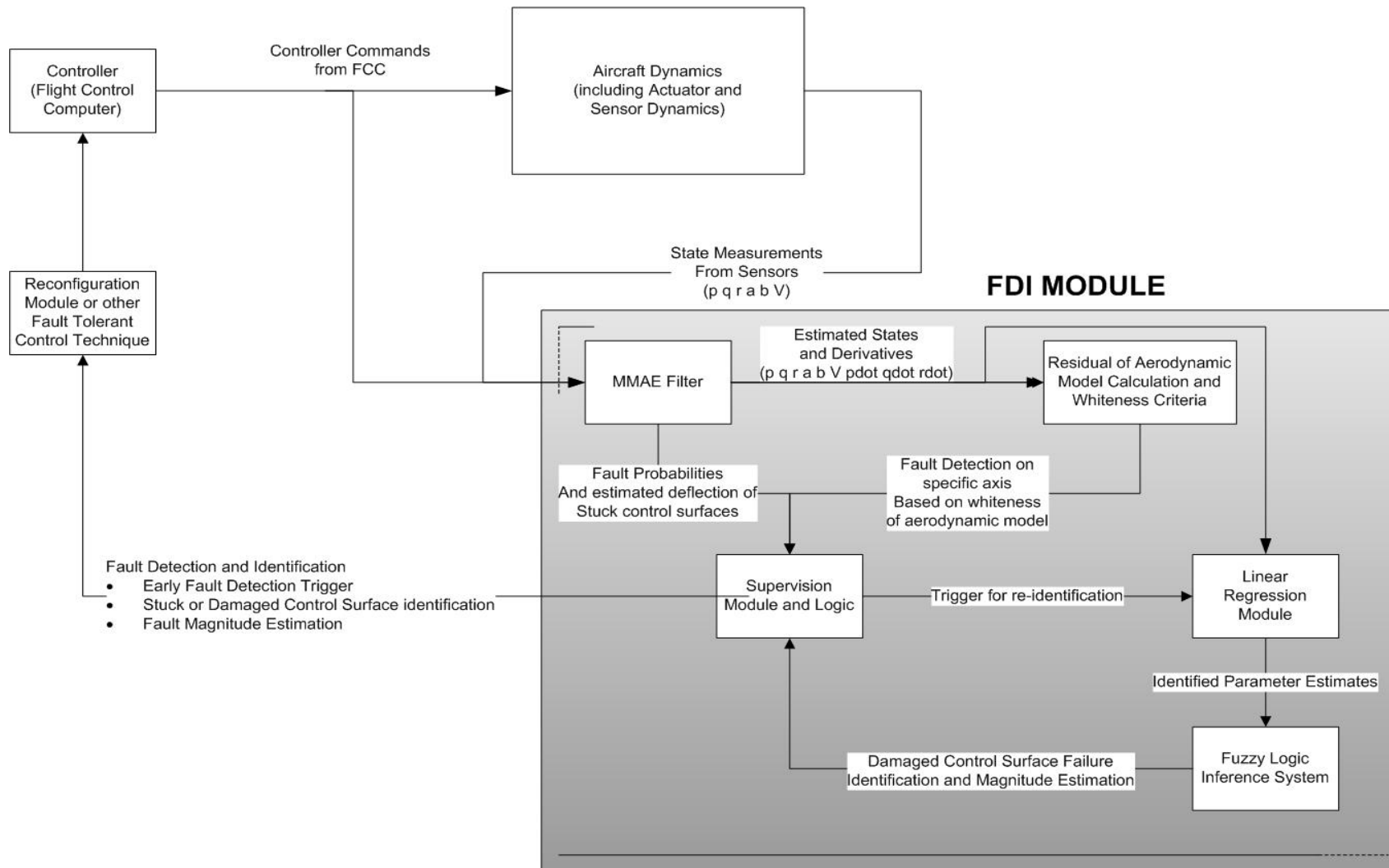


Figure 5.21. Overview of the proposed Fault Detection and Identification (FDI) module

Chapter 6 - Nonlinear Fault Tolerant Flight Control System

In this chapter, a fault tolerant flight control system is going to be developed for the ETH Zurich UAV. Although there are a lot of linear control system design methods in the literature, we will concentrate on a non-linear control method, since the introduction of faults along with the delay of detection, can lead to highly cross-coupling and non-linear behavior and can make the system deviate considerably from its equilibrium point before the reconfigurable law is implemented.

According to table 4.2, some methods suitable for non-linear FTC are Model Predictive Control (MPC), Sliding Mode Control (SMC), Adaptive Control and Feedback Linearization (FBL). The former has a lot of merits, especially the capability to physically include constraints (i.e. in actuator limits) and the simple reconfiguration procedure, since it only involves the replacement of the nominal model with the identified faulty one. However MPC is computationally demanding and its application is thus prohibited especially for non-linear systems. An interesting application was reported in [112], where MPC was combined with Feedback Linearization (FBL). Even in this case however, as reported by the authors, the complexity of the control law was quite high and a selection of a prediction horizon greater than two, led to a controller that could not be evaluated in real time onboard a large civil aircraft. This issue is even greater on a small UAV. What's more, the application of FBL requires very accurate knowledge of the system both before and after a failure, something that can be problematic. On the other hand SMC is a very promising technique, as reported in [113], [114]. However, it cannot explicitly handle complete actuator failures and special attention has to be given on the discontinuity in control law that can lead to chattering. Adaptive control is another option [115], however the adaptation of the parameters is an issue for fast changing dynamics like these experienced after a fault. Finally, FBL or Non-linear Dynamic Inversion (NDI) based adaptive controller was proposed in [106], with the parameters of the NDI module being identified by a parameter estimation procedure. However in this paper it was assumed that the estimates were accurate immediately after the fault's occurrence something really optimistic and no assessment was made for the transient period. What's more, reduced actuator effectiveness was only considered and no complete loss of an actuator and the control surface failures were treated as pairs.

Here, an NDI based technique will be used like the one in [106], since the FDI module designed in Chapter 5 has the capability to estimate the faulty characteristics of the system, namely the faulty deflection (for the stuck actuator case) and the parameters of the faulty system (for the control effectors structural failure). However, contrary to [106], we will treat the complete loss of actuator case as well as assess the controller's effectiveness during the period of parameter estimation.

6.1 Non-linear Dynamic Inversion based controller

6.1.1 Derivation of a Dynamic Inversion Controller

Let the plant be described by a set of non-linear affine in the input differential equations of the form

$$\begin{aligned}\dot{x}(t) &= f(x) + g(x)u, \\ y(t) &= h(x),\end{aligned}\tag{6-1}$$

where the state vector is $x(t) \in R^n$, the measurement vector is $y(t) \in R^m$ and the control input vector $u(t) \in R^p$. Differentiating the output y with respect to time we obtain:

$$\dot{y}(t) = \frac{\partial h}{\partial t} = \frac{\partial h}{\partial x} \frac{\partial x}{\partial t} = \frac{\partial h}{\partial x} f(x) + \frac{\partial h}{\partial x} g(x)u = F(x) + G(x)u \tag{6-2}$$

In order to force the output of the plant $y(t)$ follow a desired trajectory the signal of the desired output dynamics $\dot{y}_{des}(t)$ needs to be constructed. Then (6-2) can be easily used in order to find the appropriate control input:

$$u_c(t) = G^{-1}(x)(\dot{y}_{des}(t) - F(x)) \tag{6-3}$$

The control design task is then to build a suitable control signal for the desired output dynamics $\dot{y}_{des}(t)$. This signal can be constructed based on the error signal defined by:

$$\begin{aligned}e(t) &= y_c(t) - y_{meas}(t) \Rightarrow \dot{e}(t) = \dot{y}_c(t) - \dot{y}_{meas}(t) \dots \\ &\Rightarrow \dot{y}_{meas}(t) = \dot{y}_c(t) - \dot{e}(t)\end{aligned}\tag{6-4}$$

where $y_{meas}(t)$ is the measured output and $y_c(t)$ is the command signal. In the case of perfect tracking we should have $\dot{y}_{meas}(t) = \dot{y}_{des}(t)$ and $\dot{e}(t)$ should be driven to zero by selecting a controller K such that:

$$\dot{e}(t) = -Ke(t) \quad (6-5)$$

From (6-5) and (6-4) we get that:

$$\dot{y}_{des}(t) = \dot{y}_c(t) + Ke(t) \quad (6-6)$$

Usually however, the desired dynamics signal is constructed by selecting a suitable PID controller. This technique will be used in this thesis. The overall structure of the NDI scheme is shown in the following figure.

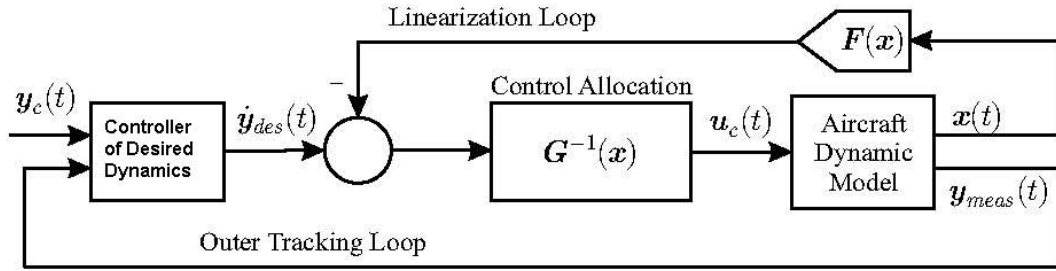


Figure 6.1. Non-linear Dynamic Inversion general scheme

6.1.2 NDI applied on a UAV

It is obvious that in order to achieve tracking control for some parameters, the availability of at least as many control inputs are required. In the case of our fixed wing UAV, there are three aerodynamic inputs and a thrust force. The three inputs are the ailerons, the elevators and the rudder. This means that three quantities can be controlled. However, in the case of any fixed wing aircraft, the time scale separation (TSS) technique can be applied. The technique is based on the different levels of control effectiveness that can be seen as the effect on a controlled parameter due to unity change of the controlling parameter. Based on the control effectiveness, we can make a distinction between slow dynamics and fast dynamics. Slow dynamics means that the control effectiveness of a certain parameter is low. Fast dynamics means that the control effectiveness is high. Time scale separation means that we can split the fast and the slow dynamics. The fast dynamics can then be seen as the inner loop, while the slow dynamics make the outer loop. For every part, dynamic inversion is applied separately.

