Fault Diagnosis and Fault-Tolerant Control for Aircraft Subjected to Sensor and Actuator Faults

Lu, Peng

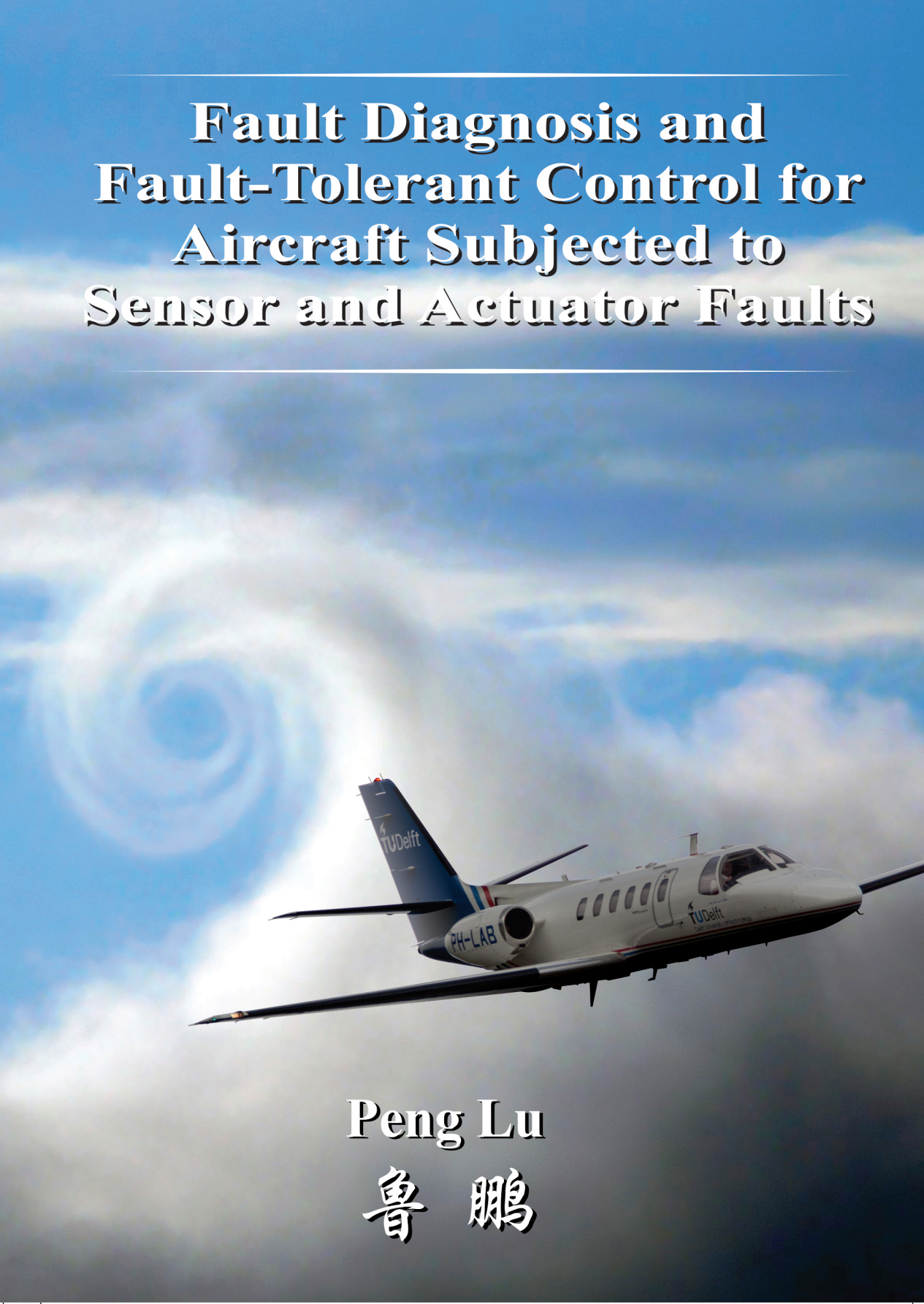
DOI
10.4233/uuid:11be69d2-44ae-429c-9746-7e3ced35f464

Publication date
2016

Document Version
Final published version

Citation (APA)
Lu, P. (2016). *Fault Diagnosis and Fault-Tolerant Control for Aircraft Subjected to Sensor and Actuator Faults*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:11be69d2-44ae-429c-9746-7e3ced35f464

Fault Diagnosis and Fault-Tolerant Control for Aircraft Subjected to Sensor and Actuator Faults



Peng Lu

鲁 鹏

Fault Diagnosis and Fault-Tolerant Control for Aircraft Subjected to Sensor and Actuator Faults

Fault Diagnosis and Fault-Tolerant Control for Aircraft Subjected to Sensor and Actuator Faults

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 12 september 2016 om 15.00 uur

door

Peng Lu

Master of Science, Northwestern Polytechnical University, China
geboren te Hefei, Anhui Province, China

Dit proefschrift is goedgekeurd door de

promotor: Prof. dr. ir. M. Mulder
copromotor: Dr. Q.P. Chu

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. M. Mulder,	Technische Universiteit Delft, promotor
Dr. Q.P. Chu,	Technische Universiteit Delft, copromotor
Prof. dr. ir. M. Verhaegen,	Technische Universiteit Delft
Prof. dr. A. Zolghadri,	University of Bordeaux
Prof. Dr. -Ing. F. Holzapfel,	Technische Universität München
Prof. Dr. -Ing. S. Hecker,	Munich University of Applied Sciences
Dr. ir. E. van Kampen,	Technische Universiteit Delft
Prof. dr. E.K.A. Gill,	Technische Universiteit Delft, reservelid

Keywords: Fault-Tolerant Control; Fault Detection and Diagnosis; Nonlinear Control; Flight Control; Sensor faults; Actuator faults; Extended Kalman Filter; Unscented Kalman Filter; Adaptive filtering; Disturbance estimation; Fault estimation; Turbulence; Real flight data

Printed by: Ipskamp Drukkers, Enschede, The Netherlands

ISBN 978-94-6186-701-8

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

To my beloved parents and wife

献给我亲爱的父母和妻子

SUMMARY

With the development of more advanced and complex aircraft, new fault modes, which have never been encountered before, may occur during flight. These faults can be difficult to detect using existing onboard monitoring systems, harming aviation safety. To mitigate potential hazardous consequences, Fault-Tolerant Control (FTC) systems have been proposed to enhance aircraft safety in case unknown faults occur.

Generally speaking, two types of FTC systems exist: Active Fault-Tolerant Control (AFTC) systems and Passive Fault-Tolerant Control (PFTC) systems. The key difference between these two systems is whether the system includes a capability to detect faults and provide fault information to the FTC system, or not. This capability is referred to as Fault Detection and Diagnosis (FDD). Because PFTC systems do not include fault detection they can only deal with limited fault types and have difficulties in dealing with sensor faults. In this thesis active systems are considered, including FDD, which are designed to deal with a great variety of fault types, including sensor faults.

Current onboard monitoring systems available to cope with sensor faults are primarily based on establishing redundancy in the physical hardware, and then perform statistical consistency checks among the set of similar sensors. These checking schemes can fail, however, when unexpected faults occur that were never encountered before and for which the scheme has not been designed. This motivates the development of *analytical* redundancy, which uses mathematical models of the aircraft.

Analytical redundancy differs from hardware redundancy as it replaces the set of physical sensors with mathematical models. Combining these mathematical models with the measured inputs and outputs to the system, methods based on analytical redundancy then check whether faults occur in the inputs, outputs or the system itself. An additional advantage is that it reduces weight, cost and volume as compared to hardware redundancy. This is beneficial especially for small Unmanned Aerial Vehicles (UAVs) such as quadrotors or micro aerial vehicles.

The performance of analytical redundancy can be significantly degraded by model uncertainties and disturbances. Disturbances include external disturbances such as wind and turbulence, and internal disturbances such as noise. In addition, most existing FDD approaches are based on the aerodynamic model of the aircraft. To compute the state of the aircraft such as the translational and rotational velocities, the aerodynamic forces and moments are required. The aerodynamic forces and moments are calculated based on the stability derivatives and control derivatives. These derivatives vary according to the flight conditions depending on variables such as dynamic pressure and Mach number. Although computational fluid dynamics, wind tunnel tests and extensive flight tests are performed to obtain these derivatives, their accuracy is limited, which leads to model uncertainties. Due to the model uncertainties, the calculation of the aerodynamic forces and moments will not be correct, which results in incorrect translational and rotational velocities. In addition, the disturbances can also generate additional aerodynamic

forces and moments which affect the calculation of the forces and moments. Overall, the performance of the analytical redundancy is limited.

Model uncertainties and disturbances also affect the performance of the FTC system. The FTC system contains a reconfigurable controller, which is usually designed based on aircraft models that are linearized around different operating points. Since the aircraft model is nonlinear, gain scheduling is required. The design of gain-scheduling linear controllers for the entire aircraft flight envelope is time-consuming, however, and nonlinear control approaches have been proposed to solve this problem.

Nonlinear Dynamic Inversion (NDI) and Backstepping (BS) are model inversion-based control approaches which can deal with nonlinearities in the system and remove the need for gain scheduling. However, since these approaches are also based on the aircraft model, uncertainties in the model and unknown disturbances will still influence the performance of the controllers. The way in which model uncertainties and disturbances affect the performance of the controllers is very similar to the influence on the FDD system. That is, the calculated aircraft velocities and angular rates will be wrong when there are model uncertainties in the calculation of the aerodynamic forces and moments.

In literature, several control approaches have been proposed to deal with model uncertainties such as NDI with online model identification. The core of these methods is that the uncertain derivatives are identified online through parameter estimation. The identified parameters are then fed back to the controller to reduce the influence of model uncertainties. However, this approach typically requires excitation of the aircraft which can be dangerous when faults occur. Adaptive Backstepping (ABS) is another type of control method which can guarantee the stability of the system by updating the unknown parameters through a parameter update law. However, the computational load involved with ABS is intensive and the tuning of the parameter update laws can be tedious, especially when there are many unknown parameters.

The main research goal of this dissertation is to increase the robustness of FDD and FTC approaches in the presence of aircraft model uncertainties and disturbances. The following three research problems are addressed:

- How to increase the robustness of sensor FDD systems?
- How to increase the robustness of actuator FTC systems?
- How to achieve simultaneous sensor and actuator FTC?

The first question is addressed in two parts of the thesis: Part I and Part II. The considered sensor types in this research are the Inertial Measurement Unit (IMU) sensor and the Air Data Sensor (ADS).

Part I aims to increase the robustness of sensor FDD systems with respect to *model uncertainties*. First of all, an airspeed-based kinematic model (AS-KM) is used instead of the aerodynamic model. In contrast to existing approaches which use the aerodynamic model, here we calculate the aerodynamic forces directly through the measurements of specific forces. This step significantly reduces the effects of model uncertainties encountered in existing techniques.

Apart from the choice of the model that underlies the computations, advanced filtering techniques are required to estimate both the state and the faults. A second advancement of this thesis is that existing filtering techniques have been improved and several novel advanced filters are proposed:

- A Selective-Reinitialization Multiple-Model Adaptive Estimation (SRMMAE) approach is proposed for the FDD. It introduces three novel Selective Reinitialization (SR) algorithms, which significantly improve the FDD performance of existing Multiple-Model Adaptive Estimation (MMAE) approaches. The SRMMAE reduces the number of false alarms, and also more quickly detects that faults have been resolved as compared to conventional MMAE techniques.

- A novel Double-Model Adaptive Estimation (DMAE) approach is proposed. Although the SRMMAE improves the performance of the MMAE, its computational load is intensive as it requires a model set consisting of a number of models to represent different fault scenarios. Especially when it deals with simultaneous faults, the number of models in the model set can be large. The proposed DMAE approach reduces this number to two, even when it deals with simultaneous faults. The SR algorithms developed before are used to maintain satisfactory performance of the SRMMAE.

- A number of other state-of-the-art filtering approaches are improved. For instance, to reduce the sensitivity of the Optimal Two-Stage Extended Kalman Filter (OTSEKF) to initial conditions, an Iterated Optimal Two-Stage Extended Kalman Filter (IOTSEKF) is proposed in this thesis. To reduce the sensitivity of the Robust Three-Step Unscented Kalman Filter (RTS-UKF) to initial condition errors, an Adaptive Three-Step Unscented Kalman Filter (ATS-UKF) is proposed.

Finally, many of the proposed techniques have been successfully validated using real flight test data. These data contain real-life model uncertainties and disturbances, which are by definition unknown and which yield a good source to validate the performance of the proposed approaches.

In Part II of the thesis a number of solutions are discussed to increase the robustness of the FDD approaches with respect to *external disturbances* such as time-varying wind and turbulence. First, when dealing with IMU sensor faults in the presence of disturbances, a ground speed-based kinematic model (GS-KM) model is proposed as the model for FDD. The AS-KM is not accurate when there are disturbances because the disturbances are unknown. The performance of using the GS-KM for IMU sensor FDD is compared to that of using the AS-KM, which demonstrates the advantage of using the GS-KM. For the GS-KM, the ground velocity measurements are assumed to be fault-free while for the AS-KM, the air velocity measurements are assumed to be fault-free.

Second, advanced filtering techniques are required to perform the state and fault estimation. For IMU sensor FDD, an IOTSEKF is proposed which is an iterated version of the OTSEKF. However, its performance is degraded by a bad choice of the covariance matrices. Therefore, an Adaptive Two-Stage Extended Kalman Filter (ATSEKF) is proposed which adaptively updates the covariance matrices and improves the performance of the IOTSEKF. Regarding ADS FDD in the presence of turbulence, the DMAE is

extended so that it can handle the disturbances such as the turbulence. Finally, in order to demonstrate the performance of the proposed approaches in the presence of disturbances such as time-varying wind and turbulence, more flight tests were performed. Aircraft response data in various flight conditions were recorded including the presence of turbulence. The recorded flight test data are used to validate the performance of the proposed approaches, which confirms their performance in the presence of external disturbances.

In Part III of the thesis the second and third research questions are considered.

To increase the robustness of actuator FTC systems, the Incremental Nonlinear Dynamic Inversion (INDI) and Incremental Backstepping (IBS) approaches are proposed as the reconfigurable controller. The INDI, which is able to deal with actuator faults, is extended in its application. An aircraft trajectory controller is designed to follow the trajectory references in the presence of actuator faults. Existing approaches require a parameter update law design or model identification. Furthermore, an aerodynamic model structure has to be designed. The idea of designing the controller is to design the flight path control and angular rate control loops, which contain model uncertainties due to unknown stability and control derivatives, using incremental control approach. The other two loops, the position control and the attitude control loops, are designed based on NDI approach since there are typically no uncertainties in these two loops. The performance of the proposed controller is compared to existing approaches such as the Nonlinear Dynamic Inversion with online Model Identification (NDI-MI). The validation results demonstrate that the proposed control structure can follow the trajectory reference well, without designing a parameter update law or online model identification. In addition, it does not require additional effort to design the aerodynamic model structure.

To achieve simultaneous sensor and actuator FTC, this thesis proposes an FTC system which consists of a sensor FDD system and a reconfigurable controller. State-of-the-art sensor FDD approaches use information of the actuator, such as the commanded or measured control surface deflections, which becomes problematic when the control surfaces themselves have (partly) failed. Therefore, in this dissertation, the sensor FDD is performed *without* using actuator information; actuator faults are dealt with by the reconfigurable controller. Through doing this, even in the presence of actuator faults, the sensor FDD can still be performed, allowing for simultaneous sensor and actuator FTC. The performance of the proposed FTC system is validated with different fault scenarios where sensor and actuator faults happen at the same time.

Future studies on sensor FDD in the presence of more fault scenarios are recommended. This can further demonstrate the performance of the proposed approaches to deal with unknown fault types. Regarding further research on FTC systems, implementation on real aircraft (especially fixed wing aircraft) is recommended, to improve aviation safety.

SAMENVATTING

Met de ontwikkeling van meer geavanceerde en complexere vliegtuigen kunnen nieuwe fouten tegengekomen worden tijdens de vlucht, die nog niet eerder waargenomen zijn. Deze nieuwe fouten kunnen moeilijk detecteerbaar zijn met de monitorende systemen aan boord, met als gevolg dat deze fouten de veiligheid en betrouwbaarheid van en vliegtuig in gevaar brengen. Om mogelijke gevaarlijke gevolgen te verminderen zijn fouttolerante besturingssystemen (FTC) voorgesteld om de veiligheid van het vliegtuig te vergroten wanneer onbekende fouten optreden.

In het algemeen kunnen FTC systemen ingedeeld worden in twee types: Actieve fouttolerante besturingssystemen (AFTC) en passieve fouttolerante besturingssystemen (PFTC). Het onderscheid tussen deze twee systemen wordt gemaakt door de aanwezigheid van een fout detectie en diagnose systeem (FDD) dat informatie over de fout aan het FTC systeem kan geven. Omdat PFTC geen gebruik maakt van FDD kan PFTC slechts overweg met een beperkt aantal types fouten. Het is voor PFTC bijvoorbeeld moeilijk om met sensorfouten om te gaan. In dit proefschrift worden actieve systemen beschouwd, inclusief FDD, die ontworpen zijn om met meer foutsoorten overweg te kunnen, waaronder ook sensorfouten.

Huidige systemen aan boord die beschikbaar zijn om met sensorfouten overweg te gaan zijn primair gebaseerd op het creëren van redundantie in de hardware, om vervolgens statistische consistentietesten uit te voeren op de set van gelijke sensoren. Deze vorm van sensorcontrole kan echter problemen ondervinden wanneer nieuwe fouttypes voorkomen, aangezien de sensor controle niet ontworpen is om met onbekende fouten om te gaan. Dit vormt de motivatie voor de ontwikkeling van een analytische sensorcontrole die gebruik maakt van een mathematisch model van het vliegtuig.

De analytische controle onderscheidt zich van de methode die gebruik maakt van overtollige hardware door een set van sensoren te vervangen door mathematische modellen van het systeem. Door de invoer en uitvoer van het model te combineren kan de analytische methode vaststellen of er fouten voorkomen in de invoer, in de uitvoer, of in het systeemproces. Een bijkomend voordeel van de analytische methode is dat het gewicht, volume en de kosten kleiner zijn dan bij de controle door middel van overtollige hardware. Dit is in het bijzonder belangrijk voor kleine onbemande luchtvaartuigen (UAV's), zoals bijvoorbeeld quadrotors en micro-UAV's.

De prestatie van de analytische methode kan significant verminderd worden door onzekerheden in het model en door verstoringen. Verstoringen kunnen extern zijn, zoals wind en turbulentie, of intern, zoals meetruis. Bestaande FDD methodes maken gebruik van het aerodynamische model van het vliegtuig. Om de toestandsvariabelen van een vliegtuig, zoals de translatie- en rotatiesnelheden, te berekenen zijn de aerodynamische krachten en momenten vereist. De aerodynamische krachten en momenten worden berekend met behulp van de stabiliteits- en besturingsafgeleiden. Deze afgeleiden zijn afhankelijk van onder andere de dynamische druk en het Mach getal en varië-

ren daardoor met de vliegconditie. Computational Fluid Dynamics(CFD) programma's, windtunnel testen en testvluchten worden gebruikt om deze afgeleiden te bepalen. Echter is het moeilijk om deze afgeleiden met hoge nauwkeurigheid te bepalen, waardoor er altijd een mate van modelonzekerheid is. Deze modelonzekerheden zorgen voor een incorrecte berekening van de aerodynamische krachten en momenten, wat leidt tot incorrecte berekening van de translatie- en rotatiesnelheden. Bijkomstig kunnen verstoringen extra aerodynamische krachten en momenten genereren die consequenties hebben voor de berekening van de krachten en momenten.

Modelonzekerheden en verstoringen hebben ook effect op de prestatie van FTC systemen. Het FTC systeem bevat een herconfigureerbare regelaar, die in het algemeen ontworpen wordt op basis van het vliegtuigmodel dat gelineariseerd is voor verschillende vliegcondities. Aangezien het vliegtuig niet lineair is, wordt een interpolatiemethode toegepast op de parameters van de lineaire regelaars. Het bepalen van de parameters voor de lineaire regelaars in alle mogelijke vliegcondities is tijdrovend. Om dit probleem te verhelpen zijn niet-lineaire regelsystemen geïntroduceerd.

Niet-lineaire model inversie (NDI) en Backstepping (BS) zijn methodes, gebaseerd op model inversie, die overweg kunnen met niet-lineaire modellen en het gebruik van de interpolatie methode onnodig maken. Echter, aangezien deze methodes gebruik maken van het model van het vliegtuig, zullen onzekerheden en verstoringen in dit model de prestatie van het regelsysteem beïnvloeden. De manier waarop modelonzekerheden en verstoringen de prestaties beïnvloeden is soortgelijk aan de invloed op het FDD systeem. De berekende translatie- en rotatiesnelheden zijn incorrect wanneer er modelonzekerheden zitten in de berekening van de aerodynamische krachten en momenten.

In de literatuur zijn verschillende methodes voorgesteld die overweg kunnen met modelonzekerheden, zoals bijvoorbeeld NDI met modelidentificatie. In de kern van deze methode worden de onzekere afgeleiden tijdens de vlucht geschat door middel van een parameter identificatiemethode. De geïdentificeerde parameters worden teruggevoerd naar het regelsysteem om zo de invloed van modelonzekerheden te minimaliseren. Deze identificatiemethode vereist echter excitatie van het vliegtuig, wat gevaarlijk kan zijn wanneer er fouten zijn. Adaptive Backstepping (ABS) is een andere regelmethode die stabiliteit garandeert door het aanpassen van de onbekende parameters door middel van een parameter identificatiemethode. ABS vereist echter veel berekeningen en het afstemmen van de parameter identificatiemethode kan lastig zijn, vooral wanneer er veel onbekende parameters zijn.