A typical distinction is that of attitude angles (slow dynamics) and rates (fast dynamics). The inner loop consists then of the three

angle rates (pitch, roll and yaw rate), having as output the desired deflection of the control surfaces. The outer loop, consists of the aerodynamic angles (ϕ roll, θ pitch and β sideslip) having as output a reference to be tracked by the inner loop. The design can be done in six steps:

1. We start with the reference flight angles (ϕ roll, θ pitch and β sideslip) and we derive the reference flight angle derivatives ($\dot{\phi}, \dot{\theta}$ and $\dot{\beta}$) using a PID controller like:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\beta} \end{bmatrix}_{ref} = k_{P_i} \left(\begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{ref} - \begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{act} \right) + k_{I_i} \int \left(\begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{ref} - \begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{act} \right) + k_{D_i} \frac{d}{dt} \left(\begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{ref} - \begin{bmatrix} \phi \\ \theta \\ \beta \end{bmatrix}_{act} \right) \quad (6-7)$$

2. From the derivatives, the rotational rates of the UAV should be derived. We can use the following equations relating the time derivatives of the aerodynamic angles with angle rates [16]:

$$\dot{\phi} = \frac{d\phi}{dt} = p + (q \sin \phi + r \cos \phi) \tan \theta \Rightarrow$$

$$\dot{\phi} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (6-8)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi = \begin{bmatrix} 0 & \cos \phi & -\sin \phi \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (6-9)$$

and

$$\beta = \arcsin \frac{v}{V} \Rightarrow$$

$$\dot{\beta} = \frac{1}{\sqrt{V^2 - v^2}} \dot{v}$$

$$= \frac{1}{\sqrt{V^2 - v^2}} [A_y + g \cos \theta \sin \phi + pw - ru] \quad (6-10)$$

$$= \frac{1}{\sqrt{V^2 - v^2}} [A_y + g \cos \theta \sin \phi] + \begin{bmatrix} \frac{w}{\sqrt{V^2 - v^2}} & 0 & \frac{-u}{\sqrt{V^2 - v^2}} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Where A_y is the lateral specific force. The rotational rates of the UAV can be derived using the matrix equation:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{ref} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ \frac{w}{\sqrt{V^2-v^2}} & 0 & \frac{-u}{\sqrt{V^2-v^2}} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\beta} \end{bmatrix}_{ref} - \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\sqrt{V^2-v^2}}[A_y + g \cos\theta \sin\phi] \end{bmatrix} \right\} \quad (6-11)$$

3. Having the desired rotational rates, we can find the desired rotational accelerations of the aircraft using a PID controller:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}_{ref} = k_{p_2} \left(\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{ref} - \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{act} \right) + k_{I_2} \int \left(\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{ref} - \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{act} \right) + k_{D_2} \frac{d}{dt} \left(\begin{bmatrix} p \\ q \\ r \end{bmatrix}_{ref} - \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{act} \right) \quad (6-12)$$

4. To find the required moments we can use the equation:

$$\begin{bmatrix} L \\ M \\ N \end{bmatrix}_{req} = I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}_{ref} + \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{act} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}_{act} \quad (6-13)$$

5. The required moments should then be normalized to calculate the moment coefficients:

$$C_{l_{req}} = \frac{L_{req}}{\frac{1}{2}\rho V^2 S b}, C_{m_{req}} = \frac{M_{req}}{\frac{1}{2}\rho V^2 S \bar{c}}, C_{n_{req}} = \frac{N_{req}}{\frac{1}{2}\rho V^2 S b} \quad (6-14)$$

And the contribution of the states should be subtracted in order to calculate the required contribution of the control deflections. This step depends of course on the modeling of the moment coefficients which is unique for every UAV:

$$\begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} = \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{req} - \begin{bmatrix} C_{l_{states}} \\ C_{m_{states}} \\ C_{n_{states}} \end{bmatrix}_{act} \quad (6-15)$$

6. The final step is to calculate the required deflections of the control surfaces (control allocation step). In a typical UAV with

conventional movement of the control surfaces, the equation used in this step is:

$$\begin{bmatrix} \delta_e \\ \delta_a \\ \delta_r \end{bmatrix}_{req} = \begin{bmatrix} C_{l_{\delta_e}} & C_{l_{\delta_a}} & C_{l_{\delta_r}} \\ C_{m_{\delta_e}} & C_{m_{\delta_a}} & C_{m_{\delta_r}} \\ C_{n_{\delta_e}} & C_{n_{\delta_a}} & C_{n_{\delta_r}} \end{bmatrix}^{-1} \begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} \quad (6-16)$$

The NDI method described above has a big known downside: To apply it, the model of the system has to be known quite accurately. However, the modeling of a UAV can contain non-modeled terms and what's more, during a failure the estimated parameters may not be accurate enough. A possible solution to the above problems is the use of an incremental form of NDI (INDI). This technique does not give the required input to control the system but the required change in the input. The design technique could be similar to the steps 1-4 of the NDI technique, however there would be changes in the two last steps:

- In step 5 of the technique, the measured states and the control deflections could be used to compute the whole aerodynamic coefficients:

$$\begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} = \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{req} - \begin{bmatrix} C_{l_{computed}} \\ C_{m_{computed}} \\ C_{n_{computed}} \end{bmatrix}_{act} \quad (6-17)$$

- In the final step, the required change in control surface deflections could be computed as:

$$\begin{bmatrix} \Delta\delta_e \\ \Delta\delta_a \\ \Delta\delta_r \end{bmatrix}_{req} = \begin{bmatrix} C_{l_{\delta_e}} & C_{l_{\delta_a}} & C_{l_{\delta_r}} \\ C_{m_{\delta_e}} & C_{m_{\delta_a}} & C_{m_{\delta_r}} \\ C_{n_{\delta_e}} & C_{n_{\delta_a}} & C_{n_{\delta_r}} \end{bmatrix}^{-1} \begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} \quad (6-18)$$

One could see that the difference between the required and the computed coefficients (if those coefficients were exact), would be the same as the difference between the required and actual angular rate derivatives. The INDI technique thus replaces equation (6-17 and 6-18) with equation 6-19:

$$\begin{bmatrix} \Delta\delta_e \\ \Delta\delta_a \\ \Delta\delta_r \end{bmatrix}_{req} = \begin{bmatrix} C_{l_{\delta_e}} & C_{l_{\delta_a}} & C_{l_{\delta_r}} \\ C_{m_{\delta_e}} & C_{m_{\delta_a}} & C_{m_{\delta_r}} \\ C_{n_{\delta_e}} & C_{n_{\delta_a}} & C_{n_{\delta_r}} \end{bmatrix}^{-1} \left(\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}_{des} - \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}_{meas} \right) \quad (6-19)$$

In order to apply the INDI method we need to measure the angular rate derivatives. As we explained in section 5.4 the angular rate derivatives are needed for parameter estimation purposes as well, and they can be computed by a differentiation method from the smoothed angular rate estimates provided by the MMAE filter. The INDI is much more robust to parameter uncertainties (especially uncertainties in the parameters related to the states) and un-modeled effects and the price to pay for this is the need to use the angular derivatives as measurements. The method does depend strongly however to the accuracy of control surface effectiveness parameters as well as the matrix of inertia.

6.1.3 Fault Tolerance in an NDI control system

In order to make the above control system fault tolerant, we need to update the controller after the failure. There are different ways to do so, depending on the specific failure experienced by the system:

- (a) In the case of a damaged control surface, an adaptive NDI could be used. The FDI system should provide the new estimates of the coefficients in order to apply the new control law. The coefficients should enter the model after the detection of the failure and the trigger of the re-identification process. Both equations (6-16) and (6-15) should be updated. The procedure was proposed in [106] but accurate knowledge of the coefficients after failure was assumed. Also, no evaluation was conducted for the transient period before the convergence of the estimated parameters.
- (b) In the case of a stuck surface on the other hand, we need to change the last step of the control law only, the control allocation step. There are a lot of algorithms for that, ranging from optimal allocation to simple and computationally efficient methods. It seems that the blending of control allocation and NDI is a straight forward approach.

6.1.3.1 Damaged Control Surface Case

In this section the capability of the controller to handle damaged control surface failures will be evaluated. The failure considered is the 30% reduction of the right elevator control surface, considered during the FDI design.

The reconfiguration will be evaluated based on the inner NDI loop, so the assessment of the control system will be based on its capability to track rate commands (pitch rate commands for the failure considered). Figure 6.2 below shows the capability of the simple NDI technique to handle the elevator failure.

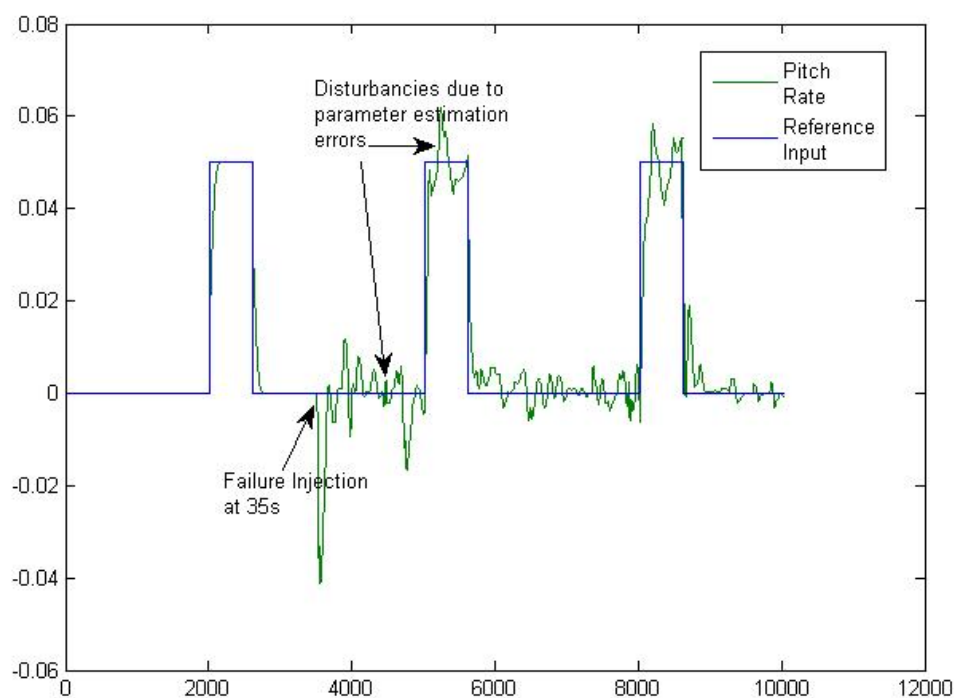


Figure 6.2. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure.

Although the controller can follow the reference input even after the failure, its performance is poor as can be observed by the disturbances (spikes) present. The poor performance of the NDI method to compensate for the failure is caused by the poor accuracy of the identified parameters. Indeed, the low excitation and the closed loop identification procedure lead to inaccuracies in all parameters, including the control effectiveness parameters. However, this inaccuracy is far greater for the state dependent parameters like C_{ma} (figure 6-3).

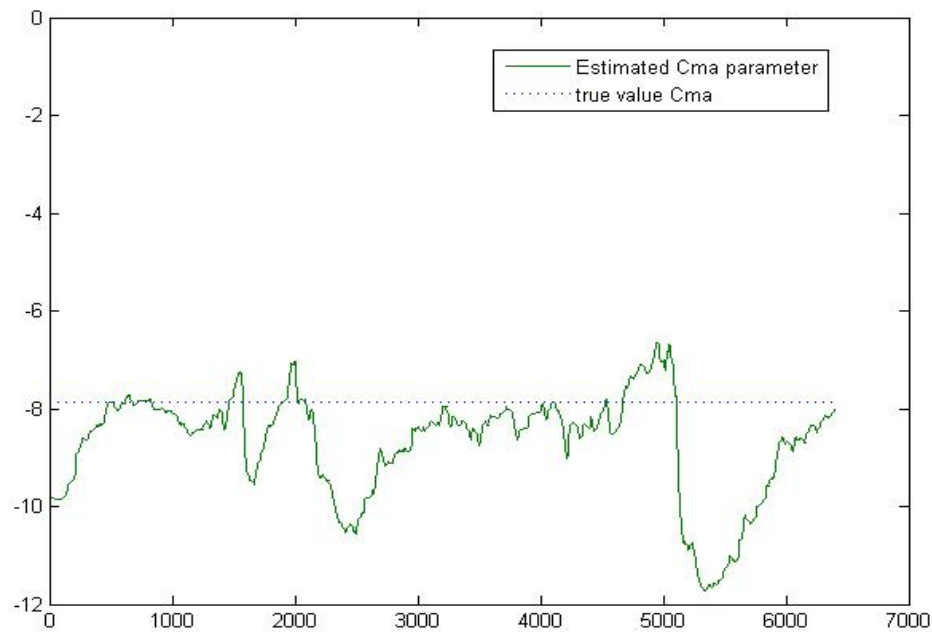


Figure 6.3. Estimated versus true Cma parameter (closed loop identification)

Contrary to simple NDI, the INDI technique is much more robust. As can be seen in figure 6-4, the controller is capable of tracking pitch rate step commands even after the failure. There are however still some disturbances in pitch rate caused by the inaccurate parameter estimation procedure (related to the damaged elevator effectiveness).

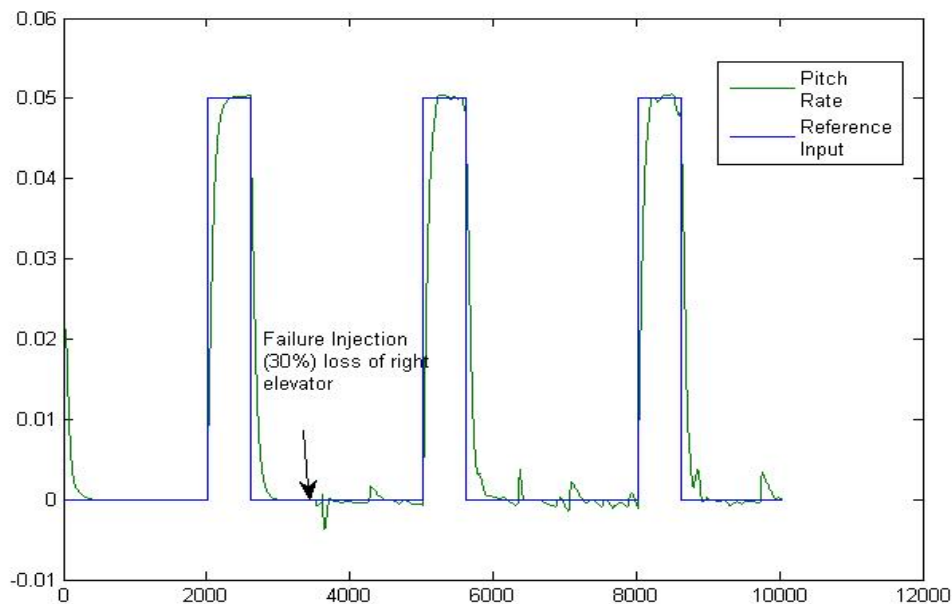


Figure 6.4. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure.

A possible solution to the above problem, is the incorporation of an elevator doublet initiated by the supervision subsystem. The excitation enhances the parameter estimation effectiveness and the capability of the controller for pitch rate command tracking is enhanced also (figure 6-5). However, as the identification procedure is continuous after the failure, the lack of excitation causes the deterioration of the estimates. This is especially true for the parameters affecting the states. Although the INDI technique can handle this issue, a stopping rule for the identification procedure can be incorporated also after the post-failure parameters are identified.

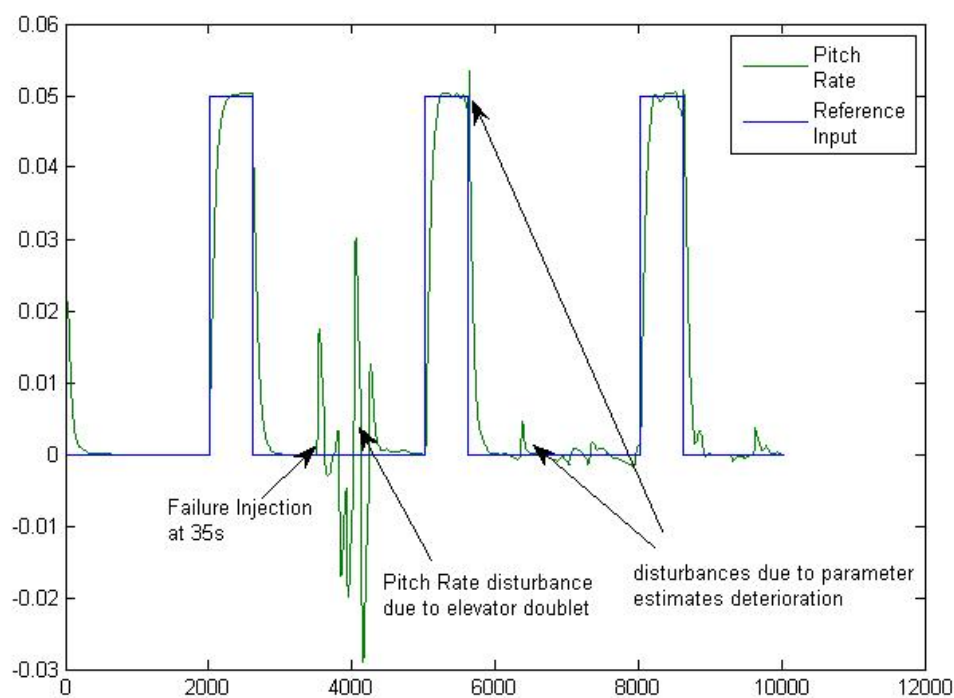


Figure 6.5. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure with elevator doublet applied for identification purposes after the failure.

Finally the INDI controller (with elevator doublet excitation command after the failure) is evaluated for the case of small to moderate wind conditions and different reference inputs. The following figures (figure 6-6 to 6-8) display the results, showing that the controller is capable to compensate for the elevator fault and track the reference input perfectly with no steady state error. The other angular rates (roll and yaw rates) are kept to zero.

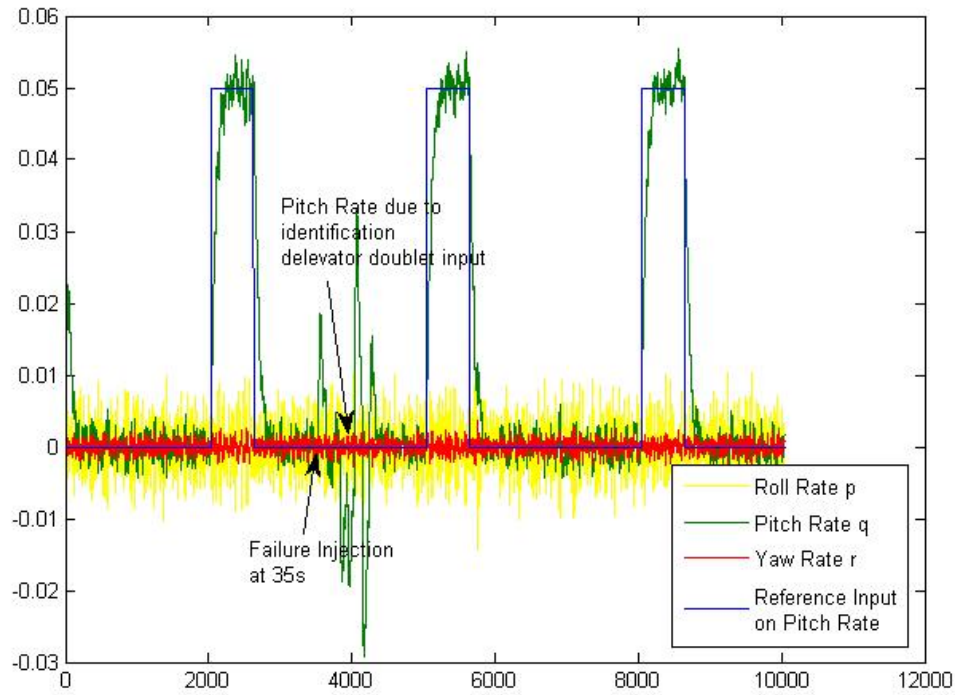


Figure 6.6. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure

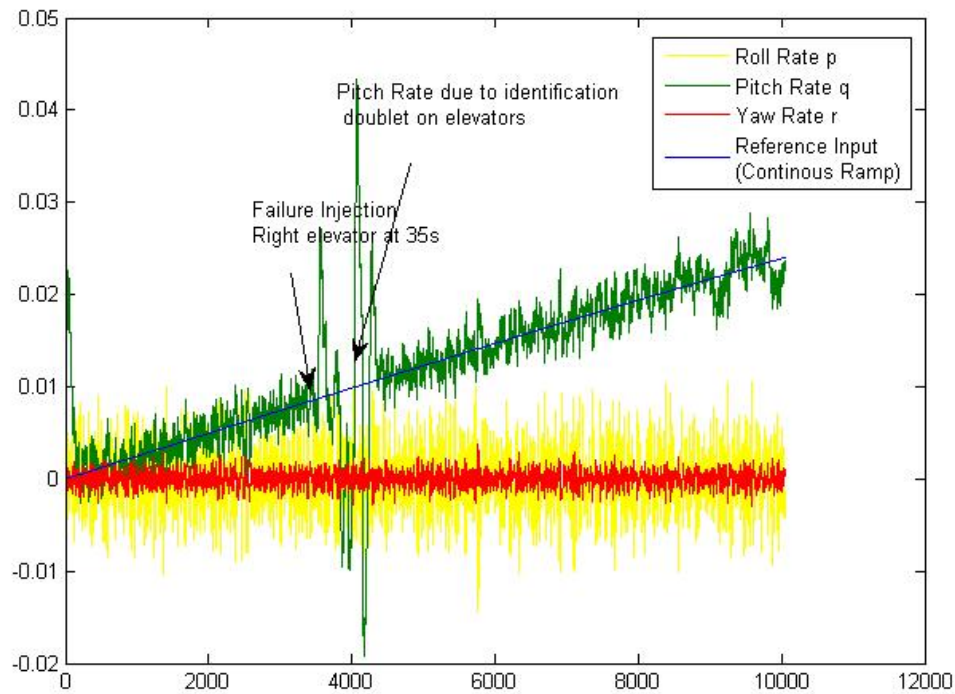


Figure 6.7. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure

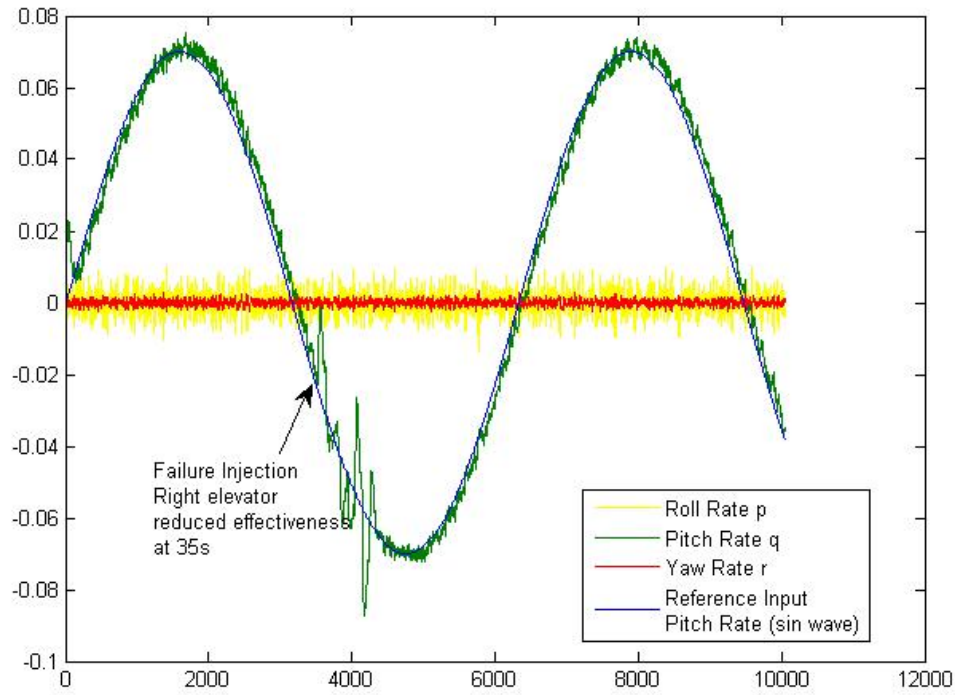


Figure 6.8. Adaptive INDI based fault tolerant controller capability to track pitch rate step commands after right elevator failure

6.1.3.2 Stuck Control Surface Case

In this section the capability of the controller to compensate stuck control surface failures will be evaluated. In fact the structure of the NDI controller will remain the same with a modification in the final (control allocation step). There are a lot of methods to do so like the pseudo-inverse method or optimal control allocation [105]. Also, a simple control allocation algorithm was proposed in [10], where in the case of a stuck control surface and depending on the failure mode, simple rules are used to drive the other actuators.