De hoofdonderzoeksdoel van dit proefschrift is om de robuustheid van FDD en FTC methodes voor modelonzekerheden en verstoringen te vergroten. De volgende onderzoeksvragen zijn opgesteld om dit hoofddoel te bereiken:

- Hoe kan de robuustheid van sensor FDD systemen vergroot worden?
- Hoe kan de robuustheid van aandrijvings-FTC systemen vergroot worden?
- Hoe kan simultaan sensor- en aandrijvings-FTC bewerkstelligd worden?

De eerste vraag wordt behandeld in twee delen van dit proefschrift: Part I en Part II. De sensoren die beschouwd worden in dit onderzoek zijn de Inertial Measurement Unit (IMU) en luchtdata sensors (ADS).

Part I beschrijft de robuustheid van sensor FDD voor modelonzekerheden. Ten eerste wordt een op de snelheid ten opzichte van de lucht gebaseerd kinematisch model (AS-KM) gebruikt in plaats van een aerodynamisch model. In tegenstelling tot bestaande methodes, die gebruik maken van het aerodynamische model, worden de aerodynamische krachten hier berekent uit de gemeten specifieke krachten, waardoor de invloed van modelonzekerheden op de berekening van de aerodynamische krachten vermindert.

Afgezien van de keuze voor het onderliggende model, zijn ook geavanceerde filtertechnieken nodig om zowel de toestandsvariabelen als de fouten te schatten. Een tweede bijdrage van dit proefschrift is een verbetering van de bestaande filtertechnieken en het ontwerp van nieuwe filtertechnieken. Deze zijn als volgt:

- Een Selective-Reinitialization Multiple-Model Adaptive Estimation (SRMMAE) methode wordt voorgesteld voor FDD. De SRMMAE methode heeft drie nieuwe Selective-Reinitialization (SR) algoritmes die de prestaties van de FDD methode in Multiple-Model Adaptive Estimation (MMAE) methodes significant verbetert. De SRMMAE methode vermindert het aantal valse foutmeldingen en kan sneller detecteren wanneer een fout verwijderd is dan MMAE methodes.

- Een innovatieve Double-Model Adaptive Estimation (DMAE) methode wordt voorgesteld. Alhoewel de SRMMAE methode de prestaties van de MMAE methode vergroot, is een grote rekenkracht nodig voor SRMMAE omdat voor elk foutscenario een apart model vereist is. Het aantal vereiste modellen kan heel groot worden, in het bijzonder wanneer er simultane fouten zijn. De voorgestelde DMAE methode beperkt het aantal vereiste modellen tot twee, zelfs bij simultane fouten. De SR algoritmes worden toegepast om de goede prestaties van de SRMMAE methode te behouden.

- Er is ook een aantal moderne filtertechnieken verbetert. Zo is bijvoorbeeld een Iterated Optimal Two-Stage Extended Kalman Filter (IOTSEKF) voorgesteld om de gevoeligheid van het Optimal Two-Stage Extended Kalman Filter (OTSEKF) voor initiële condities te minimaliseren. Om de gevoeligheid van het Robust Three-Step Unscented Kalman Filter (RTS-UKF) voor fouten in de initiële condities te verminderen wordt een Adaptive Three-Step Unscented Kalman Filter (ATS-UKF) voorgesteld.

Tot slot zijn de voorgestelde methodes succesvol gevalideerd met data uit testvluchten. De data uit deze testvluchten bevat realistische modelonzekerheden en verstoringen, welke per definitie onbekend zijn. De data is daardoor een goede bron om de prestaties van de voorgestelde methodes te valideren.

In Part II van dit proefschrift wordt een aantal methodes besproken die de robuustheid van FDD methodes voor externe verstoringen zoals variërende wind en turbulentie vergroten. Ten eerste wordt een kinematisch model, gebaseerd op de snelheid ten opzichte van de grond (GB-KM), voorgesteld als FDD model wanneer er zowel IMU sensorfouten zijn als verstoringen. Het AS-KM is niet nauwkeurig onder invloed van verstoringen, omdat deze onbekend zijn. Uit de vergelijking tussen het AS-KM en het GS-KM voor IMU sensor FDD komt het voordeel van het GS-KM naar voren. Voor het GS-KM

wordt verondersteld dat de metingen van de grondsnelheid foutvrij zijn, net zoals verondersteld wordt dat de snelheidsmetingen ten opzichte van de lucht foutvrij zijn bij het gebruik van het AS-KM.

Ten tweede zijn geavanceerde filtertechnieken nodig voor de schatting van de toestandsvariabelen en de fouten. Voor IMU sensor FDD wordt een IOTSEKF voorgesteld, welke een itererende versie is van het OTSEKF. Echter zal een slechte keuze van de covariantiematrices de prestaties verminderen. Daarom wordt het Adaptive Two-Stage Extended Kalman Filter (ATSEKF) voorgesteld om de covariantiematrices aan te passen en om de prestaties van OTSEKF te verbeteren. Voor de ADS FDD onder invloed van turbulentie wordt de DMAE methode uitgebreid zodat het om kan gaan met deze verstoringen. Tot slot zijn meer testvluchten gemaakt om de prestaties van de voorgestelde aanpak te testen onder invloed van realistische verstoringen, zoals variërende wind en turbulentie. De toestandsvariabelen van het vliegtuig zijn gemeten in verschillende vliegcondities en onder invloed van turbulentie. De gemeten data is gebruikt om de prestaties van de voorgestelde methodes onder invloed van verstoringen te valideren.

In Part III van dit proefschrift worden de tweede en derde onderzoeksvraag behandelt.

Om de robuustheid van aandrijvings-FTC te vergroten worden een incrementele niet-lineaire model inversie (INDI) regelaar en een incrementele Backstepping (IBS) regelaar voorgesteld als herconfigureerbare regelaars in het FTC systeem. De INDI methode, die overweg kan met fouten in de aandrijving van de stuurorganen, wordt toegepast in een regelsysteem dat de baan van het vliegtuig regelt terwijl er fouten zijn in de aandrijving van de stuurorganen. Bestaande methodes hebben hiervoor een parameter aanpassingsalgoritme of een modelidentificatie voor nodig. Bovendien moet de structuur voor het aerodynamische model ontworpen worden. Een op INDI gebaseerd regelsysteem wordt voorgesteld waarbij de regellussen voor de baanhoek en de rotatiesnelheden door INDI aangestuurd worden, omdat deze toestandsvariabelen beïnvloed worden door modelonzekerheden in de besturings- en stabiliteitsafgeleiden. De overige twee regellussen, voor positie en standhoeken, worden ontworpen met een NDI regelaar, aangezien er in deze regellussen geen onzekerheden zijn. De prestaties van de voorgestelde regelaar is vergeleken met bestaande regelaars, zoals niet-lineaire model inversie met modelidentificatie (NDI-MI). De validatie-resultaten tonen aan dat de INDI regelaar het referentie vliegpad goed kan volgen, zonder dat daar een parameter aanpassingsalgoritme of een modelidentificatie voor nodig is. Bovendien is er geen extra inspanning vereist om een aerodynamische modelstructuur te ontwerpen.

Om simultaan sensor- en aandrijvings-FTC te bewerkstelligen stelt dit proefschrift een FTC systeem voor dat bestaat uit een FDD systeem en een herconfigureerbare regelaar. Moderne FDD methodes gebruiken informatie van de stuurorgaanaandrijving, zoals bijvoorbeeld het aansturingssignaal voor een stuurvlak of de gemeten uitslag van het stuurvlak. Dit kan echter problematisch zijn wanneer de stuurorgaanaandrijving (deels) kapot is. Daarom wordt in dit proefschrift het simultaan sensor- en aandrijvings-FTC probleem opgelost zonder gebruik te maken van informatie van de stuurorgaanaandrijving. De fouten in de aandrijving van de stuurorganen worden gecompenseerd door de herconfigureerbare regelaar. Hiermee is simultaan sensor- en aandrijvings-FTC gerealiseerd, zelfs bij fouten in de aandrijving van de stuurorganen. De prestaties van

het voorgestelde FTC systeem zijn gevalideerd met verschillende foutscenario's, waarin zowel fouten in de sensoren als fouten in de stuurorgaanaandrijving voorkomen.

Meer studies naar sensor FDD onder invloed van meerdere foutscenario's worden aanbevolen. Deze studies zullen de prestaties van de voorgestelde methodes bij onbekende fouttypes verder kunnen demonstreren. Met betrekking tot verder onderzoek op het gebied van FTC systemen wordt aanbevolen om deze toe te passen in vliegtuigen, aangezien dit nog steeds een uitdaging is, maar tegelijk belangrijk is om de veiligheid van vliegen te verhogen.

ACRONYMS

ABS Adaptive Backstepping. viii, 8, 184, 188

ADDSAFE Advanced Fault Diagnosis for Sustainable Flight Guidance and Control. 76, 110, 126

ADIRU Air Data Inertial Reference Unit. 1, 3

ADS Air Data Sensor. viii, ix, xxv–xxix, 1, 6, 11–13, 15, 19–25, 27, 28, 31, 32, 34, 35, 37–47, 50–53, 62–64, 66, 67, 70, 72, 73, 75, 76, 78, 85–89, 91–95, 99, 101–106, 109, 113, 153, 172, 173, 175, 177, 179, 180, 210–214, 219, 220, 224, 227–230, 250, 251

AEKF Augmented Extended Kalman Filter. 110, 114, 116

AFKF Adaptive Fading Kalman Filter. 6

AFTC Active Fault-Tolerant Control. vii, 6

AHRS Attitude Heading and Reference System. xxix, 211–214, 219, 220, 224

AKF Adaptive Kalman Filter. 6

AMMAE Augmented Multiple-Model Adaptive Estimation. xxvi, 47, 53–56, 58, 60, 62–64, 68, 69

ANDI Adaptive Nonlinear Dynamic Inversion. 8

AOA angle of attack. 1–3

AS-KM airspeed-based kinematic model. viii, ix, 13, 127, 128, 130, 131, 133, 140–144, 146–151

ATS-UKF Adaptive Three-Step Unscented Kalman Filter. ix, xxv, 13, 20–22, 25, 32–43, 211, 213, 222, 224, 228

ATSEKF Adaptive Two-Stage Extended Kalman Filter. ix, 13, 15, 127, 128, 133, 134, 136, 137, 140–143, 146, 148, 228

AUKF Augmented Unscented Kalman Filter. 64, 65

BS Backstepping. viii, 8, 12, 184, 210, 218

DMAE Double-Model Adaptive Estimation. ix, xxvii, xxviii, 13, 15, 76–78, 80–82, 84–101, 103–106, 153, 155–164, 166–173, 175–180, 228–230

DMAE-NSR Double-Model Adaptive Estimation-No Selective Reinitialization. 84, 87–91

EKF Extended Kalman Filter. 6, 25, 53

FCC Flight Control Computer. 1

FDD Fault Detection and Diagnosis. vii–x, xxv–xxviii, 2–6, 8–13, 15, 19–25, 31–34, 36, 38, 39, 41, 43, 45–47, 49–70, 72, 73, 75–78, 80–83, 85, 86, 89, 91–95, 98–101, 103–106, 109, 110, 125, 153, 172, 173, 175, 177, 179, 180, 183, 209–211, 218–222, 224, 227–231

FDI Fault Detection and Isolation. 20, 32, 33, 41, 76, 92, 126, 127

FTC Fault-Tolerant Control. vii, viii, x, xxv, xxix, 2, 3, 6, 8–10, 12, 13, 15, 43, 73, 123, 136–138, 183, 185, 209–211, 215, 218–225, 227, 229–231

GPS Global Positioning System. 99, 113, 119, 123

GS-KM ground speed-based kinematic model. ix, 15, 127–131, 136, 137, 140–144, 146–151

IBS Incremental Backstepping. x, 210, 211, 215, 216, 218, 220–224, 229

IMM Interacting Multiple-Model. xxvi, 6, 46, 47, 50, 70–73, 76, 77, 81, 82, 87, 92

IMU Inertial Measurement Unit. viii, ix, xxvi–xxix, 11, 13, 15, 22, 24, 45–47, 50–58, 60–62, 70, 72, 73, 78, 86, 87, 98–100, 102, 109–114, 119, 122, 123, 125–129, 131, 133, 135–137, 139, 141, 153, 212, 227, 228, 248–250

INDI Incremental Nonlinear Dynamic Inversion. x, 184, 185, 196, 198–200, 202, 203, 205, 207, 208, 229

IOTSEKF Iterated Optimal Two-Stage Extended Kalman Filter. ix, xxvii, 13, 15, 110, 111, 113–117, 119, 123, 127, 134–136, 228

KF Kalman Filter. 6, 53, 79, 80, 84, 96, 116, 156, 157

KM kinematic model. xxvii, 110–113, 115, 117, 119, 120, 122, 123, 127, 128, 136, 140, 143, 148, 151

LPV Linear Parameter Varying. 5, 6, 210

LTI Linear Time-Invariant. 5, 6

MM multiple model. 76, 77, 106

MMAE Multiple-Model Adaptive Estimation. ix, xxvi, xxvii, 6, 12, 13, 46–51, 53, 54, 68, 73, 76, 77, 79–82, 85, 87, 92, 228, 230

NDI Nonlinear Dynamic Inversion. viii, x, 8, 12, 184, 185, 189, 207

NDI-MI Nonlinear Dynamic Inversion with online Model Identification. x, xxix, 185, 196, 198–201, 203–207

OTSEKF Optimal Two-Stage Extended Kalman Filter. ix, 13, 110, 113, 114, 116, 123, 228

OTSKF Optimal Two-Stage Kalman Filter. 155, 169

PF Particle Filter. 6

PFTC Passive Fault-Tolerant Control. vii, 6

PTU Port Transducer Unit. 1, 2

RECONFIGURE REconfiguration of CONtrol in Flight for Integral Global Upset REcovery. 126

RMSE root mean square error. 96, 120, 122, 123, 147–151, 169, 171, 200, 203

RTS-KF Robust Three-Step Kalman Filter. 13, 20, 21, 25, 43, 155, 169

RTS-UKF Robust Three-Step Unscented Kalman Filter. ix, 20, 21, 25, 27–34, 39–43, 228

SMO Sliding Mode Observer. 5, 126, 128, 140, 143

SR Selective Reinitialization. ix, 12, 47, 50, 54, 59, 60, 62, 68, 72, 73, 76, 77, 80, 82–84, 87, 90, 106

SRMMAE Selective-Reinitialization Multiple-Model Adaptive Estimation. ix, xxvi, 12, 13, 46, 47, 50, 54, 55, 57–70, 72, 73, 82, 87, 88, 90, 92, 228

TSKF Two-Stage Kalman Filter. 6

UAV Unmanned Aerial Vehicle. vii, 4, 20, 111, 230

UKF Unscented Kalman Filter. 6, 25, 27, 32–34, 47, 53, 55, 56, 58–60, 62, 64, 65, 78, 81, 96, 97, 102, 213

UMMAE Unscented Multiple-Model Adaptive Estimation. xxvi, 47, 53, 55, 56, 58, 60, 62, 63

List of symbols

Symbols

A_x, A_y, A_z	specific forces along the body axis, m/s^2
A_{xm}, A_{ym}, A_{zm}	measurements of accelerometers along the body axis, m/s^2
X, Y, Z	aerodynamic forces including thrust forces along the body axis, N
L, M, N	aerodynamic moments including thrust moments along the body axis, N
$\bar{q}$	dynamic pressure, Pa
S	wing surface area, m/s^2
$\bar{c}$	mean aerodynamic chord, m
b	wing span, m
V	true airspeed, m/s
α	angle of attack, rad
β	angle of sideslip, rad
p, q, r	roll, pitch and yaw rates along the body axis, rad/s
ϕ, θ, ψ	Euler angles angles along the body axis, rad

V_m, α_m, β_m	air data sensor measurements
α_{vm}	angle of attack measurements by the vane, rad
β_{vm}	angle of sideslip measurements by the vane, rad
p_m, q_m, r_m	measurements of roll, pitch and yaw rates along the body axis, rad/s
ϕ_m, θ_m, ψ_m	measurements of roll, pitch and yaw angles along the body axis, rad
$\boldsymbol{f}_i, \boldsymbol{f}_o$	faults of the input and output
f_V, f_α, f_β	faults in the air data sensors
$f_{A_x}, f_{A_y}, f_{A_z}$	faults in the accelerometers, m/s^2
f_p, f_q, f_r	faults in the rate gyros, rad/s
f_ϕ, f_θ, f_ψ	faults in the attitude heading and reference system, rad
$\boldsymbol{\gamma}$	innovation of the filter
$\hat{\boldsymbol{x}}$	state estimate
$\tilde{\boldsymbol{x}}$	state estimation error
$\mathbf{P}$	covariance of state estimation error
$\boldsymbol{p}_t$	model probability of all the elemental filters
$\hat{\boldsymbol{x}}_t$	state estimate of all the elemental filters
$\mathbf{P}_t$	covariance of state estimate error of all the elemental filters
a, a_i	fault scenarios and the ith fault scenario of the MMAE
p_i	model probability of the ith elemental filter of the MMAE
p_{nf}	model probability of the no-fault filter
$\hat{\boldsymbol{x}}_{nf}$	state estimate of the no-fault filter
$\mathbf{P}_{nf}$	covariance of state estimate error of the no-fault filter
p_{af}	model probability of the fault filter
$\hat{\boldsymbol{x}}_{af}$	state estimate of the fault filter
$\mathbf{P}_{af}$	covariance of state estimate error of the fault filter

$\hat{\boldsymbol{f}}$	fault estimation of the fault filter
$\bar{\boldsymbol{f}}$	probability-weighted fault estimation of the fault filter
$\hat{\boldsymbol{f}}_i$	fault estimation of the ith elemental filter
$\bar{\boldsymbol{f}}_i$	probability-weighted fault estimation of the ith elemental filter
$\hat{\boldsymbol{f}}_i^o$	estimation of the output fault of the ith elemental filter
$\bar{\boldsymbol{f}}_i^o$	probability-weighted output fault estimation of the ith elemental filter
i_{max}	index of the model with the maximum model probability
i_{min}	index of the model with the minimum model probability
u_a, v_a, w_a	airspeed velocity components along the body axis, m/s
u_w, v_w, w_w	wind velocity components along the body axis, m/s
u_{GS}^B, v_{GS}^B, w_{GS}^B	ground speed velocity components along the body axis, m/s
U_N, U_E, U_D	ground speed components in the local navigation frame, m/s
x, y, z	position coordinates in the earth fixed reference frame, m
χ	kinematic azimuth angle, rad
γ	flight path angle, rad
μ	kinematic roll angle, rad
PLA	power level angle, rad
$\mathbf{u}$, $\Delta\mathbf{u}$	input and incremental input
$\Delta\delta_a$, $\Delta\delta_e$, $\Delta\delta_r$	incremental deflections of the aileron, elevator and rudder, rad
δ_a, δ_e, δ_r	deflections of the ailerons, elevators and rudders, rad
δ_{ar}, δ_{rl}	deflections of the right aileron and lower rudder, rad
δ_{al}, δ_{ru}	deflections of the left aileron and upper rudder, rad

SUPERSCRIPTS

des	desired value generated by the controller
ref	reference value given to the controller
exp	expected value

SUBSCRIPTS

i, j, l	variable number
k	time step
ff	fault free
af	augmented fault
m	measured variable

CONTENTS

1

INTRODUCTION

1.1. AIRLINER ACCIDENTS DUE TO SENSOR AND ACTUATOR FAULTS

SAFETY is of paramount importance in the aviation community, especially for civil aircraft. In civil aviation, many developments focus on further improving aircraft safety and reliability levels. Although most modern aircraft are equipped with fault detection and protection systems, accidents due to sensor and actuator failures still occur.

Consider for example the Air Data Sensors (ADSs) which measure the key air data parameters of the aircraft, such as airspeed and angle of attack (AOA). Failures of the ADSs can result in serious accidents, such as with the Airbus A330 flight AF 447 [1]. Here, the airspeed indicators were incorrect, presumably caused by the obstruction of the pitot probes by ice crystals, which led the flight control law to change from normal law to alternate law [1]. In normal law, the flight envelope is protected whereas the protections are lost in alternate law [1]. The pilots failed to control the aircraft flight path correctly, resulting in a stall situation which led the aircraft to collapse into the Atlantic ocean.

Since the design of aircraft has become more complex, fault modes which had not been previously encountered can occur. For instance, the in-flight upset of an Airbus 330-303 aircraft (Figure 1.1(a)), operated as Qantas flight 72, was caused by multiple spikes in the AOA data coming from one of the three Air Data Inertial Reference Units (ADIRUs), a very rare and specific situation [2]. These spikes in the data caused the primary Flight Control Computers (FCCs) to command the aircraft to rapidly pitch down, which resulted in at least 110 of the 303 passengers and nine of the 12 crew members being injured and 12 of the occupants being seriously injured. After the investigation of the accident, the aircraft manufacturer became aware of the problem and improved the FCC design, to make sure that such faults will not occur again.

Several accidents involving military aircraft were also caused by ADS faults. On 23 February 2008, a B-2A bomber departed Andersen AFB Guam and crashed 17 seconds after takeoff [3]. The aircraft was destroyed by fire after impact, both pilots successfully ejected. It was found that during the air data calibration procedure, three Port Trans-

1

(a) VH-QPA Qantas Airbus A330-303, ©Lance C Broad

(b) El Al Boeing 747-200F REG 4X-AXG, the aircraft involved, in Amsterdam in June 1991, ©Werner Fischdick

Figure 1.1: Aircraft accidents due to sensor and actuator faults.

ducer Units (PTUs) contained moisture and required a bias correction that was larger than necessary. The inappropriate bias values were given to the PTUs during the air data system calibration. The air data system failure was not detected until the moment aircraft reached the weight-off-wheels state. At this exact moment, an excessive negative AOA value was calculated, which resulted in the aircraft to pitch up in such a rapid way the pilots were unable to override in time [3].