In this thesis, the information provided by the FDI module, in particular the failure detection signal and the estimated deflection of the stuck control surface are used by the controller to calculate the necessary deflections of the other actuators. The stuck control surface deflection will be treated as a state in equation (6-17) while the commanded deflections will be calculated similarly to equation (6-16) (simple NDI) or (6-18) (Incremental NDI) by using the healthy control surface only:

$$\begin{bmatrix} \delta_e \\ \delta_{a1} \\ \delta_r \end{bmatrix}_{req} = \begin{bmatrix} C_{l_{\delta_e}} & C_{l_{\delta_{a1}}} & C_{l_{\delta_r}} \\ C_{m_{\delta_e}} & C_{m_{\delta_{a1}}} & C_{m_{\delta_r}} \\ C_{n_{\delta_e}} & C_{n_{\delta_{a1}}} & C_{n_{\delta_r}} \end{bmatrix}^{-1} \begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} \quad (6-19)$$

$$\begin{bmatrix} \Delta\delta_e \\ \Delta\delta_{a1} \\ \Delta\delta_r \end{bmatrix}_{req} = \begin{bmatrix} C_{l_{\delta_e}} & C_{l_{\delta_{a1}}} & C_{l_{\delta_r}} \\ C_{m_{\delta_e}} & C_{m_{\delta_{a1}}} & C_{m_{\delta_r}} \\ C_{n_{\delta_e}} & C_{n_{\delta_{a1}}} & C_{n_{\delta_r}} \end{bmatrix}^{-1} \begin{bmatrix} C_{l_{delta}} \\ C_{m_{delta}} \\ C_{n_{delta}} \end{bmatrix}_{req} \quad (6-20)$$

The efficiency of the control system will be evaluated based on the right elevator stuck control surface failure used for the FDI design. The failure is assumed to be injected at 35 seconds of flight and the right elevator gets stuck at 0.25 degrees from trim. Figure 6.9 shows the failure detection from the MMAE filter of the FDI module. It can be seen that the detection and isolation of the failure is fast (around 1s) and the estimation of the fault deflection has converged at the same time (Figure 6.10).

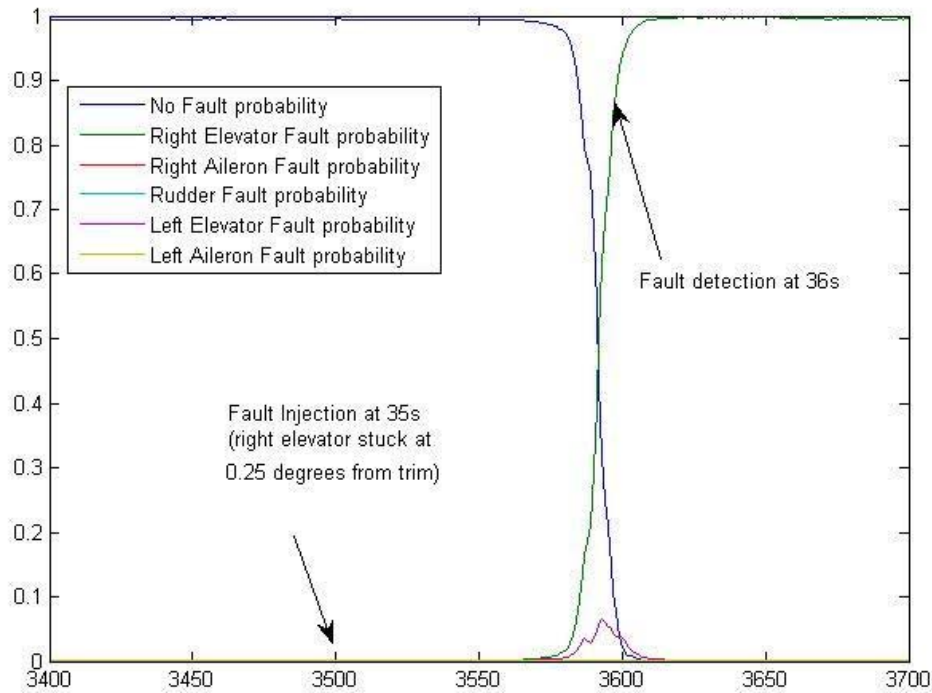


Figure 6.9. Fault probabilities calculated from the MMAE algorithm of the FDI subsystem (closed loop). The detection and isolation is achieved in 1 second.

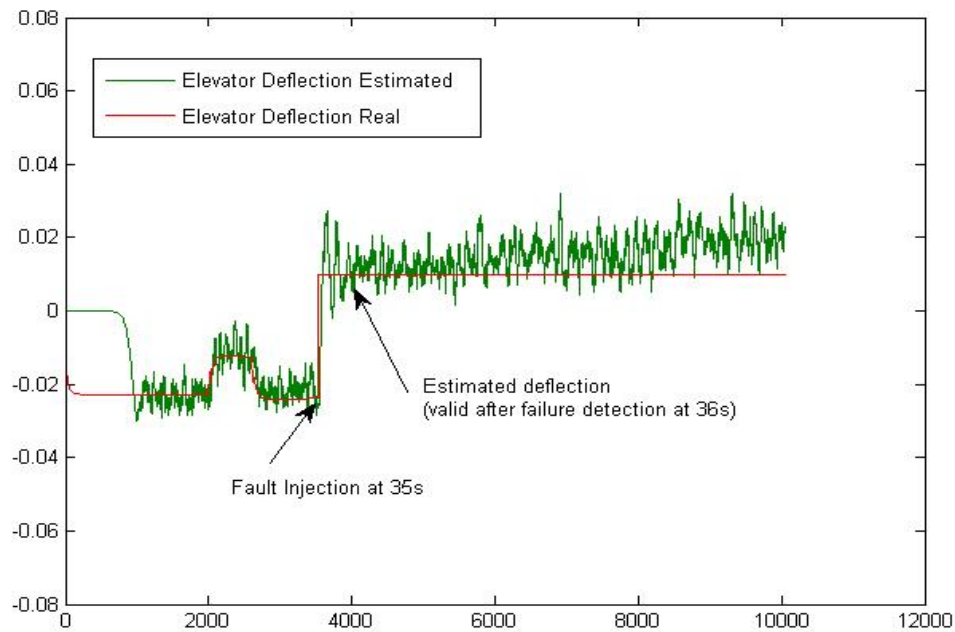


Figure 6.10. Estimated versus real right elevator deflection. The estimated values are valid only after the failure injection.

The controller is able to compensate for the failure, although due to low excitation, the estimation of the stuck deflection does exhibit fluctuations that cause the same disturbances in the controlled angular rates after the failure (Figure 6-11).

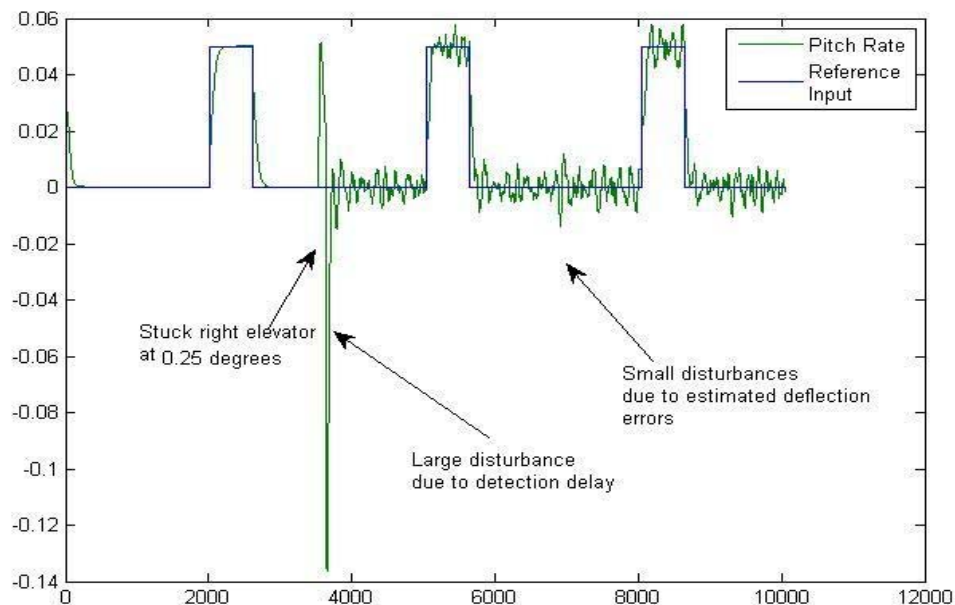


Figure 6.11. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 0.25 deg from trim.

One could note that there is a large spike at fault occurrence caused by the fault detection delay of one second. The fault however cannot destabilize the system. Similar results were obtained by exciting the system at fault occurrence (Figure 6-12). The UAV is kept under control and is capable to suppress the transient response and follow the reference input accurately.

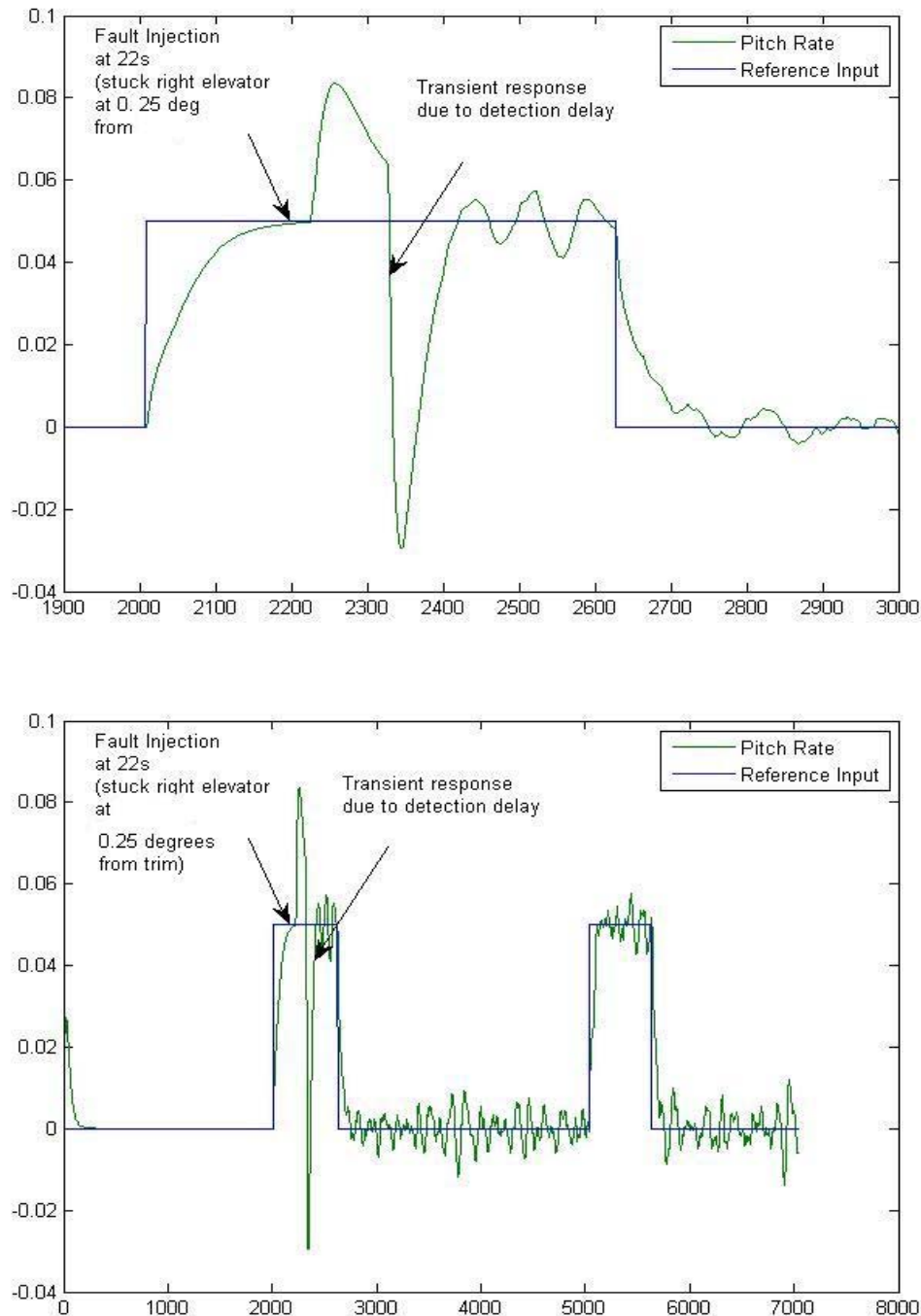


Figure 6.12. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 0.25 deg from trim. The failure is injected during excitation of the system.

In the case of a more severe failure, the transient response will be worst, however, the detection delay will also be reduced. In the following simulations a more severe failure was addressed (the right elevator gets stuck at 5 degrees from trim (almost one fifth of the maximum deflection of the control surface)). The detection delay is reduced to less than 0.5 seconds as it can be seen from figure 6-13.

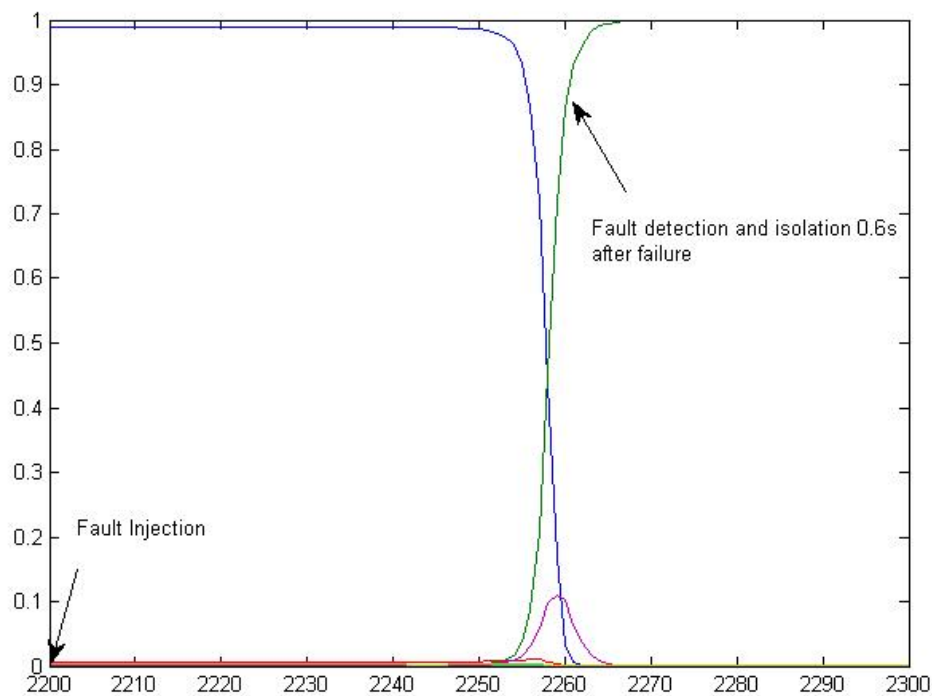


Figure 6.13. Fault probabilities calculated from the MMAE algorithm of the FDI subsystem (closed loop). The detection and isolation is achieved in 0.6 second.

The controller is able to track the reference input with no steady state error after the initial transient response. The pitch rate achieved can be seen in figure 6-14. The controller can compensate for the fault by commanding the left elevator to the opposite direction as shown in figure 6-15. The commanded deflection does not exceed the maximum limits even during the transient response period.

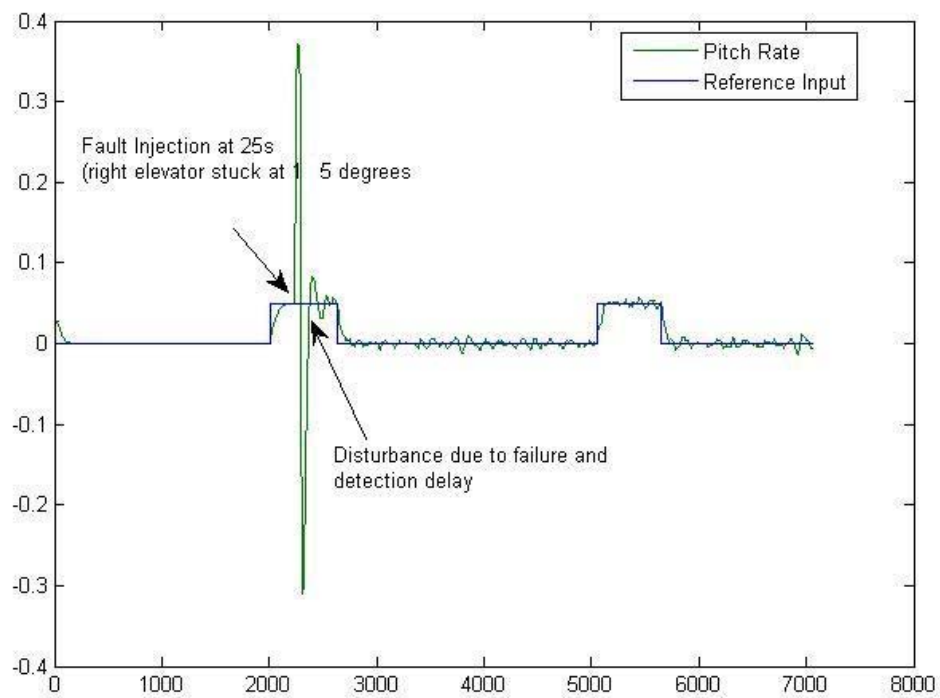


Figure 6.14. Adaptive NDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system.

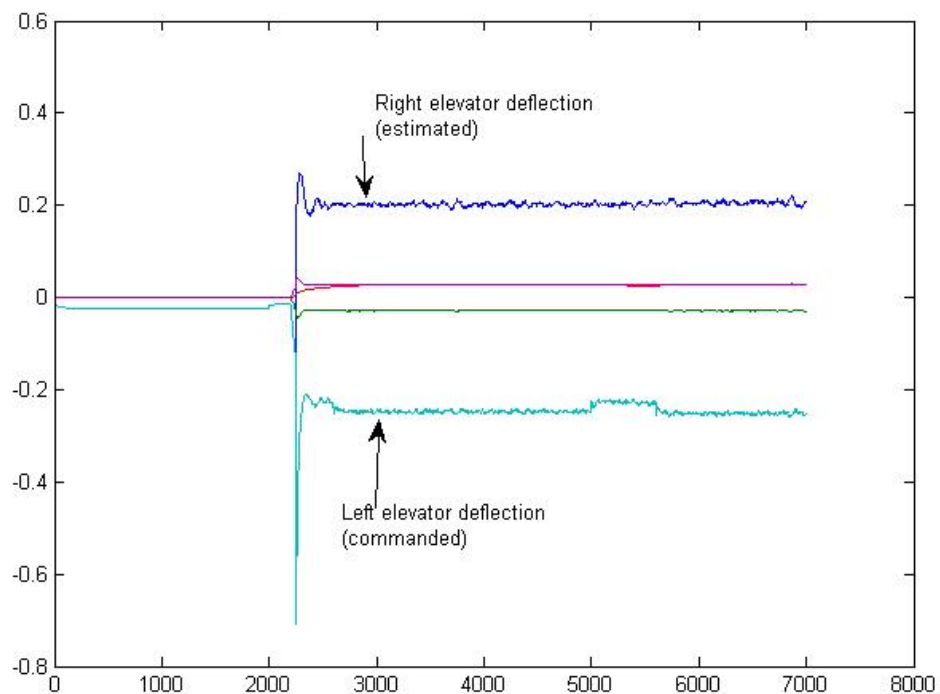


Figure 6.15. Adaptive NDI based fault tolerant controller commanded control surface deflections

The controller performance is influenced by the small fluctuations in the estimated value of the stuck control surface deflection. In order to cope with this issue, a more robust technique can be used. The estimated value of the stuck control surface does not need to be estimated continuously since a stuck actuator does not move. We can then “freeze” the estimate at some mean value at the initial face of fault identification. The use of Incremental NDI (INDI) using the stuck control surface deflection as another state, can eliminate unwanted fluctuations and still achieve perfect tracking of the reference input even in the presence of errors in the estimated faulty deflection. The results from the implementation of this technique is shown in figure 6.16 below, for the case of a right elevator stuck at 5 degrees from trim. A comparison with the results obtained from the simple NDI technique (figure 6.15) are evident. It should be noted that in this particular case, the error in estimation of the stuck control surface deflection was almost 20%.

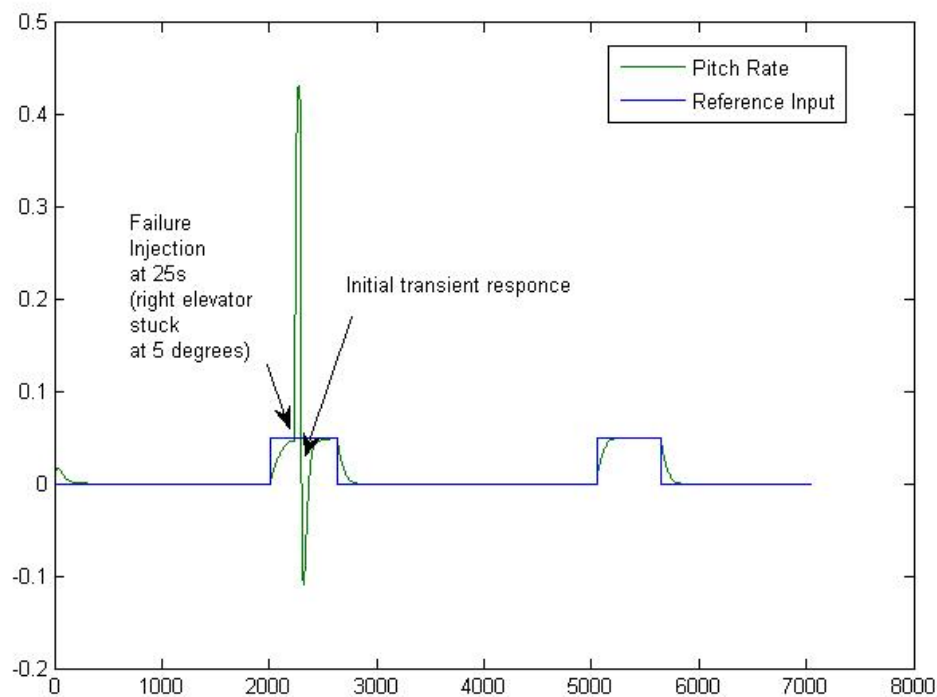


Figure 6.16. INDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system

The above technique is very robust to the estimated faulty deflection. This makes easier the implementation of the method, since it is not critical to accurately estimate the faulty deflection. In

figure 6.17 below, we can see that even if the stuck control surface deflection is estimated with an error equal to 50% of the real value, the controller is able to track the reference input with great accuracy.