Apart from the accidents caused by aircraft sensor faults, accidents also happened because of aircraft actuator faults. An infamous example was the El Al Cargo Flight LY 1862, a Boeing 747-200F freighter aircraft (Figure 1.1(b)), which departed from Amsterdam airport on a flight towards Tel Aviv on October 4, 1992. While the aircraft was climbing, engine number 3 had a failure, which also impacted engine number 4. These damages resulted in further loss of all outboard ailerons, the outboard flaps, spoilers, the inner left and outer right elevators. Although the crew attempted to make an emergence landing, the aircraft crashed due to the serious damage, killing all four on-board and thirty-nine people on the ground [4].

Not only a partial loss of hydraulics can lead to disasters, malfunctions of hydraulics can also result in severe accidents. An United Airlines Boeing 737-200, operating as flight 585, rolled steadily to the right and pitched nose down until it reached a nearly vertical attitude. The cause of the accident was probably a loss of control resulting from the movement of the rudder surface to its blowdown limit [5]. A similar accident occurred to USAir flight 427, which is a Boeing 737-3B7 aircraft [6].

Recent studies show that several accidents could have been avoided if a certain type of Fault-Tolerant Control (FTC) system was implemented on-board [7]. This motivates the research on FTC and Fault Detection and Diagnosis (FDD) systems which aim at further increasing the safety of the aircraft.

1.2. FAULT DIAGNOSIS AND FAULT-TOLERANT CONTROL

Presently, most aircraft are equipped with certain types of FDD, and FTC systems. One type of sensor and actuator FDD and FTC system can be given in Figure 1.2. In this

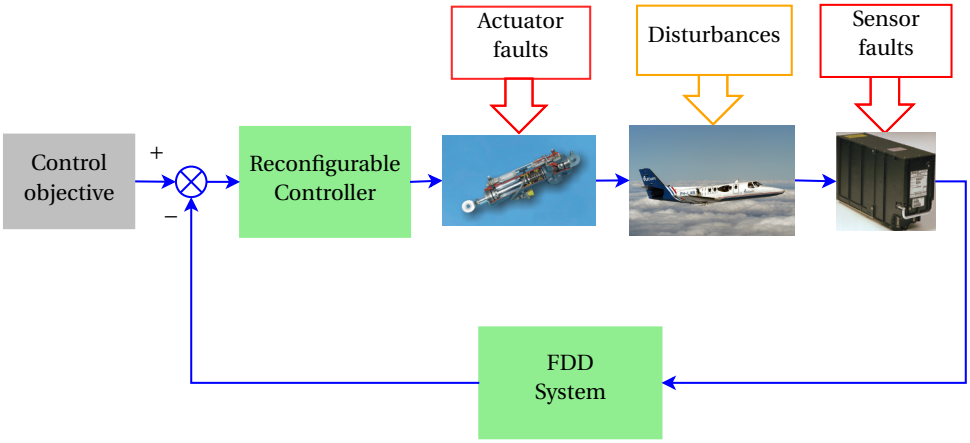


Figure 1.2: Block diagram of the aircraft sensor and actuator FDD and FTC system.

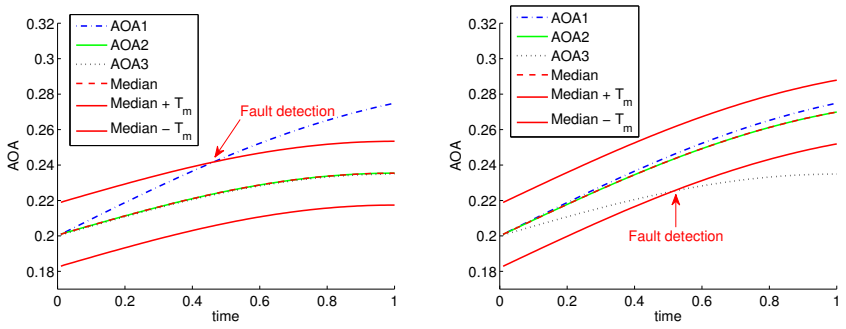
section, an overview of various FDD and FTC approaches is presented, together with their advantages and drawbacks.

1.2.1. FAULT DETECTION AND DIAGNOSIS

An FDD system performs fault detection, fault isolation and fault identification (estimation or reconstruction) [8]. The objective of fault detection is to detect the occurrence of a fault [8]. To determine which sensor or which actuator fails is called fault isolation [8]. Fault identification is essential since it estimates the magnitude of the fault which is important to achieve FTC.

Regarding aircraft sensor and actuator FDD, the current practice in the industry is hardware redundancy [9], which uses multiple parallel sensors fulfilling the same functions. For example, three ADIRUs provide three independent AOA values, denoted as AOA1, AOA2 and AOA3. The median value of these three values can be used for consistency checks [2]. In the normal case, all three AOA values are consistent. If AOA1 continues to differ from the median value by more than a predefined threshold denoted as T_m, as shown in Figure 1.3(a), it will be declared as faulty and it will not be used for subsequent calculations of the AOA values. It should be noted that the logic shown in the figure only serves as an example showing how hardware redundancy works. Real implementations of the checking logic are much more comprehensive and complex.

However, there are also limitations associated with this checking scheme. For instance, in the case shown in Figure 1.3(b), two faulty sensors, AOA1 and AOA2, behave similarly such that the third correct one (AOA3), which deviates from the median value by more than T_m, is rejected and declared to be faulty. Although the possibility of this situation to occur is extremely low (smaller than 10^{-9} per flight hour [2]), it could lead to hazardous situations. Another possible limitation is that if one of the sensors (such as AOA1) has been declared to be faulty, it is more difficult to detect a fault in the remaining two sensors (AOA2 and AOA3) since there are only two measurements and the system cannot determine which one to trust.



(a) Case when AOA1 fails. AOA3 is the true AOA value.

(b) Case when both AOA1 and AOA2 fail. AOA3 is the true AOA value.

Figure 1.3: Different cases when one or two sensors fail.

All of these limitations have led to the development of analytical redundancy (also referred to as model-based FDD [8]) [10], a method which uses a mathematical model of the monitored process rather than using multiple sensors and actuators with equal functions. The analytical redundancy method reduces weight and cost for the design, which is useful especially for small Unmanned Aerial Vehicles (UAVs) with limited payload.

Consider a nonlinear discrete-time dynamic system model of the form:

$$\begin{cases} \boldsymbol{x}_k = \bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \mathbf{E}_{k-1}\boldsymbol{d}_{k-1} + \mathbf{G}_{k-1}\boldsymbol{w}_{k-1} & (1.1) \\ \boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \mathbf{F}_k\boldsymbol{f}_k + \boldsymbol{v}_k & (1.2) \end{cases}$$

where $\boldsymbol{x}$ is the state vector, $\boldsymbol{y}$ is the measurement vector, $\boldsymbol{d}$ is the external disturbance vector, $\boldsymbol{f}$ is the fault vector, the subscript k represents the time step. The internal disturbances $\boldsymbol{w}$ and $\boldsymbol{v}$ are the process noise and measurement noise vectors, respectively, with assumption $E[\boldsymbol{w}] = \mathbf{0}$ and $E[\boldsymbol{v}] = \mathbf{0}$. $\bar{\boldsymbol{f}}$ and $\boldsymbol{h}$ are nonlinear functions; $\Delta\bar{\boldsymbol{f}}$ represents model uncertainties in $\bar{\boldsymbol{f}}$. One way to detect the fault is to monitor the residual (innovation for Kalman filter-based approaches) $\boldsymbol{\gamma}_k$ which is the difference between the measured output $\boldsymbol{y}_k$ and the estimated output $\hat{\boldsymbol{y}}_k$:

$$\begin{aligned} \boldsymbol{\gamma}_k &= \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k & (1.3) \\ &= \boldsymbol{h}(\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \mathbf{E}_{k-1}\boldsymbol{d}_{k-1} + \mathbf{G}_{k-1}\boldsymbol{w}_{k-1}) + \mathbf{F}_k\boldsymbol{f}_k + \boldsymbol{v}_k - \boldsymbol{h}(\bar{\boldsymbol{f}}(\hat{\boldsymbol{x}}_{k-1})) & (1.4) \end{aligned}$$

where $\hat{\boldsymbol{x}}_{k-1}$ is the estimate of $\boldsymbol{x}_{k-1}$ at time step $k-1$. Assume that the state estimate is unbiased and $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) = \mathbf{0}, \boldsymbol{d}_{k-1} = \mathbf{0}$, the expectation of $\boldsymbol{\gamma}_k$ is then:

$$E[\boldsymbol{\gamma}_k] = \mathbf{F}_k\boldsymbol{f}_k \quad (1.5)$$

For the fault-free case ($\boldsymbol{f}_k = \mathbf{0}$), $\boldsymbol{\gamma}_k$ should be zero-mean. In the case of faults, $\boldsymbol{\gamma}_k$ will deviate from zero and the faults can be detected if a residual evaluation function $J(\boldsymbol{\gamma}_k)$ exceeds a threshold. There are many ways to design the residual evaluation function and thresholds [8]. As an example, a norm of $\boldsymbol{\gamma}_k$ can be used as the residual evaluation function and a constant value can be used as the threshold [8].

If only the state estimate is unbiased, $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) \neq \boldsymbol{0}$, $\boldsymbol{d}_{k-1} \neq \boldsymbol{0}$, then the expectation of $\boldsymbol{\gamma}_k$ is:

$$E[\boldsymbol{\gamma}_k] = \mathbf{F}_k \boldsymbol{f}_k + \boldsymbol{h}(\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1}) + \mathbf{E}_{k-1}\boldsymbol{d}_{k-1}) - \boldsymbol{h}(\bar{\boldsymbol{f}}(\hat{\boldsymbol{x}}_{k-1})). \tag{1.6}$$

It is seen that the expectation can also deviate from zero and exceed a threshold due to the presence of $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1})$ and $\boldsymbol{d}_{k-1}$. In this case, since the model uncertainties $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1})$ and external disturbance $\boldsymbol{d}_{k-1}$ are both unknown, detecting the presence of faults or extracting fault information using $\boldsymbol{\gamma}_k$ is challenging for model-based FDD approaches. In addition, $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$ can further influence the performance of approaches such as observers, which makes the detection of faults even more challenging. This challenge motivates the design of robust FDD approaches. An FDD approach is robust if it is insensitive or even invariant to model uncertainties $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1})$, internal disturbances $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$, and external disturbances $\boldsymbol{d}_{k-1}$.

In the past few decades, many model-based FDD approaches have been proposed to achieve sensor or actuator FDD [10–14]. A brief overview of these approaches will be presented in the following. It should be noted that this overview is by no means complete, it merely provides examples of different approaches together with their drawbacks and advantages.

OBSERVERS

Observer-based approaches were commonly-used in the 1980s [8]. They are used to generate residuals which can be used for fault diagnosis [8, 12]. To cope with unknown inputs, such as external disturbances $\boldsymbol{d}_{k-1}$ in Equation (1.1), unknown input observers are proposed [8, 15]. The principle is to make the state estimation error decoupled from the influence of disturbances [8]. A lot of contributions have been made to the development of fault detection using the unknown input observers [8, 10]. Unknown input observers are also applied to the aircraft sensor and actuator FDD problem [16, 17].

Sliding Mode Observers (SMOs) can also generate residuals and can be used for sensor and actuator fault detection for a class of uncertain systems [18]. By maintaining the sliding mode on the residual in the presence of faults, sensor and actuator faults are reconstructed using a so-called equivalent output injection signal which represents the effort necessary to maintain the sliding mode [19]. Applications of SMOs to the aircraft sensor and actuator FDD problem can also be found [20].

These observers are usually designed based on Linear Time-Invariant (LTI) systems. To extend to nonlinear and time-varying systems, normally a Linear Parameter Varying (LPV) model is needed. Another limitation of these observers is that they are suitable for deterministic systems ($\boldsymbol{w} = \boldsymbol{v} = \boldsymbol{0}$). For stochastic systems with noise ($\boldsymbol{w} \neq \boldsymbol{0}$, $\boldsymbol{v} \neq \boldsymbol{0}$), the following approaches have been developed.

H_∞ FILTERS

The goal of the FDD using H_∞ optimization methods is to make the generated residual more robust to modeling uncertainties $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1})$ and external disturbances $\boldsymbol{d}_{k-1}$ (both in Equation (1.1)) and more sensitive to faults $\boldsymbol{f}_k$ (in Equation (1.2)) [8, 21]. Fault detection can be achieved by factorization [21], standard H_∞ filtering approaches [22, 23]. The optimization problem can be solved using linear matrix inequality approaches [24].

H_∞ filtering approaches have also been used to detect and estimate aircraft sensor or actuator faults [23, 25]. Marcos et al. [23] designed five H_∞ filters to detect and isolate the aircraft sensor faults. Freeman et al. [25] also designed two H_∞ filters to detect and estimate the faults in the ADS. These methods are designed based on LTI systems. LPV systems should be used for FDD of nonlinear, time-varying systems.

KALMAN FILTERS

The Kalman Filter (KF), the optimal linear filter, was proposed in 1960 [26] and later extended to the Extended Kalman Filter (EKF) for nonlinear systems. Because the EKF linearizes the nonlinear systems and thus can introduce linearization errors, its use could lead to filter divergence. In 1997, an Unscented Kalman Filter (UKF) was proposed which reduces the linearization errors by using sigma point approximations [27]. Later, the KF was further extended to Particle Filter (PF) [28] for system with non-Gaussian noise.

The history of using KF for FDD can be traced back to the 1970s [29]. To cope with unknown inputs such as faults and disturbances, many KF variations have been proposed, such as Adaptive Fading Kalman Filters (AFKFs) [30], Adaptive Kalman Filters (AKFs) [31] and Two-Stage Kalman Filters (TSKFs) [32, 33]. There are also multiple-model-based KFs, which are composed of a bank of KFs. The multiple-model-based KFs include Multiple-Model Adaptive Estimation (MMAE) [34] and Interacting Multiple-Model (IMM) [35]. These methods can all be used for FDD. Ample applications of these methods on aircraft sensor and actuator FDD can be found in the literature [36–38]. More information on various adaptive KFs can be found in Chapters 2 to 7 of this dissertation.

Although numerous model-based aircraft sensor and actuator FDD techniques are proposed, the real application on large fixed-wing aircraft is still quite uncommon. One example is the actuator fault detection system in the A380, which detects oscillatory failure modes in the actuators [39].

1.2.2. FAULT-TOLERANT CONTROL

The aforementioned FDD is important to achieve sensor FTC. However, to deal with actuator faults, not all FTC systems include an FDD system. Generally speaking, there are two types of actuator FTC systems: Passive Fault-Tolerant Control (PFTC) and Active Fault-Tolerant Control (AFTC) [40]. For PFTC, the controller is fixed and FDD is not required [40]. Therefore, it has limited fault-tolerant capabilities [40]. In contrast, AFTC makes use of the FDD information and reconfigures the controller based on the information of the faults, such that a safe flight can still be maintained.

For both FTC types, a reconfigurable controller (as shown in Figure 1.2) is necessary, especially when the considered faults are actuator faults. The objectives of designing a reconfigurable controller is to achieve certain satisfactory performance not only when there are no faults in the system but also in cases when there are sensor or actuator faults [40]. In cases such as actuator faults, reconfigurable controllers achieve control objectives by using remaining control authority to compensate for the failed control surfaces. A short introduction of several reconfigurable control techniques is given below. For a more general overview and also comparisons among different techniques involved, the reader is referred to survey papers [11, 40]. It should be noted that the following recon-

figurable control approaches mainly deal with actuator faults.

MODEL PREDICTIVE CONTROL

Model predictive control is an optimal control technique which can handle system input and output constraints [41]. It is proposed for control reconfiguration mainly because it can systematically handle changing model dynamics and constraints [42]. To deal with faults such as actuator faults, a fault model of the actuator fault is required [42]. In case of an actuator fault, the model predictive controller can handle it by changing the actuator limits and rate constraints [42, 43]. However, it requires a computationally intensive online optimization at each time step, which can limit its application in real-life such as in flight control.

ROBUST H_∞ CONTROL

Robust control can deal with various types of uncertainties [44]. The H_∞ controller is designed for worst-case uncertainty cases. Since the design of robust controllers has already taken actuator faults into consideration, it works in nominal as well as in faulty situations [44]. As such, it can suffer from performance degradation in nominal conditions, when there are no faults, because performance in nominal conditions is sacrificed to guarantee robustness. Furthermore, the order of the final designed controller is usually higher than the system order due to the inclusion of weights [45]. Consequently, model reduction is required to reduce the order of the controller [45]. Different design methods are available such as mixed sensitivity, loop shaping and μ synthesis [44]. Examples of an H_∞ controller design in the presence of actuator faults can be found in [46].

SLIDING MODE CONTROL

Sliding mode control has become a popular technique in recent years, due to its fast convergence property and its robustness to model uncertainties [18]. Sliding mode control contains linear sliding mode control and terminal sliding mode control [47]. In [48], a boundary layer adaptation scheme is proposed to handle partial actuator faults. This adaptation scheme can also prevent actuators from magnitude and rate saturation [48]. In [49], it is argued that sliding mode control is inherently robust to partial actuator faults. Despite recent developments of sliding mode controllers, they still suffer from the chattering problem (high frequency oscillations in the system states) [18]. A number of solutions are proposed to reduce the effects of the chattering [18, 47, 50–52]. Applications of sliding mode controllers for aircraft can be found in [48, 49].

ADAPTIVE CONTROL

Adaptive Control is a control technique which can adjust the controller parameters online to deal with faults. Two examples of adaptive controllers are:

- L_1 adaptive control
  L_1 adaptive control, which was recently proposed, has guaranteed tracking performance by increasing the adaptation rate [53, 54]. The tracking performance of the L_1 adaptive controller when dealing with systems with uncertainties and disturbances is also guaranteed [55]. Therefore, it can also be applied to deal with

actuator faults. Some applications of the L_1 adaptive controller can be found in [56, 57].

- Adaptive Nonlinear Dynamic Inversion (NDI) and Backstepping (BS)
  NDI or Feedback linearization, and BS [58] are model inversion-based control techniques. These techniques are suitable for nonlinear systems due to the fact that they make explicit use of an accurate model of the controlled system. Therefore, gain scheduling is not required. However, these methods are also sensitive to model uncertainties and can fail in the presence of actuator faults. To cope with that, numerous adaptive NDI and BS approaches are proposed, for example Adaptive Nonlinear Dynamic Inversion (ANDI) with online model identification [59], Adaptive Backstepping (ABS) [60] and Command Filter ABS [61–63].

Applications of FTC in real aircraft already exist. In 1999, the F-18E/F was delivered with an FTC system which can compensate for a single stabilizer actuator fault [63].

1.3. CHALLENGES IN THE FIELD OF AIRCRAFT SENSOR AND ACTUATOR FDD AND FTC

As mentioned in the previous section, although extensive research has been devoted to the sensor and actuator FDD and FTC, there are only few real applications. This is mainly because there are still a number of challenges which have not been completely solved. The main challenges related to the FDD and FTC approaches are real-life model uncertainties $\Delta\bar{\boldsymbol{f}}(\boldsymbol{x}_{k-1})$, effects of disturbances including external disturbances $\boldsymbol{d}_{k-1}$ (such as time-varying wind and atmospheric turbulence) and internal disturbances ($\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$), and computational constraints.

Take the air velocities for example, the derivatives and measurements can be given as follows:

$$\begin{cases} \dot{\boldsymbol{V}}_a = -\boldsymbol{\omega} \times \boldsymbol{V}_a - \dot{\boldsymbol{V}}_w - \boldsymbol{\omega} \times \boldsymbol{V}_w + \mathbf{T}_{be}\boldsymbol{g} + \begin{bmatrix} X/m \\ Y/m \\ Z/m \end{bmatrix} & (1.7) \\ \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{V}_a) + \boldsymbol{f}_{V_a} + \boldsymbol{\nu}_{V_a} & (1.8) \end{cases}$$

with

$$\boldsymbol{V}_a = [u_a, v_a, w_a]^T,\ \boldsymbol{\omega} = [p, q, r]^T, \boldsymbol{V}_w = [u_w, v_w, w_w]^T, \tag{1.9}$$

$$\boldsymbol{y} = [V_m, \alpha_m, \beta_m]^T, \boldsymbol{f}_{V_a} = [f_V, f_\alpha, f_\beta]^T, \boldsymbol{\nu}_{V_a} = [\nu_V, \nu_\alpha, \nu_\beta]^T \tag{1.10}$$

where u_a, v_a and w_a are the air velocity components expressed along the body axis, p, q and r are the body angular rates, u_w, v_w and w_w are the wind velocity components expressed along the body axis, V_m is the measured true airspeed, α_m and β_m are the measured angle of attack and sideslip. $\boldsymbol{f}_{V_a}$ and $\boldsymbol{\nu}_{V_a}$ represent the faults and noise in the measurements $\boldsymbol{y}$. X, Y and Z are total aerodynamic forces including propulsion forces along body axis. $\mathbf{T}_{be}$ is the transformation matrix from earth axis to body axis.

Most FDD and FTC approaches make use of the aircraft aerodynamic model which calculates the aerodynamic forces and moments. For instance, the forces X, Y and Z can be expressed as the following Taylor series (higher-order terms are neglected for simplicity of explanation):

$$X = \bar{q}SC_X = \bar{q}S(C_{X_0} + C_{X_\alpha}\alpha + C_{X_q}q + C_{X_{\delta_e}}\delta_e + C_{X_{PLA}}PLA) \tag{1.11}$$

$$Y = \bar{q}SC_Y = \bar{q}S(C_{Y_0} + C_{Y_\beta}\beta + C_{Y_p}\frac{pb}{2V} + C_{Y_r}\frac{rb}{2V} + C_{Y_{\delta_a}}\delta_a + C_{Y_{\delta_r}}\delta_r) \tag{1.12}$$

$$Z = \bar{q}SC_Z = \bar{q}S(C_{Z_0} + C_{Z_\alpha}\alpha + C_{Z_q}\frac{q\bar{c}}{V} + C_{Z_{\delta_e}}\delta_e + C_{Z_{PLA}}PLA) \tag{1.13}$$

where $\bar{q}$ is the dynamic pressure, S the surface area, b and $\bar{c}$ are the wing span and mean aerodynamic chord. PLA is the power level angle, δ_a, δ_e and δ_r are the deflections of aileron, elevator and rudder. The derivatives $C_{X_{PLA}}, C_{X_{\delta_e}}, C_{Y_{\delta_a}}, C_{Y_{\delta_r}}, C_{Z_{\delta_e}}, C_{Z_{PLA}}$ are control derivatives. Other derivatives are stability derivatives. These derivatives are time-varying according to flight conditions. Computational fluid dynamics, wind tunnel tests and flight tests are usually performed to obtain these parameters. However, accurate values of these derivatives are difficult to obtain, which results in model uncertainties. It becomes even more challenging as the structure of modern aircraft is becoming more complex such as those with flexible wings. The influence of model uncertainties on FDD can be seen in Equation (1.4). Model uncertainties can also cause the residual to deviate from zero when there are no faults, which can result in false alarms (here the FDD system provides an alarm when there are in fact no faults occurring). To reduce the influence, much research effort is focused on increasing the robustness of the FDD and FTC approaches with respect to model uncertainties [8].

Real-life external disturbances can severely degrade the performance of FDD and FTC approaches. In Equation (1.7), $\dot{\boldsymbol{V}}_w + \boldsymbol{\omega} \times \boldsymbol{V}_w$ represents the additional accelerations generated by the wind. The influence of the disturbances on FDD can be readily explained using Equation (1.4). The faults and disturbances can both influence $\boldsymbol{\gamma}_k$ and it is difficult to distinguish between the fault and disturbances. Such disturbances are unavoidable and can occur during any part of the flight. Time-varying wind and atmospheric disturbances, such as turbulence, can influence the motion of the aircraft and affect the performance of FDD and FTC approaches. Internal disturbance, sensor noise, is unavoidable as well and must be taken into consideration.

Hence, the robustness with respect to these internal and external disturbances must be taken into consideration. One of the main functions of a control system is to suppress the influence caused by such disturbances ($\boldsymbol{d}_{k-1}$, $\boldsymbol{w}_{k-1}$ and $\boldsymbol{v}_k$). For FDD, these disturbances can make the residual deviate from zero even when there are no faults. When using a fixed threshold, false alarms can occur if the threshold is low. If the threshold is chosen too high, the false alarm rate can be reduced, at the cost, however, of missed detections [8]. Finding the right balance between the rate of false alarms and possible occurrences of missed detection is a challenge for FDD.

Computational burden also limits the implementation of many FDD and FTC algorithms. A computationally intensive program may not be able to run in real-time at a high update rate, which will degrade its performance. Therefore, the development of advanced FDD and FTC approaches must take the computational load into account.

1

1.4. MAIN RESEARCH QUESTION

The challenges mentioned in the previous section motivate the main research question of this dissertation:

Research question

How to increase the robustness of the sensor and actuator FDD and FTC approaches with respect to real-life model uncertainties and internal and external disturbances?

By robustness, we mean that the performance of the FDD and FTC approaches is insensitive, or even invariant, to model uncertainties and internal and external disturbances. Only when the FDD and FTC approaches are robust to real-life model uncertainties and disturbances, can they be applied in practice to truly enhance the safety and survivability of the aircraft.

In this thesis, also some efforts are made to reduce the computational burden of the algorithms; however, it is at present considered of secondary importance. With the rapid development of more advanced computers, algorithms with intensive computational load are expected to become feasible in the next decade. The research subject presented in this dissertation is fixed-wing aircraft, despite the fact that several approaches have also been applied to rotary-wing aircraft.

1.5. RESEARCH GOALS AND APPROACH

The following goals are formulated, together with feasible approaches to address the main research question in a systematic fashion.

1.5.1. ROBUST SENSOR FDD

The models proposed for sensor FDD in this dissertation differ from the models used by existing approaches. As mentioned, existing approaches primarily use the aerodynamic model which calculates the aerodynamic forces and moments. The forces are given in Equations (1.11)-(1.13) and contain model uncertainties. In addition, to compute the forces, the control surface deflections δ_a, δ_e and δ_r are needed. The block diagram for this existing sensor FDD approach is given in Figure 1.4(a).

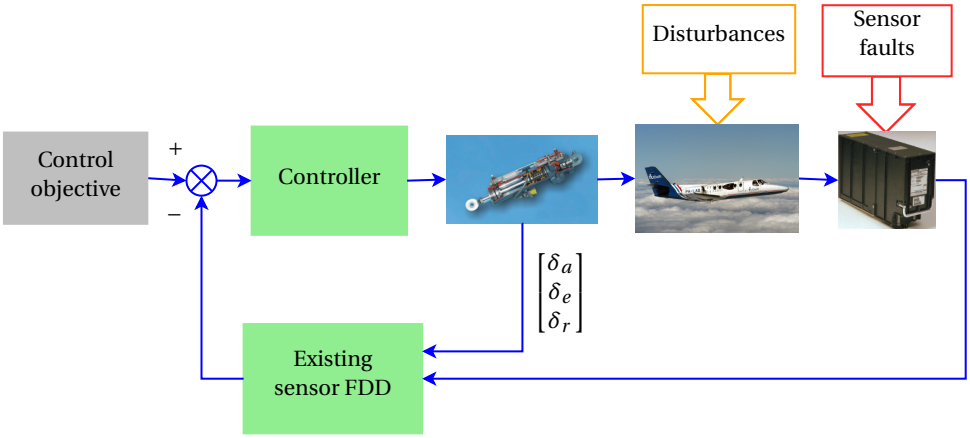
The approach proposed in this dissertation for sensor FDD is to use the kinematic models of the aircraft. By replacing Equations(1.11)-(1.13) with Equations(1.14)-(1.16), the forces X, Y and Z are directly obtained by the measurements of the accelerometers (A_{xm}, A_{ym} and A_{zm}).

$$X = mA_{xm} \tag{1.14}$$

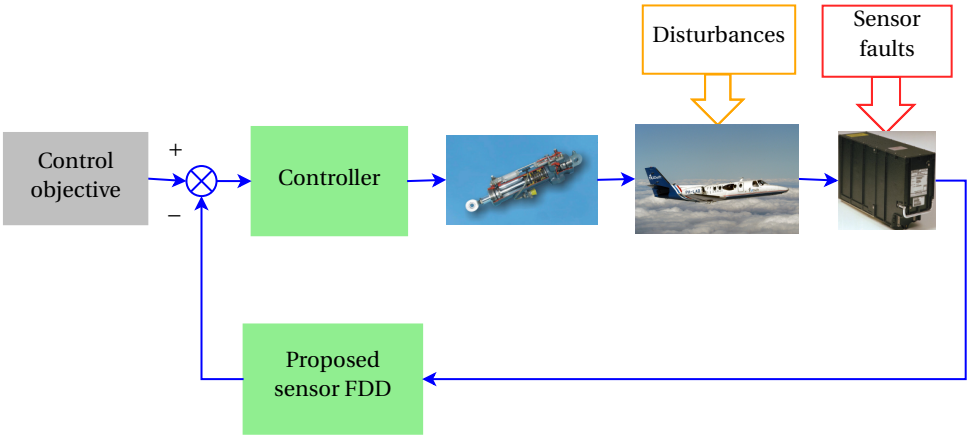
$$Y = mA_{ym} \tag{1.15}$$

$$Z = mA_{zm} \tag{1.16}$$

The accelerometers measure the specific forces (the difference between the inertial acceleration and the gravitational acceleration assuming the accelerometers are at the center of gravity of the aircraft [64]). Equations (1.7)-(1.8) together with (1.14)-(1.16) compose the *air velocity-based kinematic model*. Through doing this, model uncertainties



(a) Existing sensor FDD using aerodynamic model.



(b) Sensor FDD using kinematic model in this dissertation.

Figure 1.4: Aircraft sensor FDD block diagrams using aerodynamic model and kinematic model.

are reduced due to the measurements from the accelerometers. In addition, the control surface deflections δ_a, δ_e and δ_r are not needed. In other words, the input of the aircraft dynamic model does not consist of δ_a, δ_e and δ_r, as in many conventional FDD techniques. Rather, we propose to use the measurements A_{xm}, A_{ym} and A_{zm}, as illustrated in Figure 1.4(b). The control surface deflections are then no longer required by the sensor FDD. To give an example, the input and outputs when using the kinematic model for ADS FDD are shown in Figure 1.5. The input consists of the Inertial Measurement Unit (IMU) measurements (measurements from the accelerometers and rate gyros).

It is also noted in the figure that a filtering block is included in the block diagram. The kinematic models only provide a model, which contains fewer model uncertainties, for the FDD. The detection and diagnosis of the faults as well as state estimation still have to

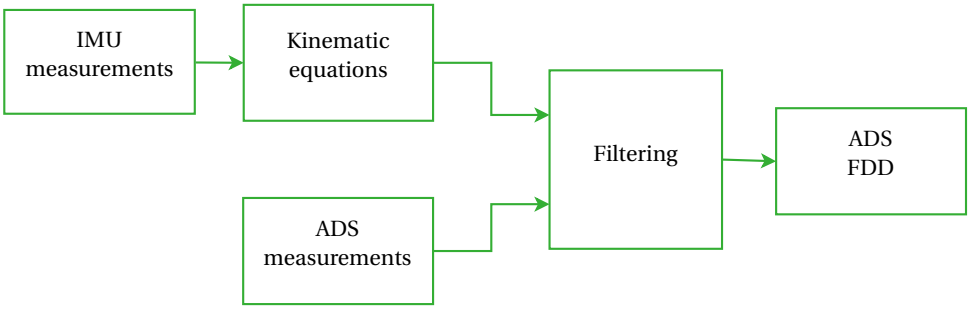


Figure 1.5: ADS FDD using the kinematic model.

be solved. In this dissertation, several advanced filtering techniques are developed and improved exactly for this purpose.

1.5.2. ROBUST ACTUATOR FTC

As for actuator FTC, reconfigurable controllers are required. NDI and BS are reconfigurable control approaches which can deal with nonlinearities in the system. However, they also suffer from performance degradation in the presence of either model uncertainties or actuator faults. In this thesis, the incremental form of these two approaches is applied, to deal with aircraft trajectory and attitude control in the presence of model uncertainties and actuator faults. Aircraft trajectory control is especially important for air traffic control, and challenging due to the model uncertainties in both the flight path control loop and the angular rate control loop. In this dissertation, a new control structure is proposed to solve trajectory control.

1.5.3. SIMULTANEOUS SENSOR AND ACTUATOR FTC

As seen in Figure 1.4(a), existing sensor FDD approaches rely on the information of the commanded or measured control surface deflections δ_a, δ_e and δ_r. However, things become more complicated when there are no direct measurements of control surface deflections available as well as sensor and actuator faults occur simultaneously. When there are simultaneous sensor and actuator faults, the input and output of the sensor FDD system using existing approaches (Figure 1.4(a)) are both incorrect, and sensor FDD is difficult to achieve. To achieve simultaneous sensor and actuator FTC, this dissertation explores a framework which deals with simultaneous sensor and actuator FTC.

1.6. RESEARCH CONTRIBUTIONS

The main contributions of this dissertation can be generalized as follows.

First of all, several improvements regarding advanced filtering techniques are made. These are listed as follows:

- A Selective-Reinitialization Multiple-Model Adaptive Estimation (SRMMAE) is proposed which significantly improves the performance of normal MMAE. The SRMMAE contains three Selective Reinitialization (SR) algorithms which remove the

drawbacks of the MMAE with respect to FDD. However, the computational load of the SRMMAE is still intensive. This leads to the following proposed filter.

- A novel Double-Model Adaptive Estimation (DMAE) approach is developed which reduces the number of the model set required by the MMAE to only two. It reduces the computational load and design complexity, especially when dealing with simultaneous faults. It also decreases the false alarms when using the MMAE or SRMMAE.

- Furthermore, state-of-the-art approaches on unknown input Kalman filters require that an existence condition is satisfied (physically it means that the influence of disturbances and faults can be decoupled). In this dissertation, the DMAE is further extended to deal with a case when the existence condition is not satisfied.

- Other improvements include the Adaptive Two-Stage Extended Kalman Filter (ATSEKF) which can adaptively estimate the faults and Adaptive Three-Step Unscented Kalman Filter (ATS-UKF) which extends the Robust Three-Step Kalman Filter (RTSKF) to nonlinear systems and is robust to initial condition errors. In addition, an Iterated Optimal Two-Stage Extended Kalman Filter (IOTSEKF) is proposed to reduce the sensitivity of the Optimal Two-Stage Extended Kalman Filter (OTSEKF) to initial condition errors.

Secondly, IMU and ADS sensor faults are considered. The sensor FDD is tackled by reducing the influence of model uncertainties as well as disturbances. More importantly, the proposed sensor FDD approaches are validated by real flight test data which contain real-life model uncertainties. Real flight test data recorded in the presence of unknown disturbances such as time-varying wind and turbulence are also applied to validate the performance of the proposed techniques.

Finally, this thesis proposes a strategy to achieve simultaneous sensor and actuator FTC. The key point is to achieve the sensor fault diagnosis without using any actuator information. To be more specific: to perform the sensor FDD without any knowledge regarding either the commanded or measured control surface deflections. This is very important since, if we succeed, we can guarantee that even when there are actuator failures, sensor fault diagnosis can still be achieved.

1.7. OUTLINE OF THE THESIS

The dissertation consists of three parts, see Figure 1.6.

In Part I, the aircraft sensor FDD in the presence of model uncertainties is treated. It consists of three chapters: Chapters 2, 3 and 4. Chapter 2 deals with ADS FDD. The influence of model uncertainties is reduced by using the airspeed-based kinematic model (AS-KM) of the aircraft. Chapter 3 treats both IMU and ADS FDD using an improved MMAE approach: the SRMMAE method. Although the SRMMAE can achieve a better FDD performance, the computational load is intensive especially when dealing with simultaneous faults. To overcome this limitation while maintaining satisfactory performance, a novel DMAE approach is proposed in Chapter 4, for the ADS FDD. Computational load is reduced and the false alarms are also reduced.

1

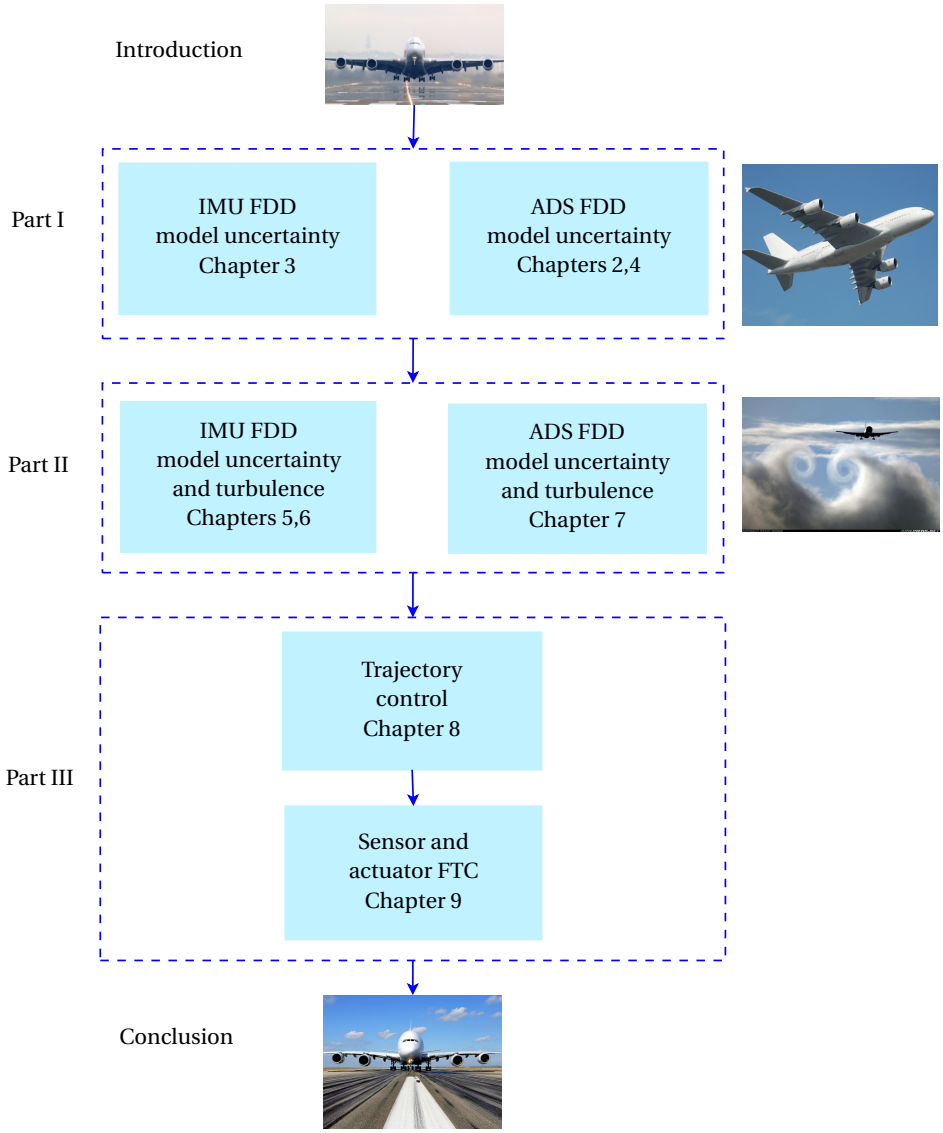


Figure 1.6: Visual outline of the dissertation.

Part II considers the sensor FDD in the presence of both model uncertainties and external as well as internal disturbances. It consists of three chapters: Chapters 5, 6 and 7. Chapter 5 proposes to deal with the IMU FDD in the presence of model uncertainties and disturbances by exploiting the ground speed-based kinematic model (GS-KM) of the aircraft. An IOTSEKF is proposed to estimate both the state and the faults. Performance is validated by real flight test data. However, the real flight tests were performed while avoiding atmospheric disturbances. Therefore, aircraft response data during the presence of turbulence was not collected. More flight tests were performed to gather aircraft response data in the presence of turbulence and the validation results using these data are included in Chapter 6. In addition, since the IOTSEKF used in Chapter 5 suffers from performance degradation when covariance matrices are not chosen wisely, a novel ATSEKF is proposed in Chapter 6, a method that adaptively updates the covariance matrices. Chapter 7 discusses a novel DMAE approach which is extended to solve the ADS FDD in the presence of model uncertainties and disturbances.

In Part III, simultaneous sensor and actuator FTC is considered in two chapters. Chapter 8 deals with aircraft trajectory control in the presence of model uncertainties and actuator faults. Finally, Chapter 9 proposes a new FTC framework which can cope with the case when sensor and actuator faults occur simultaneously.

1.8. THESIS PUBLICATIONS

In this section, the publication sources for each chapters are listed:

- Chapter 2 is based on the following publication: P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, Q.P. Chu. Adaptive Three-Step Kalman Filter for Air Data Sensor Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics*, 39(3), 590-604, 2016.

- Chapter 3 is based on the following publication:

  P. Lu, L. Van Eykeren, E. van Kampen, Q.P. Chu. Selective-Reinitialization Multiple Model Adaptive Estimation for Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics*, 38(8), 1409-1424, 2015.

- Chapter 4 is based on the following publication:

  P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, Q.P. Chu. Double-model Adaptive Fault Detection and Diagnosis Applied to Real Flight Data. *Control Engineering Practice*, (36), 39-57, 2015.

- Chapter 5 is based on the following publication:

  P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, Q.P. Chu. Aircraft Inertial Measurement Unit Fault Identification with Application to Real Flight Data. *Journal of Guidance, Control and Dynamics*, 38(12), 2467-2475, 2015.

- Chapter 6 is based on the following publication:

  P. Lu, E. van Kampen, C. C. de Visser, Q.P. Chu. Nonlinear Aircraft Sensor Fault Reconstruction In the Presence of Disturbances Validated by Real Flight Data. *Control Engineering Practice*, (49), 112-128, 2016.

- Chapter 7 is based on the following publication:

  P. Lu, E. van Kampen, C. C. de Visser, Q.P. Chu. Novel Framework to State and Unknown Input Estimation for Time-varying Systems. *Automatica*, 2016, http://dx.doi.org/10.1016/j.automatica.2016.07.009.

- Chapter 8 is based on the following publication:

  P. Lu, E. van Kampen, C. C. de Visser, Q.P. Chu. Nonlinear Aircraft Fault-Tolerant Trajectory Control Using Incremental Nonlinear Dynamic Inversion. *Journal of Guidance, Control and Dynamics*, (under review).

- Chapter 9 is based on the following publication:

  P. Lu, E. van Kampen, C. C. de Visser, Q.P. Chu. Nonlinear Simultaneous Aircraft Sensor and Actuator Fault-Tolerant Flight Control. *Journal of Guidance, Control and Dynamics*, (under review).