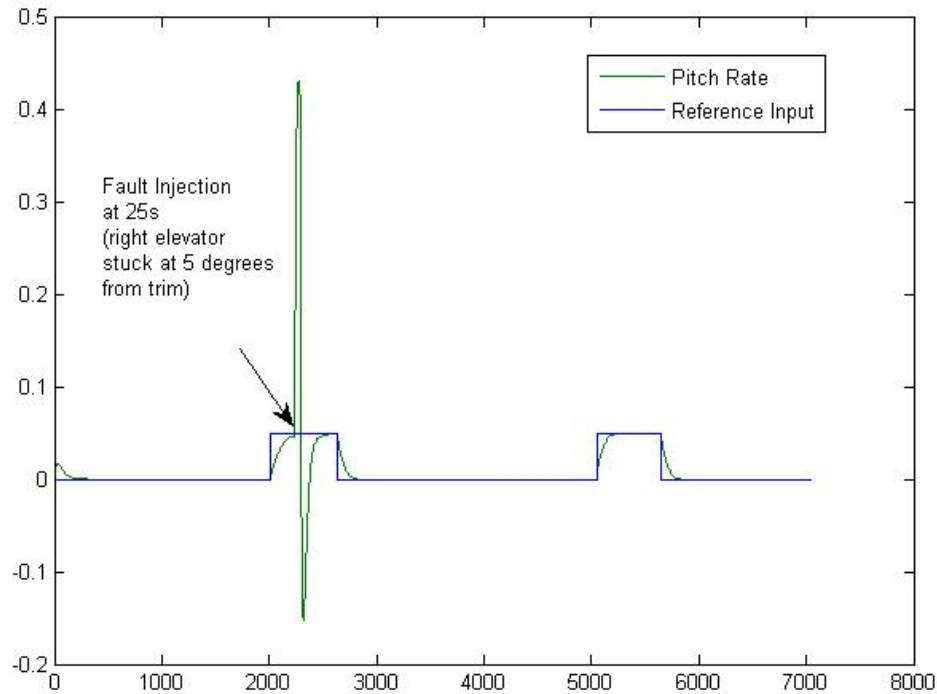


Figure 6.17. INDI based fault tolerant controller capability to track pitch rate step commands after right elevator stuck failure at 5 deg from trim. The failure is injected during excitation of the system. The estimated fault deflection accuracy was chosen 50% away from true value.

It is obvious that the above technique can be used in the case of the adaptive NDI presented in the previous section. The estimated parameters can also freeze to values with a certain accuracy and the INDI controller will compensate for the fault as well as the inaccuracies in the estimated parameters.

Chapter 7. Summary and conclusions

In this thesis, a complete active fault tolerant control system was presented and designed for a fixed wing UAV. Both the FDI and the FTC part of the controller were implemented without any assumption of ideal operation. Each module was developed separately but all modules were tested together. The main design objective was the implementation with no hardware redundancy added something that led to the implementation of a more complicated supervision logic module to differentiate between failures. It should be noted that if position sensors of the control surface deflections were added, the MMAE algorithm would be unnecessary and the FDI module would be much simpler.

The FDI module was developed by mixing two different methodologies: Multiple Model Adaptive Estimation (MMAE) and parameter identification. A supervision module was also developed to make detection of different failures possible. The FDI module was proved to provide accurate identification of all actuator failures with a detection delay of around 1 second (for small failures). The main drawback in this FDI module is the assumption of Gaussian noise used in the EKFs that are part of the MMAE filter. The EKFs need to be tuned in a real life application, however their implementation, up to now, shows that deviations from this assumption can be tolerated by the filter. Another restriction is caused by the false alarms that can be caused by miss-modeling and especially by noise (such as wind). To avoid false alarms and help the FDI module to provide accurate estimates in the case of a structural damage, significant excitation is needed. This issue is common to many FDI algorithms and there are a lot of publications in the literature that deal with it (e.g. [9],[83]). A careful selection of input signals can provide the needed excitation without risking the stability of the system.

The fault tolerant controller developed in chapter 6, is based on non-linear dynamic inversion (NDI). The controller based on the information provided by the FDI system was proved to be able to track rate commands in the presence of control surface failures (both stuck and damaged surfaces). The controller is updated in real time by the new estimated coefficients of the aerodynamic model or the estimated stuck control surface deflection. The control allocation part inherent in an NDI controller helps the re-distribution of control energy to other actuators as long as such redundancy exists. The basic drawback of an NDI controller, its sensitivity to un-modeled dynamics can in this case be resolved by

the re-identification of the model whenever such miss-modeling is detected.

Although model uncertainties were addressed for the FTC design, their influence to the FDI module was not treated in detail. EKFs are known to be sensitive to modeling uncertainties and their performance is expected to be degraded. What's more, the parameter estimation module is also dependent on the MMAE filter performance since it relays on the state estimates provided by the filter. The capability of the FDI method to be insensitive to modeling uncertainties is based on the performance of the fuzzy inference system in the supervision logic. However a detailed assessment of the performance of this scheme is left as a subject of a future research.

Also, in this thesis only the inner control loop was treated, however the fault tolerant capability is strongly related to flight envelope adaptation. It is essential for any successful implementation of FTC algorithms to restrict the flight envelope after failure in order to keep the system under control. Also, performance degradation needs to be applied, permitting slower response of the aircraft after failure in order to continue its mission.

Finally, a more complete simulation in order to assess the performance of the system in a real life scenario is needed. In order to do so, the control system, needs to be completed (outer loop should be designed also) and a navigation algorithm should be designed to generate commands to the control system for trajectory tracking.

Finally, it should be pointed out that the FDI/FTC method presented, like almost any other method found in the literature cannot provide a complete solution to the reliability improvement problem for UAVs. These methods focus on increasing fault tolerance for a given degree of redundancy and, thus, they are limited to the degree of redundancy selected. On the other hand, reliability improvement is a multi-objective optimization problem that involves reliability specifications, redundancy, fault-tolerance evaluation and cost. A schematic representation of a possible design cycle is shown in Figure 7.1.

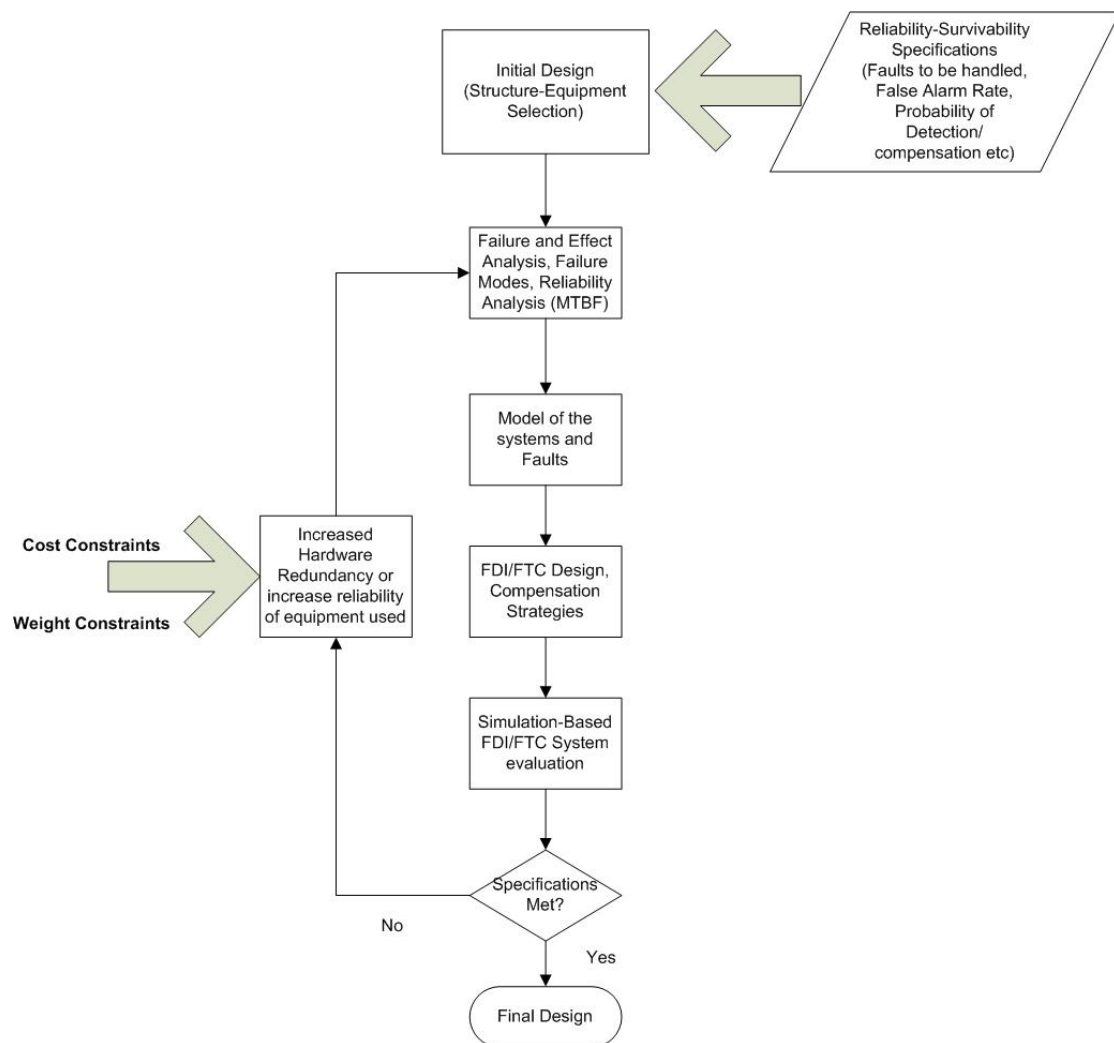


Figure 7.1 Design cycle for reliability improvement of UAVs

BIBLIOGRAPHY

- [1] M. Blanke, M. Kinnaert, J. Lunze and M. Staroswiecki. "Diagnosis and Fault Tolerant Control". *Springer-Verlag, second edition, 2006*.
- [2] R. Isermann, R. Schwarz and S. Stolz. "Fault tolerant drive-by-wire systems". *IEEE Control Systems Magazine, Volume 22, pages 64-81, 2002*.
- [3] Goupil Phillipe. "Airbus state of the art and practices on FDI and FTC", 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain, 30 June – 3 July 2009.
- [4] C. Edwards, T. Lombaerts and H. Smaili. "Fault Tolerant Flight Control-A Benchmark Challenge". *Springer-Verlag, 2010*.
- [5] Anonymous. Civil aviation safety data 1993-2007. Technical Report, Civil Aviation Authority of the Netherlands, CAA-NL, 2008.
- [6] F.W Burcham, C. G. Fullerton and T. A. Maine. "Manual Manipulation of engine throttles for emergency flight control". *Technical Report NASA/TM-2004-212045, NASA, 2004*.
- [7] Anonymous. Unmanned Aircraft Systems Roadmap 2005-2030, *Office of the Secretary of Defence, Washington, D.C, 2005*.
- [8] D.G Ward, et al. "Intelligent Control of Unmanned Air Vehicles: Program summary and representative results". *AIAA-2003-6641, 2nd AIAA "Unmanned Unlimited" Systems, Technologies and Operations, San Diego, California, 15-18 September 2003*.
- [9] K. Valavanis. "Advances in Unmanned Aerial Vehicles. State of the Art and the Road to Autonomy". *Springer-Verlag, 2007*.
- [10] G. Ducard. "Fault-tolerant Flight Control and Guidance Systems". *Springer-Verlag, 2009*.
- [11] Mack Rauw, "FDC 1.4- A Simulink Toolbox for Flight Dynamics and Control Analysis". <http://www.dutchroll.com>, 2005.

- [12] B. Stevens and F. Lewis. "Aircraft Control and Simulation", second edition. *Wiley, New York, 2003*.
- [13] R.C Nelson, "Flight Stability and Automatic Control", second edition. McGraw-Hill 1998.
- [14] T. Nguyen et al. "Simulator Study of Stall/Post-stall Characteristics of a fighter Airplane with Relaxed Longitudinal Static Stability", NASA Technical Paper 1536, 1979.
- [15] D. Jung and P. Tsiotras. "Modeling and Hardware in the Loop Simulation for Small Unmanned Aerial Vehicle", AIAA, 2007.
- [16] V. Klein and E. Morelli. "Aircraft System Identification-Theory and Practice". *AIAA Education Series, 2006*.
- [17] Halim Alwi. "Fault Tolerant Sliding Mode Control Schemes with Aerospace Applications". PhD Thesis, *Control and Instrumentation Research Group, University of Leister, 2008*.
- [18] Perhinschi M.G et al. "Modeling and Simulation of Failures for Primary Control Surfaces". *AIAA Modeling and Simulation Conference, 5-8 August 2002, Monterey, California, AIAA 2002-4786*.
- [19] Smaili M.H. "Flight data reconstruction and simulation of El Al Flight 1862". *Final Thesis, Delf University of Technology, Faculty of Aerospace Engineering, Delf, The Netherlands, 1997*.
- [20] Johnson E.N., Calise A.J. and Blauwe. "In Flight Validation of Adaptive Flight Control Methods". *AIAA Guidance, Navigation and Control Conference, 18-21 August 2008, Honolulu, Hawaii, AIAA 2008-6989*.
- [21] Patton J. Ron. "Fault-Tolerant Control Systems: The 1997 Situation". *Proceedings of the 3rd IFAC symposium on fault detection, supervision and safety of technical processes (pp 1033-1055), August 1997*.
- [22] Zhang Y. and Jiang J. "Bibliographical review on reconfigurable fault-tolerant control systems". *Annual Reviews in Control vol 32 (pp 229-252), 2008*.