I

SENSOR FAULT DETECTION AND DIAGNOSIS IN THE PRESENCE OF MODEL UNCERTAINTIES

2

Adaptive Three-Step Kalman Filter for Air Data Sensor Fault Detection and Diagnosis

In this chapter, the aircraft ADS FDD in the presence of model uncertainties is considered. The model used for FDD is the airspeed-based kinematic model. First, the kinematic model including ADS faults is introduced. Then, the problem of the existing approach for state and fault estimation (Robust Three-Step Unscented Kalman Filter) is analyzed and modified, which leads to a novel state and fault estimation approach (Adaptive Three-Step Unscented Kalman Filter). The Adaptive Three-Step Unscented Kalman Filter is applied to the ADS FDD using the airspeed-based kinematic model. Simulation and real flight test data validations demonstrate the satisfactory performance of the proposed approach.

Parts of this chapter are based on:

P Lu, L Van Eykeren, E van Kampen, C C de Visser, Q P Chu. Adaptive Three-Step Kalman Filter for Air Data Sensor Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics.* available online. [65]

ADS FDD is important for the safety of aircraft. In this chapter, first an extension of the RTS-KF to nonlinear systems is made by proposing a Robust Three-Step Unscented Kalman Filter (RTS-UKF). The RTS-UKF is found to be sensitive to the initial condition error when dealing with ADS fault estimation. A theoretical analysis of this sensitivity is presented and a novel ATS-UKF is proposed which is able to cope with not only the estimation of the ADS faults but also the detection and isolation of faults. The ATS-UKF contains three steps: time update, fault estimation and measurement update. This approach can reduce the sensitivity to the initial condition error. Finally, the ADS FDD performance of the ATS-UKF is validated using simulated aircraft data. Additionally, its performance is further validated using real flight test data to demonstrate its performance under realistic uncertainties and disturbances. The results using both the simulated data and real flight test data demonstrate the satisfactory FDD performance of the ATS-UKF and verify that it can be applied in practice to enhance the safety of aircraft.

2.1. INTRODUCTION

PRESENTLY, Fault Detection and Isolation (FDI) has an important role in achieving fault-tolerance of aircraft [11]. During the past few decades, many approaches have been proposed for sensor or actuator FDI [12, 66, 67]. In aerospace engineering, the FDI of sensors and actuators for fixed-wing aircraft is widely studied, as can be found in Patton [11], Marzat et. al [13], and Hajiyev and Caliskan [68]. Investigation of the FDI for Unmanned Aerial Vehicles can also be found [13, 69]. For recent advances, the reader is referred to Goupil [9] and Zolghadri [14, 70]. The ADSs measure the dynamic pressure, airspeed, angle of attack and angle of sideslip of the aircraft, providing essential information on the aircraft states to the pilot [71]. The ADSs are usually installed outside the aircraft fuselage and can suffer from icing or water accumulation, which may result in faults such as blockage faults [25]. These faults may negatively influence the information provided to the pilot, which can lead to catastrophic accidents. In the recent past, there have been commercial aircraft accidents caused by ADS faults. Due to faults in the ADSs, the flight crew of Austral Lineas Aeroeas Flight 2553 improperly referenced the airspeed indicator and induced a structure failure by exceeding safe airspeed limits [72]. More recently, the final report of the Air France Flight 447 accident stated that erroneous airspeed measurements from the pitot probes were a contributing factor [73]. Since 2003, commercial aircraft have had more than 35 recorded incidents of multiple ADS faults [72]. There have also been accidents of military aircraft caused by ADS faults. The crash of a B-2 bomber is due to a large bias to the ADSs which is caused by moisture in the port transducer units [73]. These facts indicates the importance of the fault detection of the ADSs.

The fault detection of ADSs has been investigated in a number of studies [25, 74]. Some researchers propose to use alternative air data sensing systems such as a flush air-data sensing system [74, 75]. Nebula et. al propose a virtual air data system against ADS failures [76, 77]. Looye and Joos [78] propose to use the data from a navigation system to determine the air data information. On the other hand, the faults of the ADSs can be detected. Houck and Atlas [71] are one of the first to analyze ADS faults. The limitation of their approach is that independent static pressure measurements are not always available in UAV applications [72]. Cervia et al. [79] and Eubank et al. [72] de-

tect the faults using a multiple-redundancy air data system. The air data system studied by Cervia et al. is based on pseudo-quadruplex redundancy which employs four self-aligning air data probes. Freeman et al. [25] investigate analytical redundancy instead of hardware redundancy for the ADS fault detection. They use a longitudinal dynamics model of the aircraft and two linear H_∞ filters are designed to detect the faults and provide robustness to model errors.

Alternatively, the kinematic model can be used to detect the faults in the ADSs, thereby reducing the influence of model uncertainties caused by the calculation of the aerodynamic forces and moments [80, 81]. Van Eykeren and Chu [82] use an adaptive Extended Kalman Filter to detect the faults in the ADSs. However, the estimation of the faults is not addressed in their work. In Lu et al. [83], a Selective-Reinitialization Multiple-Model Adaptive Estimation approach is proposed for the ADS FDD. The approach improved the FDD performance of the Multiple-Model-based approaches. However, the computational load of the approach is intensive when dealing with simultaneous faults.

In this chapter, a newly-developed RTS-KF [84] is combined with the kinematic model to estimate the ADS faults. First, the RTS-KF is extended to cope with nonlinear systems by proposing a novel RTS-UKF. The RTS-UKF is able to reduce linearization errors. However, it is found that the RTS-UKF is sensitive to the initial condition error. Second, the sensitivity of this three-step Kalman Filter to the initial condition error is analyzed theoretically. It is proved that the RTS-UKF does not use some of the measurements to update the state estimation which causes the sensitivity to the initial condition error.

Finally, a novel ATS-UKF is proposed which does not only estimate the ADS faults, but also detect and isolate the faults. The ATS-UKF contains three steps: time update, fault estimation and measurement update. The fault detection is performed before the fault estimation. This approach also reduces its sensitivity to the initial condition. The fault detection is performed by checking the innovation variances. In the presence of faults, the innovation variance increases. If the innovation variance exceeds a pre-defined threshold, then the fault alarm is triggered. The FDD performance of the ATS-UKF is tested using simulated aircraft data with the objective of detecting, isolating and estimating ADS faults. Two different fault scenarios (multiple faults and simultaneous faults) are implemented to test the performance and the results demonstrate the satisfactory performance of the ATS-UKF. The fault types contain not only bias and drift fault, but also oscillatory faults.

Furthermore, the FDD performance of the ATS-UKF is validated using real flight test data of a Cessna Citation II aircraft. The sensor measurements from the real flight test contain biases and uncertainties and are suitable for testing the performance of the ATS-UKF. Different fault scenarios are generated and the faults are injected into the real flight data. The ADS FDD results of the ATS-UKF demonstrate its performance and verified that it can be applied in practice to enhance the safety of the aircraft.

The structure of the chapter is as follows: In Section 2.2, the ADS FDD problem is formulated. The kinematic model including ADS faults is introduced. Section 2.3 extends the RTS-KF to cope with nonlinear systems by proposing the RTS-UKF. The RTS-UKF is applied to estimate the ADS faults, which turns out to be sensitive to the initial condition. The sensitivity problem is analyzed theoretically and a novel ATS-UKF is proposed to deal with not only the estimation of the ADS faults, but also the detection and isola-

tion of the faults. The performance is tested using a simulated aircraft model. In Section 2.4, the performance of the ATS-UKF is further validated using the real flight data of the Cessna Citation II aircraft. The performance is shown and some remarks are given. Finally, the conclusions are made in Section 2.5.

2

2.2. AIR DATA SENSOR FDD USING THE KINEMATIC MODEL

The objective of this chapter is the FDD of the aircraft ADSs. However, model-based approaches are sensitive to model uncertainties. In order to make the proposed approach more robust, the kinematic model of aircraft, which does not involve the computation of aerodynamic forces and moments, is used instead of the aerodynamic model.

2.2.1. AIRCRAFT KINEMATIC MODEL WITH ADS FAULTS

The kinematic model of the aircraft including ADS faults is described as

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) \tag{2.1}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \boldsymbol{v}(t) + \mathbf{F}(t)\boldsymbol{f}^o(t), \quad t = t_i,\ i = 1, 2, \ldots \tag{2.2}$$

where $\boldsymbol{x} \in \mathbb{R}^L$ represents the system states, $\boldsymbol{u}_m \in \mathbb{R}^l$ the measured input, $\boldsymbol{y} \in \mathbb{R}^m$ the measurement. The functions $\bar{\boldsymbol{f}}$ and $\boldsymbol{h}$ are nonlinear functions. $\mathbf{G}$ and $\mathbf{F}$ are the noise distribution matrix and output fault distribution matrix. The function $\boldsymbol{f} \in \mathbb{R}^p$ represents output faults.

The system equation variables are defined as follows:

$$\boldsymbol{x} = [V\ \alpha\ \beta\ \phi\ \theta\ \psi]^T \tag{2.3}$$

$$\boldsymbol{u}_m = [A_{xm}\ A_{ym}\ A_{zm}\ p_m\ q_m\ r_m\ ]^T = [A_x\ A_y\ A_z\ p\ q\ r\ ]^T + \boldsymbol{w} \tag{2.4}$$

$$\boldsymbol{y} = [V_m\ \alpha_m\ \beta_m\ \phi_m\ \theta_m\ \psi_m]^T \tag{2.5}$$

$$\boldsymbol{w} = [w_{Ax}\ w_{Ay}\ w_{Az}\ w_p\ w_q\ w_r\ ]^T \tag{2.6}$$

$$\boldsymbol{v} = [v_V\ v_\alpha\ v_\beta\ v_\phi\ v_\theta\ v_\psi]^T \tag{2.7}$$

$$\boldsymbol{f}^o = [f_V\ f_\alpha\ f_\beta]^T \tag{2.8}$$

where the input $\boldsymbol{u}_m$ is the IMU measurement which measures the specific forces (A_x, A_y and A_z) and angular rates (roll rate p, pitch rate q, and yaw rate r) of the aircraft. y is the output measurement which measures the air data information (true airspeed V, angle of attack α, and angle of sideslip β) and Euler angles (roll angle ϕ, pitch angle θ, and yaw angle ψ). $[f_V\ f_\alpha\ f_\beta]^T$ are the faults of the ADSs, i.e. f_V, f_α and f_β are the faults in the velocity sensor, angle of attack sensor, and angle of sideslip sensor, respectively. It is assumed that there are no faults in the IMU and Attitude and Heading Reference System which measures the Euler angles and the influence of changing wind such as turbulence is limited. Therefore, the input noise vector $\boldsymbol{w}(t)$ can be assumed to be a continuous time white noise process while the output noise vector $\boldsymbol{v}(t)$ can be assumed to be a discrete

time noise sequence.

$$E[\boldsymbol{w}(t)] = \mathbf{0}$$

$$E[\boldsymbol{w}(t)\boldsymbol{w}^T(t_\tau)] = \mathbf{Q}\delta(t-\tau)\ ,\ \mathbf{Q} = \text{diag}(\sigma^2_{w_{Ax}}, \sigma^2_{w_{Ay}}, \sigma^2_{w_{Az}}, \sigma^2_{w_p}, \sigma^2_{w_q}, \sigma^2_{w_r}), \tag{2.9}$$

$$E[\boldsymbol{v}(t)] = \mathbf{0}$$

$$E[\boldsymbol{v}(t_i)\boldsymbol{v}^T(t_j)] = \mathbf{R}\delta(t_i - t_j)\ ,\ \mathbf{R} = \text{diag}(\sigma^2_{v_V}, \sigma^2_{v_\alpha}, \sigma^2_{v_\beta}, \sigma^2_{v_\phi}, \sigma^2_{v_\theta}, \sigma^2_{v_\psi}), \tag{2.10}$$

$$E[\boldsymbol{w}(t)\boldsymbol{v}^T(t_i)] = \mathbf{0}\ ,\ t = t_i\ ,\ i = 1,2,\ldots \tag{2.11}$$

The kinematic model is given in the following [82, 83]. For detailed derivations, please refer to Appendix B.

$$\begin{aligned}\dot{V} = {} & (A_{xm} - w_{Ax} - g\sin\theta)\cos\alpha\cos\beta + (A_{ym} - w_{Ay} + g\sin\phi\cos\theta)\sin\beta \\ & + (A_{zm} - w_{Az} + g\cos\phi\cos\theta)\sin\alpha\cos\beta\end{aligned} \tag{2.12}$$

$$\begin{aligned}\dot{\alpha} = {} & \frac{1}{V\cos\beta}\big[-(A_{xm} - w_{Ax})\sin\alpha + (A_{zm} - w_{Az})\cos\alpha + g\cos\phi\cos\theta\cos\alpha \\ & + g\sin\theta\sin\alpha\big] + q_m - w_q - [(p_m - w_p)\cos\alpha + (r_m - w_r)\sin\alpha]\tan\beta\end{aligned} \tag{2.13}$$

$$\begin{aligned}\dot{\beta} = {} & \frac{1}{V}\big[-(A_{xm} - w_{Ax} - g\sin\theta)\cos\alpha\sin\beta + (A_{ym} - w_{Ay} + g\sin\phi\cos\theta)\cos\beta \\ & - (A_{zm} - w_{Az} + g\cos\phi\cos\theta)\sin\alpha\sin\beta\big] + (p_m - w_p)\sin\alpha - (r_m - w_r)\cos\alpha\end{aligned} \tag{2.14}$$

$$\dot{\phi} = (p_m - w_p) + (q_m - w_q)\sin\phi\tan\theta + (r_m - w_r)\cos\phi\tan\theta \tag{2.15}$$

$$\dot{\theta} = (q_m - w_q)\cos\phi - (r_m - w_r)\sin\phi \tag{2.16}$$

$$\dot{\psi} = (q_m - w_q)\frac{\sin\phi}{\cos\theta} + (r_m - w_r)\frac{\cos\phi}{\cos\theta} \tag{2.17}$$

and $\mathbf{G}(\boldsymbol{x}(t))$ is defined as:

$$\mathbf{G}(\boldsymbol{x}(t)) = \begin{bmatrix} -c\alpha c\beta & -s\alpha c\beta & -s\alpha c\beta & 0 & 0 & 0 \\ s\alpha/(Vc\beta) & 0 & -c\alpha/(Vc\beta) & c\alpha t\beta & -1 & s\alpha t\beta \\ c\alpha s\beta/V & -c\beta/V & s\alpha s\beta/V & -s\alpha & 0 & c\alpha \\ 0 & 0 & 0 & -1 & -s\phi t\theta & -c\phi t\theta \\ 0 & 0 & 0 & 0 & -c\phi & s\phi \\ 0 & 0 & 0 & 0 & -s\phi/c\theta & -c\phi/c\theta \end{bmatrix} \tag{2.18}$$

where s(•), c(•) and t(•) denote the trigonometric functions sin(•), cos(•) and tan(•) respectively. Therefore, the measurement model including the ADS faults is

$$V_m = V + f_V + v_V \tag{2.19}$$

$$\alpha_m = \alpha + f_\alpha + v_\alpha \tag{2.20}$$

$$\beta_m = \beta + f_\beta + v_\beta \tag{2.21}$$

$$\phi_m = \phi + v_\phi \tag{2.22}$$

$$\theta_m = \theta + v_\theta \tag{2.23}$$

$$\psi_m = \psi + v_\psi \tag{2.24}$$

Table 2.1: Fault scenario of multiple faults

Time interval	Sensor	Fault type	Fault magnitude	Fault unit
$10\text{ s} < t < 20\text{ s}$	V	bias	2	[m/s]
$30\text{ s} < t < 40\text{ s}$	α	drift	$0.01t$	[rad/s]
$50\text{ s} < t < 60\text{ s}$	β	oscillatory	$-2\pi\sin(\pi t)/180$	[rad]

Table 2.2: Fault scenario of simultaneous faults

Time interval	Sensor	Fault type	Fault magnitude	Fault unit
$10\text{ s} < t < 20\text{ s}$	V	oscillatory	$2\sin(\pi t)$	[m/s]
	α	drift	$0.01t$	[rad/s]
	β	drift	$-0.01t$	[rad/s]
$30\text{ s} < t < 40\text{ s}$	V	drift	$-0.2t$	[m/s^2]
	α	bias	$-2\pi/180$	[rad]
	β	oscillatory	$-2\pi\sin(\pi t)/180$	[rad]

The measurement model can be rewritten into

$$\boldsymbol{y}(t) = \boldsymbol{x}(t) + \mathbf{F}(t)\boldsymbol{f}^o(t) + \boldsymbol{v}(t), \qquad t = t_i, i = 1, 2, \ldots \tag{2.25}$$

where

$$\mathbf{F} = [\mathbf{I}_3\ \mathbf{0}_{3\times 3}]^T \tag{2.26}$$

The objective of the ADS FDD problem is to detect, isolate and estimate $\boldsymbol{f}^o = [f_V\ f_\alpha\ f_\beta]^T$. This chapter assumes that there are no faults in the IMU sensors. If there are faults in the IMU sensors, they can be detected and estimated by other methods using another set of kinematic model [85].

2.2.2. FAULT SCENARIOS FOR THE ADS FDD

In this chapter, two different fault scenarios are used to test the performance of the approaches. The fault scenario for multiple ADS faults is given in Table 2.1 while that for simultaneous ADS faults is given in Table 2.2. The fault type, magnitude and unit are given in the table. The units of the drift faults are given by the units of the drift rates. It can be seen that the fault types not only contains bias faults but also drift faults and oscillatory faults.

2.2.3. STATE OBSERVABILITY AND FAULT RECONSTRUCTIBILITY

This section check the observability of the system described by Equations (2.1) and (2.2). The observability analysis of the system can be performed by checking the rank of the

following observability matrix:

$$\mathbf{O} = \begin{bmatrix} \delta_{\boldsymbol{x}} \boldsymbol{h} \\ \delta_{\boldsymbol{x}} (L_{\bar{\boldsymbol{f}}} \boldsymbol{h}) \\ \vdots \\ \delta_{\boldsymbol{x}} (L_{\bar{\boldsymbol{f}}}^{L-1} \boldsymbol{h}) \end{bmatrix} \tag{2.27}$$

where $\delta_{\boldsymbol{x}} \boldsymbol{h}$ denotes the gradient of $\boldsymbol{h}$ with respect to $\boldsymbol{x}$. The Lie derivative is defined as follows:

$$\begin{aligned} L_{\bar{\boldsymbol{f}}} \boldsymbol{h} &= \delta_{\boldsymbol{x}} \boldsymbol{h} \cdot \bar{\boldsymbol{f}} \\ &\vdots \\ L_{\bar{\boldsymbol{f}}}^{L-1} \boldsymbol{h} &= \delta_{\boldsymbol{x}} (L_{\bar{\boldsymbol{f}}}^{L-2} \boldsymbol{h}) \cdot \bar{\boldsymbol{f}} \end{aligned} \tag{2.28}$$

It can be readily checked that $\mathbf{O}$ is of full rank. Therefore, the system state is observable. In order to reconstruct the faults, additional conditions are required which are given in (2.29).

2.3. EXTENSION OF THE ROBUST THREE-STEP KALMAN FILTER

This section extends the RTS-KF to estimate output faults. First, in Section 2.3.1, the RTS-KF is extended to nonlinear systems by proposing a RTS-UKF. This RTS-UKF is applied to the ADS fault estimation problem and is found to be sensitive to the initial condition errors. This sensitivity problem is analyzed theoretically in Section 2.3.2. Then, in Section 2.3.3, an ATS-UKF is proposed which can detect, isolate and estimate the faults. Finally, the ATS-UKF is applied to the ADS FDD problem in Section 2.3.4 to demonstrate its FDD performance.

2.3.1. ROBUST THREE-STEP UNSCENTED KALMAN FILTER

The RTS-KF [84] can be used for output FDD. Consider the aircraft kinematic model described by Equations (2.1) and (2.2). For this system, since the system state is observable, the existence condition of a RTS-KF is [84]:

$$m \geq p, \qquad \text{rank}\, \mathbf{F}_k = p \tag{2.29}$$

In this study, $m = 6$, $p = 3$ and rank $\mathbf{F}_k = 3$. Therefore, a RTS-KF can be designed to estimate the ADS faults.

However, the RTS-KF is designed for linear systems while the kinematic model is nonlinear. Therefore, the RTS-KF needs to be extended to cope with nonlinear systems. The UKF is a nonlinear filter which can achieve a better level of accuracy than the EKF [27, 86]. This section extends the RTS-KF to nonlinear systems by proposing a RTS-UKF.

According to the technique in Lu et al. [81], the RTS-UKF can be derived as follows:

2

Step1 Sigma points calculation and time update

$$\mathscr{X}_{0,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} \tag{2.30a}$$

$$\mathscr{X}_{i,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} - \left(\sqrt{(L+\gamma_0)\mathbf{P}_{k-1|k-1}}\right)_i, \quad i = 1,2,...,L \tag{2.30b}$$

$$\mathscr{X}_{i,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} + \left(\sqrt{(L+\gamma_0)\mathbf{P}_{k-1|k-1}}\right)_{i-L}, \quad i = L+1, L+2,...,2L \tag{2.30c}$$

$$w_0^{(m)} = \gamma_0/(L+\gamma_0) \tag{2.31a}$$

$$w_0^{(c)} = \gamma_0/(L+\gamma_0) + (1-\alpha_0^2+\beta_0) \tag{2.31b}$$

$$w_i^{(m)} = w_i^{(c)} = 1/\{2(L+\gamma_0)\}, \quad i = 1,2,...,2L \tag{2.31c}$$

with $\mathscr{X}_{i,k-1}$ the sigma points of the states (dimension L) at step $k-1$. $w_i^{(m)}$ and $w_i^{(c)}$ are the weights associated with the ith point with respect to $\hat{\boldsymbol{x}}_{k-1|k-1}$ and $\mathbf{P}_{k-1|k-1}$, respectively. $\gamma_0 = \alpha_0^2(L+\kappa) - L$ is a scaling factor, α_0 determines the spread of the sigma points around $\hat{\boldsymbol{x}}_{k-1|k-1}$, κ is a secondary scaling factor, β_0 is used to incorporate the prior knowledge of the distribution of x. In this chapter, $\kappa = 0$, $\alpha_0 = 0.8$ and $\beta_0 = 2$.

After the creation of the sigma points through the nonlinear transformation, the predicted mean and covariance are computed as follows

$$\mathscr{X}_{i,k|k-1} = \mathscr{X}_{i,k-1} + \int_{t_{k-1}}^{t_k} \bar{\boldsymbol{f}}(\mathscr{X}_{i,k-1}, \boldsymbol{u}(\tau), \tau)\, d\tau \tag{2.32}$$

$$\hat{\boldsymbol{x}}_{k|k-1} = \sum_{i=0}^{2L} w_i^{(m)} \mathscr{X}_{i,k|k-1} \tag{2.33}$$

$$\mathbf{P}_{k|k-1} = \sum_{i=0}^{2L} w_i^{(c)} [\mathscr{X}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\mathscr{X}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}]^T + \mathbf{Q}_d \tag{2.34}$$

$$\mathscr{X}^*_{i,k|k-1} = [\mathscr{X}_{0:2L,k|k-1} \quad \mathscr{X}_{0,k|k-1} - \nu\sqrt{\mathbf{Q}_d} \quad \mathscr{X}_{0,k|k-1} + \nu\sqrt{\mathbf{Q}_d}]_i \tag{2.35}$$

$$\mathscr{Y}^*_{i,k|k-1} = \boldsymbol{h}(\mathscr{X}^*_{i,k|k-1}) \tag{2.36}$$

$$\hat{\mathbf{y}}_k = \sum_{i=0}^{2L^a} w_i^{*(m)} \mathscr{Y}^*_{i,k|k-1} \tag{2.37}$$

$$\mathbf{P}_{xy,k} = \sum_{i=0}^{2L^a} w_i^{*(c)} [\mathscr{X}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\mathscr{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k]^T \tag{2.38}$$

$$\mathbf{P}_{yy,k} = \sum_{i=0}^{2L^a} w_i^{*(c)} [\mathscr{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k][\mathscr{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k]^T + \mathbf{R} \tag{2.39}$$

where $L^a = 2L$, $\nu = \sqrt{L+\gamma_0}$, $w_i^{*(m)}$ and $w_i^{*(c)}$ are calculated similar to Equation (2.31) with the replacement of L by L^a, $\mathbf{Q}_d$ is approximated by $\mathbf{G}(\hat{x}_{k|k-1})\mathbf{Q}\mathbf{G}^T(\hat{\boldsymbol{x}}_{k|k-1})\Delta t$ where $\Delta t = t_k - t_{k-1}$ [87].

Step2 Estimation of the faults

$$\boldsymbol{\gamma}_k = (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \tag{2.40}$$
$$\mathbf{N}_k = (\mathbf{F}_k^T \mathbf{P}_{yy,k}^{-1} \mathbf{F}_k)^{-1} \mathbf{F}_k^T \mathbf{P}_{yy,k}^{-1} \tag{2.41}$$
$$\hat{\boldsymbol{f}}_k^o = \mathbf{N}_k \boldsymbol{\gamma}_k \tag{2.42}$$
$$\mathbf{P}_k^f = (\mathbf{F}_k^T \mathbf{P}_{yy,k}^{-1} \mathbf{F}_k)^{-1} \tag{2.43}$$

where $\boldsymbol{\gamma}_k$ is the innovation, $\hat{\boldsymbol{f}}_k^o$ is the estimation of $\boldsymbol{f}_k^o$ and $\mathbf{P}_k^f$ is its error covariance matrix. $\mathbf{N}_k$ is the gain matrix which can achieve an unbiased estimation of $\boldsymbol{f}_k^o$.

Step3 Measurement update

$$\mathbf{K}_k = \mathbf{P}_{xy,k} \mathbf{P}_{yy,k}^{-1} \tag{2.44}$$
$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \mathbf{K}_k (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k - \mathbf{F}_k \hat{\boldsymbol{f}}_k^o) \tag{2.45}$$
$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k (\mathbf{P}_{yy,k} - \mathbf{F}_k \mathbf{P}_k^f \mathbf{F}_k^T) \mathbf{K}_k^T \tag{2.46}$$

This measurement update is different from that of the normal UKF [27, 81] which is given below for comparison and quick reference:

$$\mathbf{K}_k = \mathbf{P}_{xy,k} \mathbf{P}_{yy,k}^{-1} \tag{2.47}$$
$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \mathbf{K}_k (\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \tag{2.48}$$
$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{P}_{yy,k} \mathbf{K}_k^T \tag{2.49}$$

It can be seen that the measurement update of the normal UKF, as given by Equations (2.47)-(2.49), does not take the fault estimation and error covariance into account. Also note that the normal UKF does not estimate the faults, which means that it does not contain Equations (2.40)-(2.43).

The ADS fault estimation using the RTS-UKF is shown in the following.

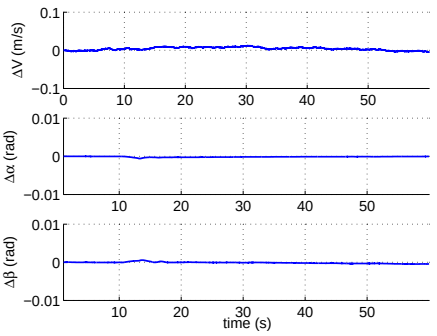
ADS FAULT ESTIMATION USING THE RTS-UKF

The performance of the RTS-UKF will be demonstrated under different initial conditions. The simulation data is taken from the simulation model of a Cessna Citation II aircraft. During 10 s $< t <$ 17 s. there is a 3-2-1-1 command on the aileron. The fault scenario is given in Table 2.1. The true initial state $\boldsymbol{x}_0$ is as follows:

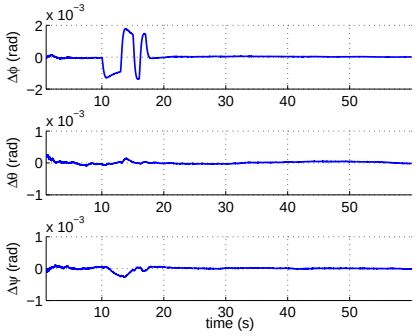
$$\boldsymbol{x}_0 = [90,\ 0.056,\ 0,\ 0,\ 0.0037,\ 0]^T \tag{2.50}$$

First, the true initial condition (2.50) is used as the initial guess $\hat{\boldsymbol{x}}_0$ in the filter. $\mathbf{P}_0 = 10^{-3} \cdot \mathbf{I}_6$. The standard deviations of the measurement noises are:

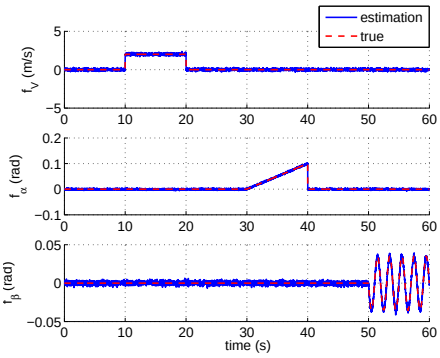
$$\sigma_{w_{Ax}} = \sigma_{w_{Ay}} = \sigma_{w_{Az}} = 0.001 \text{ m/s}^2$$
$$\sigma_{w_p} = \sigma_{w_q} = \sigma_{w_r} = 0.000018 \text{ rad/s}$$
$$\sigma_{v_V} = 0.1 \text{ m/s},\ \sigma_{v_\alpha} = \sigma_{v_\beta} = 0.0018 \text{ rad}$$
$$\sigma_{v_\phi} = \sigma_{v_\theta} = \sigma_{v_\psi} = 0.0018 \text{ rad}$$



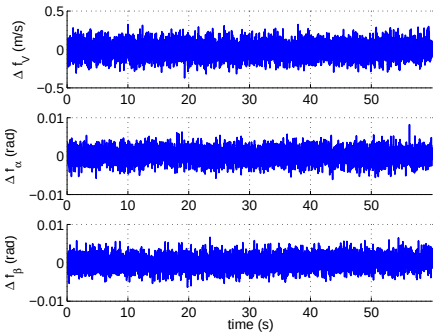
(a) Error of estimation of V, α and β



(b) Error of estimation of ϕ, θ and ψ



(c) Estimation of f_V, f_α and f_β



(d) Estimation error of f_V, f_α and f_β

Figure 2.1: Result of state and ADS fault estimation using the RTS-UKF approach and initial condition (2.50) in the presence of multiple faults

Therefore, $\mathbf{Q}$ and $\mathbf{R}$ can be inferred from Equations (2.9) and (2.10). The results are shown in Figure 2.1.

The estimation errors of V, α and β, as shown in Figure 2.1(a), are close to zero-mean. The estimation errors of ϕ, θ and ψ using the RTS-UKF are given in Figure 2.1(b). It can be seen that the estimation errors are zero-mean except during the period when there is a maneuver (10 s $< t <$ 17 s). However, during this period the estimation errors are small, e.g., the maximum estimation error of ϕ is less than 2×10^{-3} rad.

The estimation of f_V, f_α and f_β is given in Figure 2.1(c). As can be seen, all the faults are estimated in an unbiased sense. The estimation errors can be found in Figure 2.1(d).

Next, the performance with two different initial conditions for $\hat{x}_0$ is tested. The two initial conditions are as follows:

$$\hat{\boldsymbol{x}}_0 = [90,\ 0,\ 0,\ 0,\ 0,\ 0]^T, \tag{2.51}$$

$$\hat{\boldsymbol{x}}_0 = [1,\ 0,\ 0,\ 0,\ 0,\ 0]^T. \tag{2.52}$$

The initial condition (2.52) significantly deviates from the true initial condition (2.50) whereas condition (2.51) slightly deviates from condition (2.50). $\mathbf{P}_0$ is the same with the previous simulation and is $10^{-3} \cdot \mathbf{I}_6$.

The state estimation errors of the RTS-UKF using the initial condition Equation (2.51) are shown in Figure 2.2(a) and 2.2(b). As can be seen from Figure 2.2(a), the estimation errors of V, α and β are larger than those shown in Figure 2.1(a). The estimation errors of ϕ, θ and ψ, shown in Figure 2.2(b), are the same as those shown in Figure 2.1(b).

The state estimation errors of the RTS-UKF using the initial condition Equation (2.52) are shown in Figure 2.2(c) and 2.2(d). The estimation errors of V, α and β, shown in Figure 2.2(c), are significantly worse than those shown in Figure 2.1(a) and Figure 2.2(a). However, the estimation errors of ϕ, θ and ψ, shown in Figure 2.2(d), are still zero-mean.

The estimates of f_V, f_α and f_β using the initial condition Equations (2.51) and (2.52) are demonstrated in Figure 2.2(e) and 2.2(f) respectively. As can be seen from Figure 2.2(e), when the initial x_0 deviates from the true state, the estimates of the faults also deviate from their true magnitudes especially that of f_α. When the initial condition deviates significantly from the true initial condition, the performance becomes significantly worse, as can be seen in Figure 2.2(f).

Based on the above simulation results, it is seen that the RTS-UKF is sensitive to the initial condition errors. This sensitivity problem will be analyzed theoretically in the following section.

2.3.2. PROBLEM ANALYSIS OF THE ROBUST THREE-STEP FILTER

In the previous sections, it was shown that the performance of the RTS-UKF is influenced by the given initial condition. This section analyzes the problem of the sensitivity to the initial condition.

Rewrite Equation (2.45) into

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \mathbf{L}_k \boldsymbol{\gamma}_k \tag{2.53}$$

where $\mathbf{L}_k$ is defined as

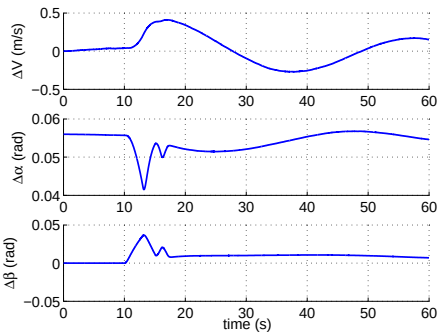
$$\mathbf{L}_k := \mathbf{K}_k (\mathbf{I} - \mathbf{F}_k \mathbf{N}_k) \tag{2.54}$$

The covariance matrix $\mathbf{P}_{xy,k}$, fault distribution matrix $\mathbf{F}_k$, the innovation $\boldsymbol{\gamma}_k$ and $\boldsymbol{x}$ can be partitioned as follows:

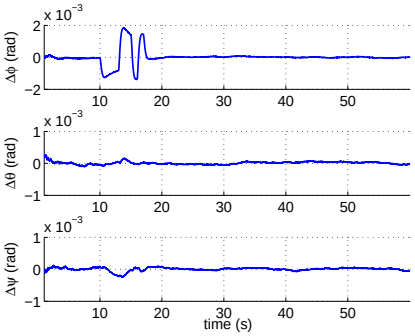
$$\begin{aligned} \mathbf{P}_{xy,k} &= \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix}, \quad \mathbf{F}_k = \begin{bmatrix} \mathbf{I}_p \\ \mathbf{0} \end{bmatrix}, \\ \boldsymbol{\gamma}_k &= \begin{bmatrix} \boldsymbol{\gamma}_p \\ \boldsymbol{\gamma}_{m-p} \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_p \\ \boldsymbol{x}_{L-p} \end{bmatrix} \end{aligned} \tag{2.55}$$

Since $\mathbf{P}_{yy,k}$ is invertible, its inverse can be partitioned as follows:

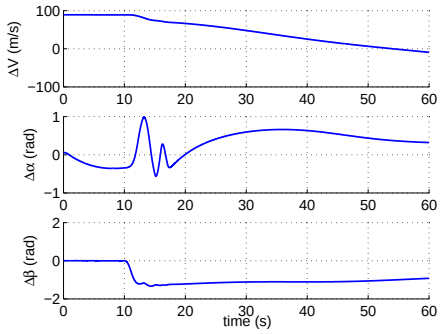
$$\mathbf{P}_{yy,k}^{-1} = \begin{bmatrix} \tilde{\mathbf{R}}_{11} & \tilde{\mathbf{R}}_{12} \\ \tilde{\mathbf{R}}_{21} & \tilde{\mathbf{R}}_{22} \end{bmatrix} \tag{2.56}$$



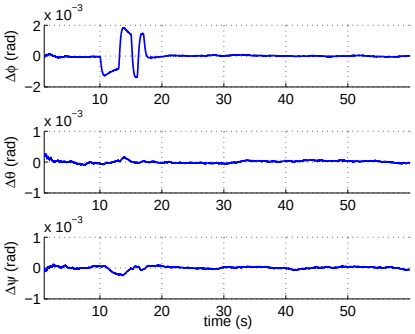
(a) Error of estimation of V, α and β using condition (2.51)



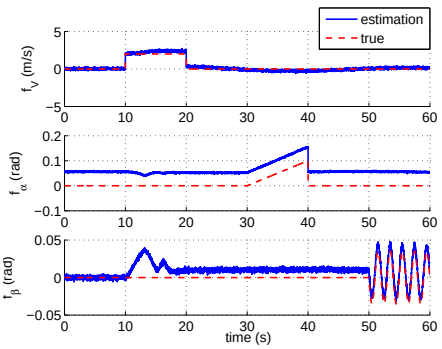
(b) Error of estimation of ϕ, θ and ψ using condition (2.51)



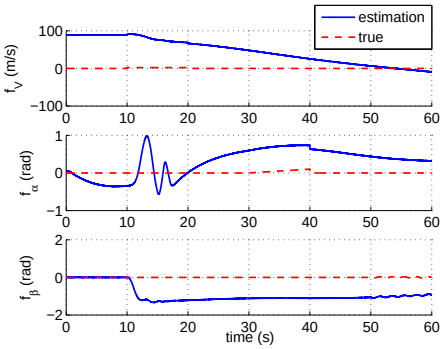
(c) Error of estimation of V, α and β using condition (2.52)



(d) Error of estimation of ϕ, θ and ψ using condition (2.52)



(e) Estimation of f_V, f_α and f_β using condition (2.51)



(f) Estimation of f_V, f_α and f_β using condition (2.52)

Figure 2.2: Result of state and fault estimation using the RTS-UKF approach and two different initial conditions in the presence of multiple faults

where $\tilde{\mathbf{R}}_{11} \in \mathbb{R}^{p\times p}$, $\tilde{\mathbf{R}}_{12} \in \mathbb{R}^{p\times(m-p)}$, $\tilde{\mathbf{R}}_{21} \in \mathbb{R}^{(m-p)\times p}$ and $\tilde{\mathbf{R}}_{22} \in \mathbb{R}^{(m-p)\times(m-p)}$. Therefore, Equation (2.41) can be computed by

$$\begin{aligned}\mathbf{N}_k &= [\tilde{\mathbf{R}}_{11}^{-1}\ \mathbf{0}]\begin{bmatrix}\tilde{\mathbf{R}}_{11} & \tilde{\mathbf{R}}_{12}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{I}_p & \tilde{\mathbf{R}}_{11}^{-1}\tilde{\mathbf{R}}_{12}\end{bmatrix}\end{aligned} \tag{2.57}$$

Substituting Equation (2.57) into Equation (2.54), it follows

$$\mathbf{L}_k = \mathbf{P}_{xy,k}\mathbf{P}_{yy,k}^{-1}\begin{bmatrix}\mathbf{0} & -\tilde{\mathbf{R}}_{11}^{-1}\tilde{\mathbf{R}}_{12}\\ \mathbf{0} & \mathbf{I}_{m-p}\end{bmatrix} \tag{2.58}$$

$$= \begin{bmatrix}\mathbf{0} & \mathbf{L}_{12}\\ \mathbf{0} & \mathbf{L}_{22}\end{bmatrix} \tag{2.59}$$

where $\mathbf{L}_{12}$ and $\mathbf{L}_{22}$ are defined as

$$\begin{aligned}\mathbf{L}_{12} &:= \mathbf{P}_{12}(\tilde{\mathbf{R}}_{22} - \tilde{\mathbf{R}}_{21}\tilde{\mathbf{R}}_{11}^{-1}\tilde{\mathbf{R}}_{12})\\ \mathbf{L}_{22} &:= \mathbf{P}_{22}(\tilde{\mathbf{R}}_{22} - \tilde{\mathbf{R}}_{21}\tilde{\mathbf{R}}_{11}^{-1}\tilde{\mathbf{R}}_{12})\end{aligned}$$

Therefore, the measurement update of the robust three-step filter, denoted in Equation (2.53), can be further written as follows:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \begin{bmatrix}\mathbf{L}_{12}\boldsymbol{\gamma}_{m-p}\\ \mathbf{L}_{22}\boldsymbol{\gamma}_{m-p}\end{bmatrix} \tag{2.60}$$

It can be seen that $\boldsymbol{\gamma}_p$ is not used in the measurement update. Since $\boldsymbol{\gamma}_p$ is not used, the estimation of $\boldsymbol{x}_p$ is not updated by measurements of $\boldsymbol{x}_p$. Therefore, the estimation of $\boldsymbol{x}_p$(V, α and β), is sensitive to the initial condition. If the initial $\boldsymbol{x}_0$ significantly deviates from the true value, it will not be corrected to the true value. However, the estimation of ϕ, θ and ψ is not influenced since they are updated by the measurement. This is consistent with the result shown in Figures 2.2(b) and 2.2(d), where the estimation of ϕ, θ and ψ is still good even when that of V, α and β is not.

In case that $p = m$ and rank $\mathbf{F}_k = m$, it can be found that

$$\mathbf{N}_k = \mathbf{F}_k^{-1} \tag{2.61}$$

$$\mathbf{L}_k = \mathbf{0} \tag{2.62}$$

Consequently, the measurement update of the three-step Kalman filter is

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} \tag{2.63}$$

This means that all the states are not updated by their measurements. In this situation, all the state estimation will be sensitive to the initial condition.

Through the analysis in this section and the performance demonstration of the RTS-UKF in Section 2.3.1, the need for a modification of the RTS-UKF is emphasized. In real life, the exact initial condition is difficult to obtain due to uncertainties in the system (which can also be found in Section 2.5). The RTS-UKF will interpret the initialization error as a fault, which results into wrong fault estimation. Therefore, the RTS-UKF can not be applied to the FDD of the ADSs.

2.3.3. NOVEL ADAPTIVE THREE-STEP UNSCENTED KALMAN FILTER FOR ADS FDD

Having found the cause for performance degradation of the RTS-UKF, this section proposes a novel ATS-UKF to solve the ADS FDD. The sensitivity to the initial condition of the RTS-UKF can be solved by performing the measurement update of normal UKF.

It should be noted that the RTS-UKF only considers the estimation of the faults. It does not detect and isolate the faults. The proposed ATS-UKF deals with not only the estimation of the faults, but also the detection and isolation.

In the following, the initial measurement update and FDI scheme are introduced. Then the complete FDD system is introduced.

INITIAL MEASUREMENT UPDATE

The solution to reduce the sensitivity of the RTS-UKF to the initial condition is proposed in this subsection, which is to use the measurement update of normal UKF (Equations (2.47)-(2.49)) when the state estimation is influenced by the initialization error. However, when the correction is sufficient, i.e., when the measurement update of the UKF is sufficient, needs to be determined. This chapter proposes a criteria which can determine whether the measurement update of the UKF is sufficient. The details are given as follows:

Let $C_{ii,k},\ i = 1,2,3$ denote the ith diagonal elements of the innovation covariance matrix $\mathbf{C}_k$ associated with the measurements which are not used in the update of the RTS-UKF at time step k. (i.e., the measurement of V, α and β respectively in this chapter).