- [23] Venkatasubramanian k. et .al. "A review of process fault detection and diagnosis Part I (Quantitative model-based methods) , Part II (Qualitative models and search strategies, Part III (Process history based methods)". *Computers and Chemical Engineering* vol 27 (pp 293-346), 2003.
- [24] Marzat J. Et. al. "Autonomous Fault Diagnosis: State of the Art and Aeronautical Benchmark". *3rd European Conference for Aero-Space Sciences EUCASS, Versailles, France, 2009.*
- [25] S.X Ding. "Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms and Tools". Springer-Verlag, 2008.
- [26] Ward D.G., Monaco J.F. and Bodson M. "Development and Flight Testing of a Parameter Identification Algorithm for Reconfigurable control". *AIAA Journal of Guidance, Control and Dynamics*, vol. 21, pp. 948-956, November-December 1998.
- [27] Shore D. and Bodson M. "Flight Testing of a Reconfigurable Control System on an Unmanned Aircraft". *AIAA Journal of Guidance, Control and Dynamics*, vol. 28, pp. 698-707, July-August 2005.
- [28] Morelli E.A. "Real Time Parameter Estimation in the Frequency Domain". *Proceedings of the 1999 AIAA Atmospheric Flight Mechanics Conference, AIAA paper 99-4043, Portland, August 1999.*
- [29] Raol J.R., Girija G. and Singh J. "Modelling and Parameter Estimation of Dynamic Systems". *IEE Control Series* vol. 65, 2004.
- [30] Song Y., Campa G., Napolitano M., Seanor B. and Perhinschi M.G. "Comparison of On-Line Parameter Estimation Techniques within a Fault Tolerant Flight Control System". *AIAA Journal of Guidance, Control and Dynamics*, vol. 25(3), pp. 528-537, May-June 2002.
- [31] Lombaerts T.J.J., et. al. "Real time damaged aircraft model identification for reconfigurable control". *Proceedings of the AIAA Atmospheric Flight Mechanics Conference, AIAA paper 2007-6717, Hilton Head, SC, August 2007.*

- [32] Chen W. and Patton R.J. "Robust model-based Fault Diagnosis for Dynamic Systems". *Kluwer Academic Publishers*, 1999.
- [33] Shim D.S. and Yang C.K. "Geometric FDI based on SVD for redundant inertial sensor systems". *Proceedings of the 5th Asian Control Conference*, vol.29, pp.1093-1099, Melbourne, Australia, 2004.
- [34] Yang C.K. and Shim D.S. "Double Faults isolation based on the reduced order parity vectors in redundant sensor configuration". *International Journal of Control, Automation and Systems*, 5(2), pp. 155-160, 2007.
- [35] Castaldi P. et al. "Design of residual generators and adaptive filters for the FDI of aircraft model sensors". *Control Engineering Practice*, vol. 18(5), pp. 449-459, May 2010.
- [36] Benini M., Castaldi P. and Simani S. "Fault Diagnosis for Aircraft Systems Models". *VDM Verlag*, 2009.
- [37] Witczak M. "Modelling and Estimation Strategies for Fault Diagnosis of Non-Linear Systems. From Analytical to Soft Computing Techniques". *Springer-Verlag*, 2007.
- [38] Heredia G., Ollero A., Mahtani R. and Bejar M. "Detection of Sensor Faults in Autonomous Helicopters". *Proceeding of the 2005 IEEE International Conference on Robotics and Automation, (ICRA 2005)*, pp.2229-2234, Barcelona, Spain, 2005.
- [39] Grewal M.S. and Andrews A.P. "Kalman Filtering. Theory and Practice Using MATLAB". *John Wiley and Sons*, Second Edition, 2001.
- [40] Ni L. "Fault-Tolerant Control of Unmanned Underwater Vehicles". PhD Thesis, VA Tech. University, Blacksburg, VA, 2001.
- [41] Hajiyeve C. and Caliskan F. "Fault Diagnosis and Reconfiguration in Flight Control Systems". *Kluwer Academic Publishers*, 2003.

- [42] Meybeck P.S. "Multiple-Model Adaptive Algorithms for Detecting and Compensating Sensor and Actuator/Surface Failures in Aircraft Flight Control Systems". *International Journal of Robust and Nonlinear Control*, vol. 9(14), pp.1051-1070, 1999.
- [43] Kobayashi T. "Aircraft Engine Sensor/Actuator/Component Fault Diagnosis Using a Bank of Kalman Filters". *Technical Report NASA/CR 2003-212298*, 2003.
- [44] Zhang Y. and Li X.R. "Detection and Diagnosis of Sensor and Actuator Failures Using IMM Estimator". *IEEE Transactions on Aerospace and Engineering Systems*, vol. 34(4) pp.1293-1313, 1998.
- [45] Fisher K.A. and Maybeck P.S. "Multiple-Model Adaptive Estimation with Filter Spawning". *IEEE Transactions on Aerospace and Engineering Systems*, vol. 38(3) pp.755-768, 2002.
- [46] Freddi A., Longhi S. and Monteriu A. "A model-based fault diagnosis system for unmanned aerial vehicles". *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pp. 71-76, Barcelona, Spain, June 30-July 3, 2009.
- [47] Venkateswaran N. et. al. "Analytical Redundancy-based fault detection of gyroscopes in spacecraft applications". *Acta Astronautica* vol. 50(9), pp. 535-545, 2002.
- [48] Patton R.J., Uppal F.J., Simani S. and Polle B. "Robust FDI applied to thruster faults of a satellite system". *Control Engineering Practice*, vol. 18, pp. 1093-1109, 2010.
- [49] Caliskan F., Zhang Y., Wu N.E. and Shin J.Y. "Estimation of Actuator fault parameters in a nonlinear Boeing 747 model using a linear two-stage Kalman filter". *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pp. 71-76, Barcelona, Spain, June 30-July 3, 2009.
- [50] Hou M. and Patton R.J. "An LMI approach to H_∞/H_- fault detection observers". *Proceedings of the UKACC International Conference*, vol. 1, pp. 305-310, 1996.

- [51] Henry D., Zolghadri A., Castang F. and Monsion M. "A new Multi-Objective Filter Design for Guaranteed Robust FDI performance". *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, Florida, pp. 173-178, USA, 2001.
- [52] Henry D. and Zolghadri A. "Design of fault diagnosis filters: A multi-objective approach". *Journal of the Franklin Institute*, vol. 342, pp. 421-446, 2005.
- [53] Henry D. and Zolghadri A. "Design and analysis of robust residual generators for systems under feedback control". *Automatica*, vol. 41, pp. 251-264, 2005.
- [54] Ding S.X, Jeinsch T., Frank P.M. and Ding E.L. "A unified approach to the optimization of fault detection systems". *International Journal of Adaptive Control and Signal Processing*, vol. 14, pp. 725-745, 2000.
- [55] Keller J.K. and Darouach M. "Optimal two-stage Kalman filter in the presence of random bias". *Automatica*, vol. 33(9), pp. 1745-1748, 1997.
- [56] Wu N.E., Zhang Y. and Zhou K. "Detection, Estimation and accommodation of loss of control effectiveness", *International Journal of Adaptive Control and Signal Processing*, vol. 14, pp. 775-795, 2000.
- [57] Mangoubi R.S. "Robust Estimation and Failure Detection. A concise Treatment". *Springer-Verlag*, 1998.
- [58] Castro H., Bennani S. and Marcos A. "Integrated vs Decoupled Fault Detection Filter & Flight Control Law Designs for a Re-entry Vehicle", *Proceedings of the IEEE International Conference on Control Applications*, pp.3295-3300, Munich, Germany, October 4-6, 2006.
- [59] Kerr M.L., Marcos A., Penin L.F. and Bornschlelg E. "Gain-scheduled fdi for a re-entry vehicle". *AIAA Guidance, Navigation and Control Conferences and Exhibit*, AIAA-2008-7266, Honolulu, Hawaii, 18-21 August, 2008.

- [60] Marcos A., Ganguli S. and Balas G. "An application of H_∞ fault detection and isolation to a transport aircraft", *Control Engineering Practice*, vol. 13, pp. 105-119, 2005.
- [61] Bokor J. And Szabo Z. "Fault detection and isolation in nonlinear systems", *Annual Reviews in Control*, vol. 33, pp.113-123, 2009.
- [62] Marzat J., Piet-Lahanier H., Damongeot F. and Walter E. "Fault diagnosis for nonlinear aircraft based on control-induced redundancy", *IEEE Conference on Control and Fault-Tolerant Systems*, Nice, France, 1-24 Sep. 2010.
- [63] Cork L. and Walker R. "Sensor Fault detection for UAVs using a nonlinear dynamic model and the IMM-UKF algorithm. *Information, Decision and Control*, Aalborg, pp. 230-235, 2007.
- [64] Verma V., Gordon G., Simmons R. and Thrun S. "Real-time fault diagnosis", *IEEE Robotics & Automation Magazine*, vol 11(2), pp. 56-66, 2004.
- [65] Zhang Q., Camplillo F., Cerou F. and LeGland F. "Nonlinear system fault detection and isolation based on bootstrap particle filters", *44th IEEE Conference on Decision and Control*, pp. 3821-3826, Seville, Spain, 12-15 December, 2005.
- [66] Korbricz J., Witczak M. and Puig V. "LMI-based strategies for designing observers for non-linear discrete time systems", *Bouletin of the Polish Academy of Sciences*, vol 55(1), pp. 31-42, 2007.
- [67] De Persis C. and Isidori A. "A Geometric Approach to Nonlinear Fault Detection and Isolation", *IEEE Transactions on Automatic Control*, vol 46(6), pp. 853-865, 2001.
- [68] Edwards C., Spurgeon S. and Patton R.J. "Sliding mode observers for fault detection and isolation", *Automatica*, vol 36(1), pp. 541-553, 2000.
- [69] Busvelle E. and Gauthier J.P. "High-gain and non-high-gain observers for nonlinear systems", *Contemporary Trends on Nonlinear Geometric Control Theory*, pp. 257-286, 2002.