Define the change of the innovation variance $\Delta C_{ii,k}$ as

$$\Delta C_{ii,k} := C_{ii,k} - C_{ii,k-1}, \quad i = 1,2,3. \tag{2.64}$$

When the following inequality holds, the measurement update can be regarded as sufficient. The inequality is

$$\Delta C_{ii,k} < \eta_i, \quad i = 1,2,3. \tag{2.65}$$

where $\eta_i,\ i = 1,2,3$ are pre-defined constants which can be tuned to stop the measurement update. The principle is that if there are initialization errors, $C_{ii,k}$ is not constant. When the filter achieves steady-state, $C_{ii,k}$ is approximately constant. Therefore, $\Delta C_{ii,k}$ should be small. If $\Delta C_{ii,k}$ is smaller than η_i, then it indicates that the filter has reached steady-state and the measurement update of the UKF is sufficient. If η_i is chosen to be small, then the number of initial measurement update will be bigger while the influence

of the initial condition error will be less. $\mathbf{C}_k$ can be estimated using the following [88, 89]:

$$\hat{\mathbf{C}}_k = \frac{1}{N} \sum_{j=k-N+1}^{k} \boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^T \tag{2.66}$$

where $\boldsymbol{\gamma}_j$ denotes the innovation at time step j.

FAULT DETECTION AND ISOLATION

The fault detection is performed by monitoring the innovation variance of the filter. In the presence of ith fault, $C_{ii,k}$ increases. The fault detection and isolation logic at time step k is:

if $C_{ii,k} > T_i$, $F_{A_i} = 1$. otherwise $F_{A_i} = 0$, i = 1,2,3.

where $\boldsymbol{F}_A = [F_{A_V}\ F_{A_\alpha}\ F_{A_\beta}]^T$ are the alarm indicators. T_i are the thresholds which are designed to detect the faults in the V, α and β sensors respectively. These thresholds are designed based on the fault-free case. It can be seen that the fault detection and isolation are simultaneously realized.

The weighted fault estimation can be calculated as follows:

$$\tilde{f}_{i,k}^o = F_{A_i} \hat{f}_{i,k}^o, \qquad i = 1,2,3. \tag{2.67}$$

ADAPTIVE THREE-STEP UNSCENTED KALMAN FILTER

When the initial measurement update is sufficient, there are two options to achieve FDD which are as follows:

1. After the initial measurement update, the FDI scheme is used to detect and isolate the faults. The RTS-UKF is used to estimate the faults.

2. After the initial measurement update, the FDI scheme is used to detect and isolate the faults. If there are no faults detected, the UKF is used and the fault estimation is considered to be zero. If there are faults detected, then the RTS-UKF is used for the fault estimation and measurement update.

The ATS-UKF proposed in this chapter, is based on the latter one since it can reduce the computational load. The measurement update of the ATS-UKF switches adaptively between that of the normal UKF and that of the RTS-UKF through the FDI scheme. The specific three steps of the ATS-UKF are given as follows:

1. Time update
   This is the same as in the UKF, which also includes the sigma point calculation. The steps are described by Equations (2.30)- (2.39).

2. Fault estimation
   Before estimating the faults, the FDI, which has been introduced above, is performed. If $\boldsymbol{F}_A = \mathbf{0}$, then $\hat{\boldsymbol{f}}_k^o = \mathbf{0}$. If $\boldsymbol{F}_A \neq \mathbf{0}$, then the faults are estimated using the RTS-UKF, which is described by Equations (2.40)-(2.43). This step is the FDD.

3. Measurement update
   As mentioned, during the initial measurement update, the measurement update of the normal UKF is applied. When the initial measurement update is done, the measurement update of the ATS-UKF is as follows:

   (a) If $\boldsymbol{F}_A = \mathbf{0}$, use the the measurement update of the normal UKF.
   In this situation, there are no faults detected in the system. To reduce the computational load, the measurement update of the UKF is used. This means that the faults are considered to be zero, so that the faults estimation and measurement update of the RTS-UKF are not needed. The steps are described by Equations (2.47)-(2.49).

   (b) If $\boldsymbol{F}_A \neq \mathbf{0}$, use the measurement update of the RTS-UKF.
   In this situation, faults are detected. Therefore, the measurement update of the RTS-UKF is needed to obtain an unbiased state estimation and fault estimation, which can not be achieved using the normal UKF. The steps are described by Equations (2.44)-(2.46).

2.3.4. ADS FDD USING THE ATS-UKF

In this section, the FDD as well as the state estimation performance of the proposed ATS-UKF is demonstrated using two different fault scenarios. The initial condition is the same as in Equation (2.52). The threshold to stop the initial measurement update is $\boldsymbol{\eta} = [5 \times 10^{-3},\ 2 \times 10^{-5},\ 2 \times 10^{-5}]^T$ and the threshold to detect the fault is $\boldsymbol{T} = [0.2,\ 1 \times 10^{-4},\ 5 \times 10^{-5}]^T$.

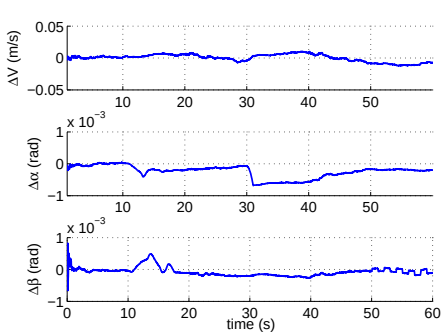
MULTIPLE FDD

In this scenario, consecutive ADS faults are generated, which are shown in Table 2.1. The results using the ATS-UKF are shown in Figure 2.3.

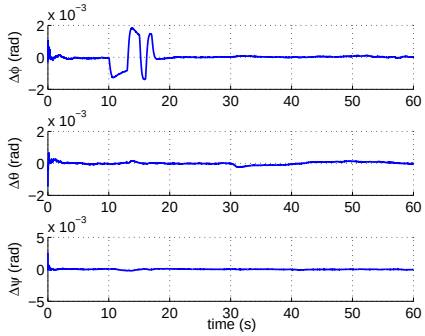
It is found that using the above thresholds, the initial measurement update is only performed for two time steps. The estimation of V, α and β is shown in Figure 2.3(a). Despite the fact that the initial x_0 significantly deviates from the true state, the estimation is still satisfactory. This means that the sensitivity to the initial condition of the RTS-UKF is tackled by the ATS-UKF. The estimation of ϕ, θ and ψ, as shown in Figure 2.3(b), is also satisfactory. This demonstrates the state estimation performance of the ATS-UKF.

The fault detection and isolation is achieved by checking F_{A_V}, F_{A_α} and F_{A_β}, which is shown in Figure 2.3(c). From the figure, it can be seen that f_V is detected instantaneously. The detection of f_α takes longer than that of f_V. This is because f_α is a drift fault which is a slow time-varying fault. f_β is also detected instantaneously. However, F_{A_β} switches from 1 to 0 nine times. This is as expected since the oscillatory fault crosses zero nine times. When the magnitude of the fault is zero, it can be regarded as no fault. From the figure, it is obvious that fault isolation is also achieved. For instance, when $F_{A_V} = 1$, both F_{A_α} and F_{A_β} are equal to zero, which means only f_V occurs.

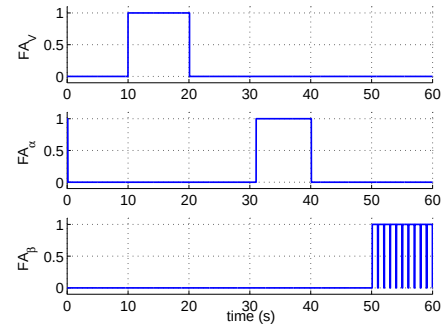
The estimation of f_V, f_α and f_β is shown in Figure 2.3(d). As can be seen, all the faults are estimated in an unbiased sense. The weighted fault estimation, calculated using Equation (2.67), is shown in Figure 2.3(e). The error of the estimation of f_V, f_α and f_β is shown in Figure 2.3(f). It is seen that all the estimation errors are zero-mean.



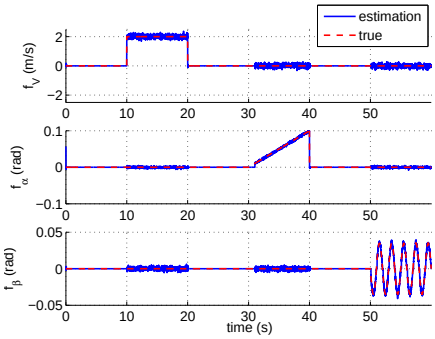
(a) Error of estimation of V, α and β



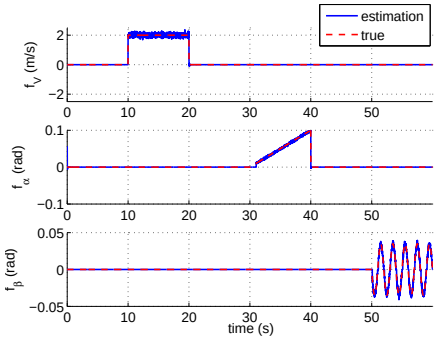
(b) Error of estimation of ϕ, θ and ψ



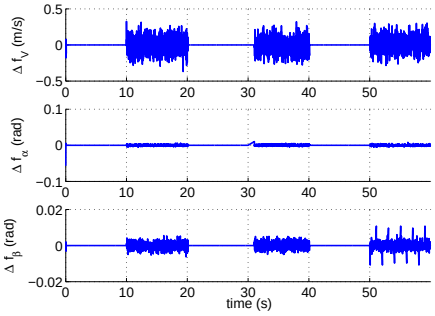
(c) Fault detection and isolation



(d) Estimation of f_V, f_α and f_β



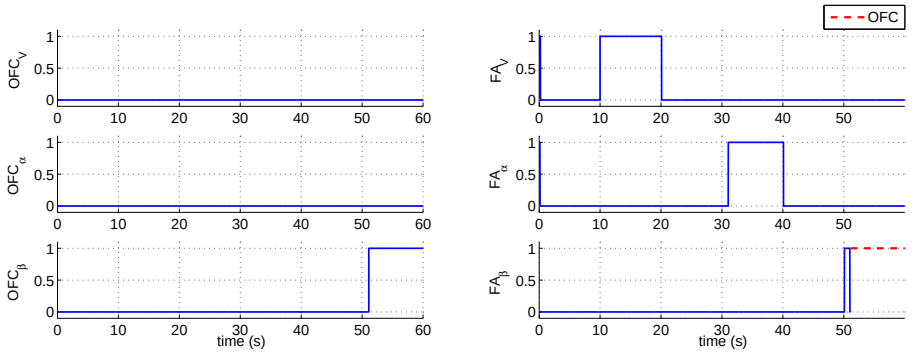
(e) Weighted estimation of f_V, f_α and f_β



(f) Error of estimation of f_V, f_α and f_β

Figure 2.3: Result of state estimation and ADS FDD using the proposed ATS-UKF approach in the presence of multiple faults

This demonstrates the fault estimation performance of the ATS-UKF. It is also noticed that when there are no faults or the faults are not detected, the estimates of the fault are



(a) Detection of oscillatory faults, OFC indicates the detection of oscillatory faults

(b) Fault detection and isolation including oscillatory fault indication

Figure 2.4: Result of fault detection and isolation using the proposed ATS-UKF approach in the presence of multiple faults

zero and so are the estimation errors. This is due to the fact that when there are no faults detected, the measurement update of the UKF is used and the faults are considered to be zero.

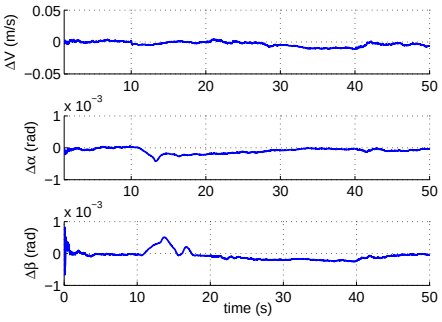
The fault detection of the oscillatory faults, shown in Figure 2.3(c), shows a chattering behavior. To detect the presence of oscillatory failures, the detection logic of oscillatory faults in Goupil [39] is used. The basic idea is to count the crossings of the fault estimate (shown in Figure 2.3(e)) through a positive and negative threshold within a sliding time window. In this chapter, the oscillatory faults are detected if one full oscillation is detected. The result of detecting the oscillatory fault is shown in Figure 2.4(a). In the figure, OFC denotes oscillatory failure case (OFC). As can be seen, an oscillatory fault is only detected in the β sensor. If we take the bigger value of F_{Ai} and OFC_i (i is associated with the sensor of V, α and β), the fault detection including the detection of the oscillatory fault can be obtained, which is demonstrated in Figure 2.4(b). In the figure, the red dashed line indicates that the detected fault is an oscillatory fault.

SIMULTANEOUS FDD

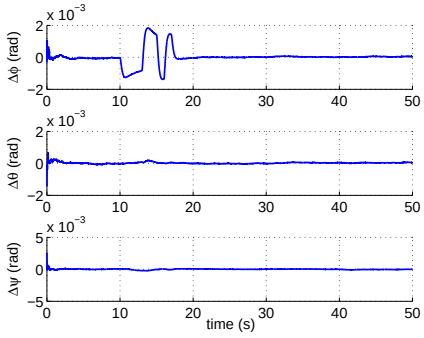
In this scenario, simultaneous faults are generated which are shown in Table 2.2. The ATS-UKF is used to detect, isolate and estimate these faults. The results are given in Figure 2.5.

The estimation errors of V, α, β and ϕ, θ and ψ using the ATS-UKF are shown in Figure 2.5(a) and Figure 2.5(b), respectively. As can be seen, even in the presence of simultaneous faults, the state estimation performance of the ATS-UKF is still satisfactory.

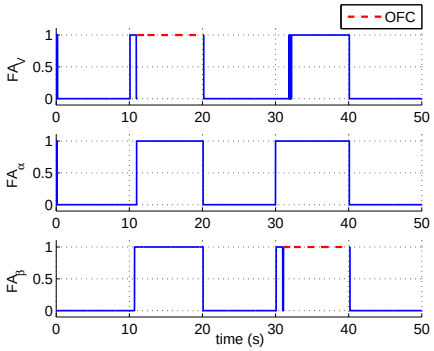
The fault detection and isolation performance is shown in Figure 2.5(c). As can be seen, there are no false alarms, which demonstrates its good performance. For the detection of f_α and f_β during 10 s$< t <$ 20 s, and f_V during 30 s$< t <$ 40 s, there are detection delays since there are drift faults. The dashed lines in the figure indicate that the detected faults are oscillatory faults.



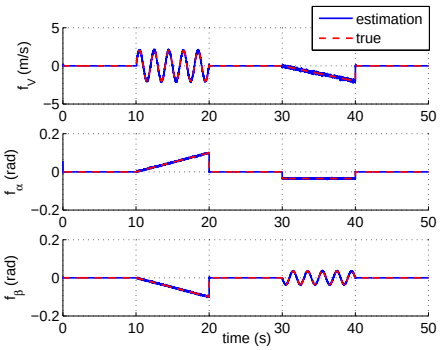
(a) Error of estimation of V, α and β



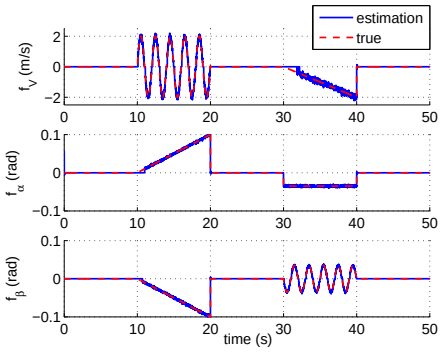
(b) Error of estimation of ϕ, θ and ψ



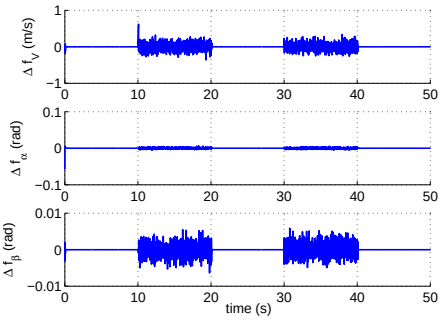
(c) Fault detection and isolation



(d) Estimation of f_V, f_α and f_β



(e) Weighted estimation of f_V, f_α and f_β



(f) Error of estimation of f_V, f_α and f_β

Figure 2.5: Result of state and ADS FDD using the proposed ATS-UKF approach in the presence of simultaneous faults

Figure 2.5(d) and 2.5(e) show the estimation and weighted estimation of f_V, f_α and

f_β, respectively. It can be seen that the fault estimation performance is satisfactory. All faults are estimated in an unbiased sense including the oscillatory faults. The estimation error of f_V, f_α and f_β is shown in Figure 2.5(f). It can be seen that the error is zero-mean, which confirms the good estimation performance of the ATS-UKF.

2.4. PERFORMANCE VALIDATION USING REAL FLIGHT DATA

In the previous section, the FDD performance of the ATS-UKF is tested using simulated aircraft data. In this section, the FDD performance of the ATS-UKF is validated using real flight test data of the Cessna Citation II aircraft. Air data information such as α and β are measured for post flight analysis. The real flight data contains uncertainties such as biases and spikes. Additionally, in real flight, external disturbances, such as changing wind, can also influence the air data measurements. Therefore, the real flight data poses challenges to the ADS FDD problem and provides a realistic validation of the performance of FDD approaches such as the ATS-UKF.

The primary objective of the flight test is aerodynamic model identification where a number of maneuvers were performed by the aircraft in order to obtain sufficient excitation. Since there were no faults during the flight, sensor faults are injected into the real flight data to validate the FDD performance of the ATS-UKF. Besides the fault scenarios presented in Tables 2.1 and 2.2, a fault-free case is also studied.

The real flight data used in this chapter is the same as that in Lu et. al [85]. In Lu et. al, the estimated wind turns out to be time varying. This can test the ADS FDD performance of the ATS-UKF under the condition of winds.

The update rates of the on-board sensors are given in Table 2.3.

Table 2.3: Update frequencies of different measurements

Measurements	Unit	Update frequency
V	[m/s]	100 Hz
u_n, v_n, w_n	[m/s]	1 Hz
α, β	[rad]	100 Hz
ϕ, θ	[rad]	100 Hz
ψ	[rad]	10 Hz

2.4.1. REAL-LIFE MEASUREMENT MODEL

For simulated aircraft data, the measurement model is given in Equations (2.19) - (2.24). If $\boldsymbol{f}^o = \mathbf{0}$, the measurements are only corrupted by white Gaussian noises, as can be seen from the equations. However, this is never the case in real life. In this flight test, the air data information, such as α and β, is measured by multiple vanes on a boom (shown in Figure 2.6) which is mounted on the nose of the aircraft. It is assumed that the vanes are rigid so that the effects of vibrations can be neglected. The angle of attack and angle of sideslip measured by the vanes are denoted by α_{vm} and β_{vm}, respectively. The

measurements α_{vm} and β_{vm} is different from Equations (2.20) and (2.21), respectively. The measurement model for the real-life measurements is given as follows [90–92]:

$$V_m = V + v_V \tag{2.68}$$

$$\alpha_m = C_{\alpha 0} + (1 + C_{up})\alpha + \frac{x_\alpha q}{V} + v_\alpha \tag{2.69}$$

$$\beta_m = C_{\beta 0} + (1 + C_{si})\beta - \frac{x_\beta r}{V} + \frac{z_\beta p}{V} + v_\beta \tag{2.70}$$

$$\phi_m = \phi + v_\phi \tag{2.71}$$

$$\theta_m = \theta + v_\theta \tag{2.72}$$

$$\psi_m = \psi + v_\psi \tag{2.73}$$

where x_α, x_β and z_β are the position of the vanes in the body frame, $C_{\alpha 0}$, $C_{\beta 0}$, C_{up} and C_{si} are the boom correction parameters. In this chapter, z_β is assumed to be zero. The parameter estimation can be found in Lu et al. [92]. For the ADS FDD using real flight data, this measurement model is used. However, it should be noted that in this real-life measurement model, boom bending is considered to be negligible for the maneuvers flown.

Figure 2.6: The vanes on the boom for measuring the angle of attack and angle of sideslip. Photo credits by Daan Pool.

2.4.2. ADS FDD USING REAL FLIGHT DATA IN THE ABSENCE OF FAULTS

Using the real-life measurements, uncertainties and disturbances such as varying winds can also have a negative influence on the FDD performance, which can result in false alarms. Under this condition, the FDD approaches should not give false alarms. In this section, the ATS-UKF is tested in a fault-free case to verify whether it gives false alarms.

In order to show the effectiveness of the ATS-UKF, the RTS-UKF is also applied to estimate the ADS faults. The initial condition $\boldsymbol{x}_0$ given for the RTS-UKF is the first measurement which is:

$$[104.8733,\ 0.0796,\ 0.0073,\ -0.0019,\ 0.0733,\ 4.6692]^T \tag{2.74}$$

The initial condition x_0 given for the ATS-UKF is:

$$[1,\ 0,\ 0,\ 0,\ 0,\ 0]^T \tag{2.75}$$

In this manner, the initial condition given for the RTS-UKF is close to the true state whereas that of the ATS-UKF significantly deviates from the true state. The threshold to stop the initial measurement update is $\boldsymbol{\eta} = [5 \times 10^{-3},\ 2 \times 10^{-5},\ 2 \times 10^{-5}]^T$ and the threshold to detect the fault is $\boldsymbol{T} = [0.2,\ 1 \times 10^{-4},\ 5 \times 10^{-5}]^T$, which are the same as those used in the previous section.