- [70] Alwi H. and Edwards C. "Fault detection and fault-tolerant control of a civil aircraft using sliding mode based scheme", *IEEE Transactions on Control Systems Technology*, vol 16(3), pp. 499-510, 2008.
- [71] Andrieu V. and Praly L. "On the existence of a Kazantzis-Kravaris/Leunberger observer", *SIAM Journal on Control and Optimization*, vol 45(2), pp. 432-456, 2007.
- [72] Szaszi I., Marcos A., Balas G.J. and Bokor J. "Linear parameter-varying detection filter design for a Boeing 747-100/200 aircraft", *Journal of Guidance, Control and Dynamics*, vol 28, pp. 461-470, 2005.
- [73] Akhenak A., Chadli M., Ragot J. and Maquin D. "Design of observers for Tagaki-Sugeno fuzzy models for fault detection and isolation", *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pp. 71-76, Barcelona, Spain, June 30-July 3, 2009.
- [74] Clark R.N. "Instrument fault detection", *IEEE Transaction on Aerospace and Electronics*, vol. AES-14(2), pp. 456-465, 1978.
- [75] Ortiz A. and Neogi N. "A Dynamic threshold approach to fault detection in uninhabited aerial vehicles", *AIAA Guidance, Navigation and Control Conference*, pp. 1-18, Honolulu, Hawaii, August 18-21, 2008 (AIAA 2008-7420).
- [76] Puig V., Stancu A., Escobet T., Nejari F., Quevedo J. and Patton R.J. "Passive robust fault detection using interval observers: Application to the DAMADICS benchmark problem", *Control Engineering Practice*, vol 14, pp. 621-633, 2006.
- [77] Pouliezos A.D. and Stavrakakis G.S. "Real-Time Fault Monitoring of Industrial Processes", Kluwer Academic Publishers, 1994.
- [78] Isermann R. "Fault diagnosis via parameter estimation and knowledge processing", *Automatica*, vol. 29(4), pp. 815-835, 1994.

- [79] Tan C. and Edwards C. "Sliding mode observers for reconstruction of simultaneously actuator and sensor faults", in Proceedings of the Conference on Decision and Control, pp. 1455-1460, 2003.
- [80] Zhang Y.M. and Jiang J. "Active fault-tolerant control system against partial actuator failures", IEE Proceedings on Control Theory Applications, vol. 149(1), pp. 815-835, 2002.
- [81] Zarchan P. and Musoff H. "Fundamentals of Kalman Filtering: A Practical Approach", Second Edition, Volume 208, Progress in Aeronautics and Astronautics, AIAA Inc, Reston, VA, USA, 2005.
- [82] Monaco F.J, Ward D.G. and Bateman A.J.D. "A Retrofit Architecture for Model-Based Adaptive Flight Control", AIAA 1st Intelligent Systems Technical Conference, pp. 1-18, Chicago, Illinois, September 19-22, 2004 (AIAA 2004-6281).
- [83] Oppenheimer M.W. and Doman David B. "Efficient Reconfiguration and Recovery from Damage of Air Vehicles". AIAA Guidance, Navigation and Control Conference and Exhibit, pp. 1-29, Keystone, Colorado, August 21-24, 2006 (AIAA 2006-6552).
- [84] Tomayko James "Story of Self-Repairing Flight Control Systems". *NASA Dryden Flight Research Center, 2003.*
- [85] Steinberg Marc "A Historical Overview of Research in Reconfigurable Flight Control"
(http://acgsc.org/Meetings/Meeting_95/Subcommitte%20E/5.4.pdf)
- [86] Zhou K. and Ren Z. "A new controller architecture for high performance, robust and fault tolerant control". *IEEE Transactions on Automatic Control, Vol. 46, pp. 1613-1618.*
- [87] Ye S., Zhang Y., Li Y., Wang X. and Rabbath C-A. "Robust Fault-Tolerant Tracking Control with Application to Flight Control Systems with Uncertainties". Proceedings of the 10th IASTED International Conference on Control and Applications, 2008.
- [88] Hess R.A. and Wells S.R. "Sliding Mode Control applied to Reconfigurable Flight Control Design". AIAA Journal of Guidance, Control and Dynamics, Vol. 26, pp. 452-462, 2003.

- [89] Alwi H. and Edwards C. "Fault Detection and Fault Tolerant Control of a Civil Aircraft using a Sliding-Mode-Based Scheme". *IEEE Transactions on Control Systems Technology*, Vol. 16(3), pp.499-510, 2008.
- [90] Boscovic J.D. and Mehra R.K. "A Multiple-Model-based Reconfigurable Flight Control System Design". *Proceedings of the 37th IEEE Conference on Decision and Control*, pp. 4503-4508, Tampa, Florida, December 1998.
- [91] Aravena J., Zhou K., Li X.R. and Chowdhury F. "Fault tolerant safe flight controller bank". *Proceedings of the IFAC Symposium SAFEPROCESS '06, Beijing*, pp. 8908-8912, 2006.
- [92] Harefors M. and Bates D.G. "Integrated propulsion-based flight control system design for a civil transport aircraft". *Proceedings of the 2002 IEEE International Conference on Control Applications*, pp. 132-137, 2002.
- [93] Burcham F.W., Fullerton C.G. and Maine T.A. "Manual manipulation of engine throttles for emergency flight control. Technical Report NASA/TM-2004-212045, NASA, 2004.
- [94] Tucker T. "Touchdown: the development of propulsion controlled aircraft at NASA Dryden. Monographs in Aerospace History, 1999.
- [95] Rago C., Prasanth R., Mehra R.K. and Fortenbaugh R. "Failure detection and identification and fault tolerant control using the IMM-KF with applications to the Eagle-Eye UAV. *Proceedings of the 37th IEEE Conference on Decision and Control*, pp. 4503-4508, Tampa, FL, December 1998.
- [96] Zhang Y. and Jiang J. "Integrated active fault-tolerant control using IMM approach. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, pp. 1221-1235, 2001.
- [97] Shin J-Y and Gregory I. "Robust Gain-Scheduled Fault Tolerant Control for a Transport Aircraft", *Proceedings of the 16th IEEE Conference on Control Applications (CCA 2007)*, 1-3 Oct. 2007.

- [98] Ganguili S., Marcos A. and Balas G.J. "Reconfigurable LPV control design for Boeing 747-100/200 longitudinal axis", Proceedings of the American Control Conference, pp. 3612-3617, 2002.
- [99] Maciejowski J.M. and Jones C.N. "MPC fault-tolerant control case study: flight 1862. Proceedings of the IFAC Symposium SAFEPROCESS '03, Washington, USA, pp. 119-124, 2003.
- [100] Campell M.E., Lee J.W., Scholte E. and Rathbun D. "Simulation and Flight Test of Autonomous Aircraft Estimation, Planning and Control Algorithms. AIAA Journal of Guidance, Control and Dynamics, Vol. 30(6), pp. 1597-1609, Nov.-Dec. 2007.
- [101] Shin Y., Calise A. J. and Johnson M.D. "Adaptive control of advanced fighter aircraft in nonlinear flight regimes". AIAA Journal of Guidance, Control and Dynamics, Vol. 31(5), pp. 1464-1477, Sep.-Oct. 2008.
- [102] Tao G., Chen S., Tang X. and Joshi S.M. "Adaptive Control of Systems with Actuator Failures". Springer-Verlag, London Berlin Heidelberg, 2004.
- [103] Shore D. and Bodson M. "Flight Testing of a Reconfigurable Control System on an Unmanned Aircraft". AIAA Journal of Guidance, Control and Dynamics, Vol. 28(4), pp. 698-707, July-August 2005.
- [104] Fekri S., Athans M. and Pascoal A. "Issues, Progress and New Results in Robust Adaptive Control". International Journal of Adaptive Control and Signal Processing, Vol 20(10), pp. 519-579, 2006.
- [105] Zhong Y., Yang L. and Shen G. "Control Allocation Based Reconfigurable Flight Control for Aircraft with Multiple Control Effectors", 47th AIAA Aerospace Sciences Meeting, pp. 1-12, Orlando, Florida, January 5-8, 2009 (AIAA 2009-58).

- [106] Lombaerts T.J.J., Huisman H.O., Chu Q.P., Mulder J.A. and Joosten D.A. "Flight Control Reconfiguration based on Online Physical Model Identification and Nonlinear Dynamic Inversion". Proceedings of the AIAA Guidance, Navigation and Control Conference, 18-21 August 2008, Honolulu, Hawaii, USA (AIAA 2008-7435 pp. 1-24).
- [107] Mahmoud M., Jiang J. and Zhang Y.M. "Active Fault tolerant control systems: Stochastic analysis and synthesis". Lecture notes in Control and Information Sciences, Volume 287, Berlin, Germany, Springer, 2003.
- [108] Zhang Y.M. and Jiang J. "Fault tolerant control system design with explicit consideration of performance degradation", IEEE Transactions on Aerospace and Electronic Systems, Vol. 39(3), pp. 838-848, 2003.
- [109] Jiang J. and Zhang Y.M. "Accepting performance degradation in fault tolerant control system design", IEEE Transactions on Control Systems Technology, Vol. 24(2), pp. 284-292, 2006.
- [110] Thelliol D., Join C. and Zhang Y. "Actuator Fault-Tolerant Control Design based on Reconfigurable Reference Input", International Journal of Applied Mathematics and Computer Sciences, Vol. 18(4), pp. 553-560, 2008
- [111] Staroswieski M., Yang H. and Jiang B. "Progressive accommodation of parametric faults in linear quadratic control", Automatica, Vol. 43, pp. 2070-2076, 2007.
- [112] Joosten D.A., Van den Boom T.J.J. and Lombaerts T.J.J. "Fault-tolerant control using dynamic inversion and model-predictive control applied to an aerospace benchmark", Proceedings of the 17th World Congress of the International Federation of Automatic Control, pp. 12030-12035, Seoul, Korea, July 6-11, 2008.
- [113] Xu Y., Jiang B., Gao Z. and Zhang K. "Fault tolerant control for near space vehicle: a survey and some new results", Journal of Systems Engineering and Electronics, Vol. 22(1), pp. 88-94, 2011.

[114] Ahn C., Kim Y. and Kim H.J. "Adaptive sliding mode controller design for fault tolerant flight control system", Proceedings of AIAA Guidance, Navigation and Control Conference, pp. 1-8, 2006.

[115] Tang X.T., Tao G. and Joshi S.M. "Adaptive actuator failure compensation for parametric strict feedback systems and an aircraft application", Automatica, Vol. 39(11), pp. 1975-1982, 2003.

APPENDIX

«Aerodynamic Modeling of the ETH Zurich Aerobatic UAV»

The modeling of the ETH Zurich aerobatic UAV aerodynamic forces and moments are presented below [9].

Lift Force: $Z^w = \bar{q}SC_z(a)$, $C_z(a) = C_{z_1} + C_{z_a}a$

Lateral Force: $Y^w = \bar{q}SC_Y(\beta)$, $C_Y(\beta) = C_{Y_1}\beta$

Drag Force: $X^w = \bar{q}SC_X(a, \beta)$, $C_X(a, \beta) = C_{X_1} + C_{X_a}a + C_{X_{a^2}}a^2 + C_{X_{\beta^2}}\beta^2$

Roll Torque: $L^b = \bar{q}SbC_L(\delta a, \beta, \tilde{p}, \tilde{r})$,

$C_L(\delta a, \beta, \tilde{p}, \tilde{r}) = C_{L_a}\delta a + C_{L_\beta}\beta + C_{L_{\tilde{p}}}\tilde{p} + C_{L_{\tilde{r}}}\tilde{r}$

Pitch Torque: $M^b = \bar{q}S\bar{c}C_M(\delta e, a, \tilde{q})$,

$C_M(\delta e, a, \tilde{q}) = C_{M_{\delta e}}\delta e + C_{M_a}a + C_{M_{\tilde{q}}}\tilde{q}$

Yaw Torque: $N^b = \bar{q}SbC_N(\delta r, \tilde{r}, \beta)$,

$C_N(\delta r, \tilde{r}, \beta) = C_{N_{\delta r}}\delta r + C_{N_\beta}\beta + C_{N_{\tilde{r}}}\tilde{r}$

In the above equations, the superscript «w» indicates the wind axis system and «b» the body axis.

The other symbols indicate:

- S: wing surface, b: wing span, \bar{c} : mean aerodynamic chord
- \bar{q} : dynamic pressure, a : angle of attack, β : sideslip angle,
- $\delta a, \delta e, \delta r$: deflection of aileron, elevator and rudder surfaces
- $\tilde{p}, \tilde{q}, \tilde{r}$: dimensionless angular rates
 $(\tilde{p} = \frac{bp}{2V_T}, \tilde{q} = \frac{\bar{c}q}{2V_T}, \tilde{r} = \frac{br}{2V_T})$