The estimation of V, α and β using the RTS-UKF is given in Figure 2.7(a). The fault estimation using the RTS-UKF is given in Figure 2.7(b). As can be seen, the estimated faults deviate from their true magnitudes. This result shows that the RTS-UKF is not able to be applied for real applications unless modifications are made.

The results of the ATS-UKF are shown in Figures 2.7(c) - 2.7(f). The estimation of V, α and β, and ϕ, θ and ψ are shown in Figures 2.7(c) and 2.7(d) respectively. The estimates of α and β using the ATS-UKF are different from those using the RTS-UKF. The fault detection result, shown in Figure 2.7(e), indicates that there are no faults. This demonstrates that the ATS-UKF does not give false alarms in the presence of no faults even when the real flight data is used. The weighted estimates of f_V, f_α and f_β, shown in Figure 2.7(f), are zero-mean. This confirms the fault estimation performance of the ATS-UKF.

2.4.3. ADS FDD USING REAL FLIGHT DATA IN THE PRESENCE OF MULTIPLE FAULTS

In this subsection, the ADS FDD performance of the ATS-UKF will be verified using the real-life measurement model in the presence of multiple faults (given in Table 2.1). The initial condition for RTS-UKF is the same as in (2.74) and that of the ATS-UKF is the same as in (2.75).

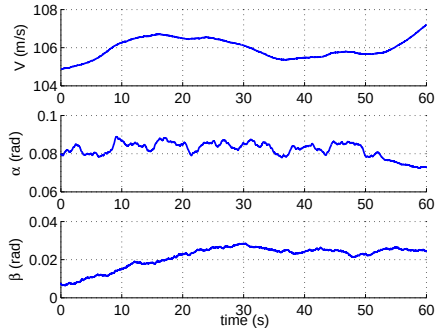
The results using these two approaches are shown in Figure 2.8.

The estimation of V, α and β using the RTS-UKF is shown in Figure 2.8(a). The fault estimation is shown in Figure 2.8(b). Although the initial condition of the RTS-UKF is chosen to be the measurements, the estimation of the faults are still biased. This shows the drawback of the RTS-UKF when it is used in practice because the initial condition error will be estimated as a bias fault.

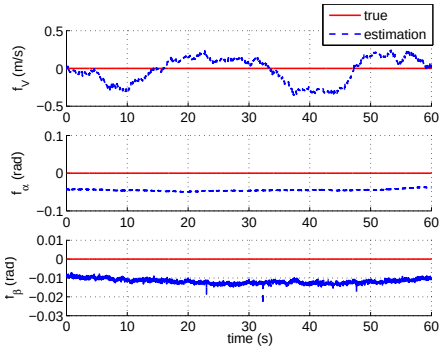
The estimation of V, α, β and ϕ, θ, ψ using the ATS-UKF is presented in Figure 2.8(c) and 2.8(d) respectively. It can be seen that the estimates of α, β are again different from those of the RTS-UKF shown in Figure 2.8(a).

The fault detection and isolation using the ATS-UKF is given in Figure 2.8(e). No false alarms are generated and the isolation is also correct. It can be seen that the performance is as good as that in Figure 2.4(b) where the simulation data is used. The oscillatory fault is also detected, which is shown by the dashed line.

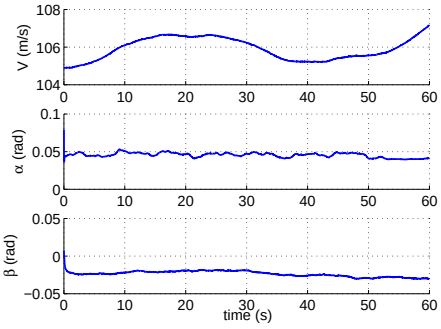
The weighted estimation of f_V, f_α and f_β using the ATS-UKF is presented in Figure 2.8(f). Even though the initial condition of the ATS-UKF deviates from the true state significantly, its performance is not sensitive to the initial condition. Since the faults are estimated in an unbiased sense, the estimates of α, β using the ATS-UKF are more reliable than those using the RTS-UKF.



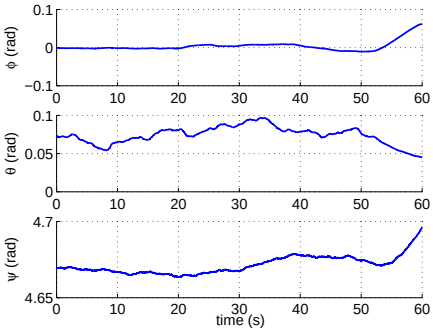
(a) Estimation of V, α and β using the RTS-UKF



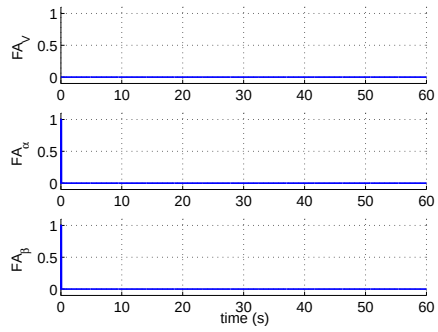
(b) Estimation of f_V, f_α and f_β using the RTS-UKF



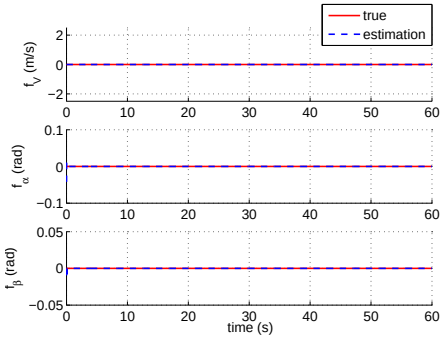
(c) Estimation of V, α and β using the ATS-UKF



(d) Estimation of ϕ, θ and ψ using the ATS-UKF

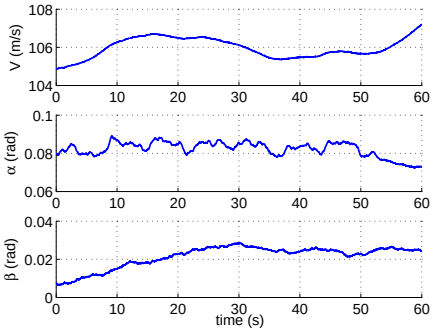


(e) Fault detection and isolation using the ATS-UKF

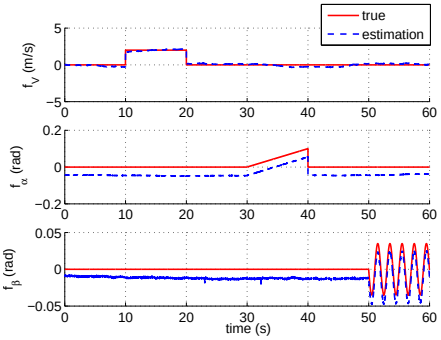


(f) Weighted estimation of f_V, f_α and f_β using the ATS-UKF

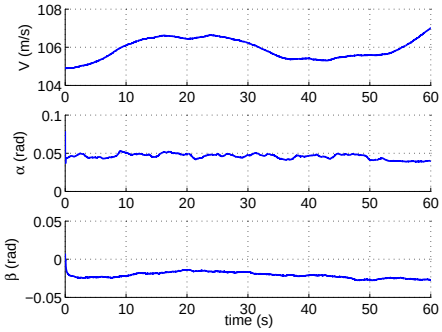
Figure 2.7: State estimation and ADS FDD of the real-life measurement model of the aircraft using the RTS-UKF and the ATS-UKF approach in the absence of faults



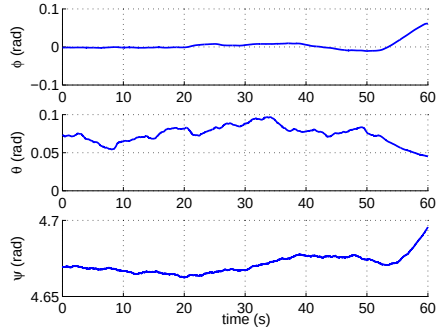
(a) Estimation of V, α and β using the RTS-UKF



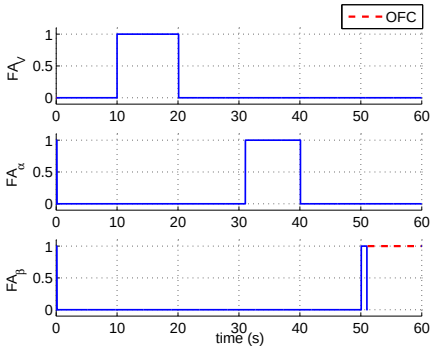
(b) Estimation of f_V, f_α and f_β using the RTS-UKF



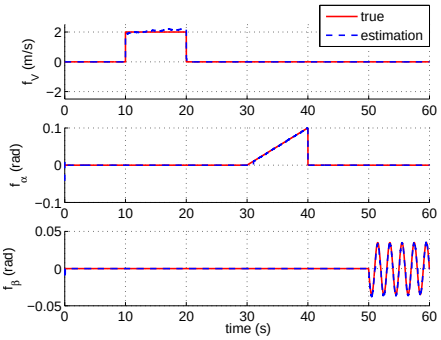
(c) Estimation of V, α and β using the ATS-UKF



(d) Estimation of ϕ, θ and ψ using the ATS-UKF



(e) Fault detection and isolation using the ATS-UKF

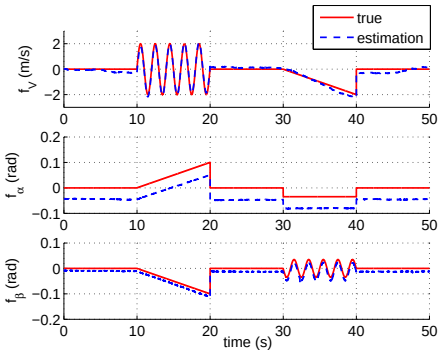


(f) Weighted estimation of f_V, f_α and f_β using the ATS-UKF

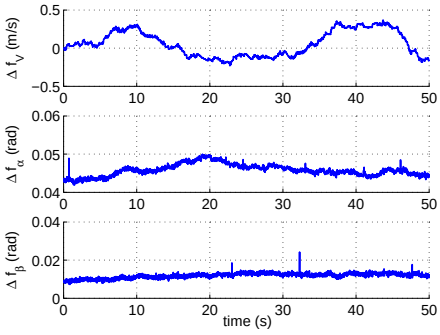
Figure 2.8: State estimation and ADS FDD of the real-life measurement model of the aircraft using the RTS-UKF and the ATS-UKF approach in the presence of multiple faults

2.4.4. ADS FDD USING REAL FLIGHT DATA IN THE PRESENCE OF SIMULTANEOUS FAULTS

In this subsection, simultaneous faults (given in Table 2.2) are injected into the real flight data to validate the performance of the ATS-UKF. The result of the RTS-UKF is also pre-



(a) Estimation of f_V, f_α and f_β



(b) Error of estimation of f_V, f_α and f_β

Figure 2.9: ADS FDD of the real-life measurement model of the aircraft using the RTS-UKF approach in the presence of simultaneous faults

sented, which is given in Figure 2.9. From this figure, it is seen the fault estimation of the RTS-UKF is again biased although the initial x_0 is chosen to be the measurements. This highlights the limitation of the RTS-UKF when used in reality.

The results using the ATS-UKF is shown in Figure 2.10. The state estimation is presented in Figures 2.10(a) and 2.10(b) respectively.

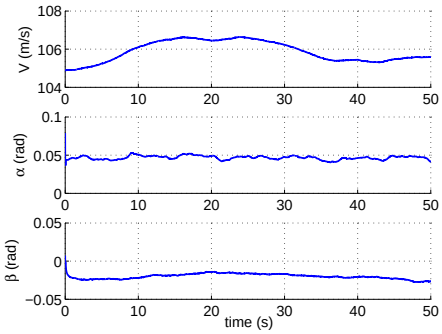
The fault detection and isolation using the ATS-UKF is shown in Figure 2.10(c). No false alarms are observed from the figure and the oscillatory faults are also detected. The performance is as good as that in Figure 2.5(c) where the simulation data is used. This confirms the FDI performance of the ATS-UKF.

Estimation and weighted estimation of f_V, f_α and f_β using the ATS-UKF are shown in Figures 2.10(d) and 2.10(e), respectively. All fault estimates achieve an unbiased estimation. The fault estimation errors are demonstrated in Figure 2.10(f). Although the errors are not zero-mean, they are small compared to the states.

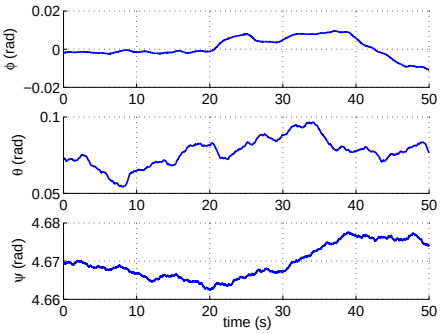
2.5. CONCLUSIONS

This chapter deals with the ADS FDD of aircraft. First, the RTS-KF is extended to the RTS-UKF to cope with nonlinear systems. Second, the RTS-UKF is found to be sensitive to the initial condition. The problem is analyzed theoretically and subsequently, a novel ATS-UKF is proposed to detect, isolate and estimate the ADS faults. The ATS-UKF contains three steps: time update, fault estimation and measurement update. The ATS-UKF is validated using simulated aircraft data, which shows good FDD performance.

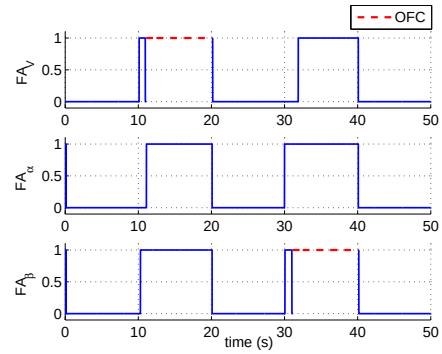
The performance of the ATS-UKF is further validated using real flight data of the Cessna Citation II aircraft to test its performance under real-life uncertainties. It was found that although the measurement data contains biases which can not be removed completely and the initial condition is far from the true state, the ATS-UKF is still able to maintain its satisfactory FDD performance. This demonstrates that it can be applied in practice.



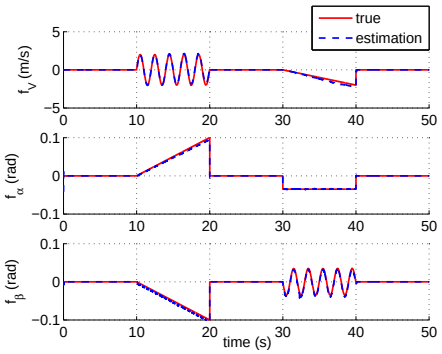
(a) Estimation of V, α and β



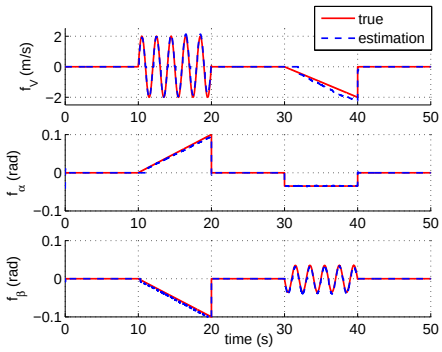
(b) Estimation of ϕ, θ and ψ



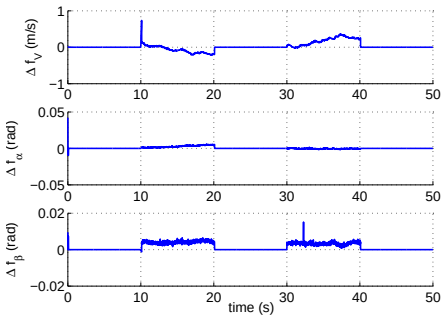
(c) Fault detection and isolation



(d) Estimation of f_V, f_α and f_β



(e) Weighted estimation of f_V, f_α and f_β



(f) Error of estimation of f_V, f_α and f_β

Figure 2.10: State estimation and ADS FDD of the real-life measurement model of the aircraft using the ATS-UKF in the presence of simultaneous faults

This ATS-UKF, which deals with ADS FDD, can be incorporated into FTC systems to

further enhance the safety of the aircraft. It can detect faults without giving false alarms. In addition, it can provide both unbiased state estimation and fault estimation, which are important for the recovery from sensor faults.

In the future, the ATS-UKF should be integrated into a FTC system. Finally, it is highly recommended that a real-world flight experiment is designed and executed to detect and estimate ADS faults in aircraft during flight.

3

SELECTIVE-REINITIALIZATION MULTIPLE-MODEL ADAPTIVE ESTIMATION FOR FAULT DETECTION AND DIAGNOSIS

In the previous chapter, ADS FDD in the presence of model uncertainties was coped with. In this chapter, both IMU and ADS FDD will be considered. The model used is still the airspeed-based kinematic model. The approach for state and fault estimation is Selective-Reinitialization Multiple-Model Adaptive Estimation approach. This approach offers a faster fault detection compared to the approach used in the previous chapter. In the first part of this chapter, the IMU FDD is considered. Then the ADS FDD is addressed.

The simulation data used in this chapter is available at: https://www.researchgate.net/profile/Peng_Lu15/publications?pubType=dataset.

Parts of this chapter are based on:

P Lu, L Van Eykeren, E van Kampen, Q P Chu. Selective-Reinitialization Multiple Model Adaptive Estimation for Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics*, 38(8), 1409-1424, 2015. [83]

The existing MMAE approach is able to detect faults quickly. However, there are three main problems when it is used for Fault Detection and Diagnosis: false alarms, requirement of designing additional models to identify the faults and slow response to detect the removal of the faults. In this chapter, a novel SRMMAE approach is proposed. This approach introduces a state augmentation strategy which can identify the faults without designing additional models and reduce false alarms. The major contribution of this approach is that three selective reinitialization algorithms are proposed which can improve the performance of the MMAE significantly. The SRMMAE approach eliminates false alarms and is quick to detect the removal of the faults. The performance of the propose approach is compared with the MMAE and the IMM with an example of the fault diagnosis of the Inertial Measurement Unit and Air Data Sensors for a Cessna Citation II aircraft. The simulation results suggest that the SRMMAE outperforms the MMAE and IMM in effectiveness and efficiency.

3.1. INTRODUCTION

PRESENTLY, FDD has an important role of achieving fault-tolerance [11]. State-of-the-art aircraft FDD systems mainly rely on hardware redundancy to enhance the reliability and safety of the aircraft [93]. The same holds for the electronic flight control systems or fly-by-fire systems of modern aircraft [9]. Sensor or actuator faults may cause serious problems; hence, quick detection and isolation of these faults is required [9]. To achieve a better FDD performance (e.g. earlier detection of small and subtle faults), a valuable solution is the model-based analytical redundancy techniques [11]. During the last decades, many such approaches [13, 19, 66, 94–98] have been proposed to perform the FDD of aircraft actuators and sensors. Some recent advances and trends can be found in Zolghadri [14]. Zolghadri et al. [70] address the actuator faults including oscillatory, runaway and jamming faults. They also address the IMU sensor faults. Their presented techniques have been tested on highly representative benchmarks, real flight data or real-world aerospace systems [70]. The present chapter is concerned with the FDD of IMU sensor and ADS faults. The fault types include bias and drift faults.

One of the most effective approaches for FDD is the multiple model-based approach [99]. The MMAE [34, 37, 93, 100, 101] runs a bank of filters in parallel, termed "elemental filters". Each of these filters is based on a hypothesized parameter vector matching to a particular fault mode. When the MMAE is used to estimate the value of faults, a number of additional filters are required, which increases the computational load significantly. To cope with that, Maybeck et al. proposed a moving bank MMAE algorithm [36, 102] and a hierarchical structure [37] to reduce the number of required filters online. However, false alarms and slow detection of the removal of faults still remain. To improve the FDD performance, Ducard et al. [103] proposed an actuator excitation method which tests the triggered alarms to confirm the alarms or remove them. Another effective solution is the reinitialization of the elemental filters.

For the multiple-model estimation, how to reinitialize each single-model-based filter is very important [99]. As for the traditional MMAE, each elemental filter uses its own initial state estimate and covariance of state estimate error for reinitialization [99]. All the filters are running in parallel without interactions with each other [99, 104]. The IMM approach [35, 99, 105] is another multiple model-based approach in which all the

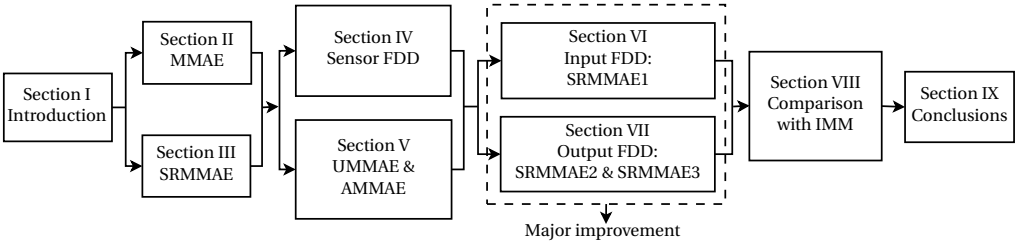


Figure 3.1: Structure of the chapter

elemental filters interact with each other. Each filter is reinitialized at every time step using the mixture of the state estimates from the single-model-based filters [99]. However, the IMM usually has a degraded FDD performance as a result of its tendency to enhance the state estimation ability in order to track the objectives [106].

In this chapter, a novel SRMMAE approach is proposed to solve the aforementioned problems. It uses the UKF [27, 86] as a local filter in order to improve the state estimation performance. Input and output faults are augmented as additional states, which reduces false alarms. The major innovation of this approach is that it proposes three SR algorithms which are able to improve the FDD performance of the MMAE significantly. The MMAE using the three SR algorithms are called SRMMAE1, 2 and 3 respectively. The SRMMAE1 is designed for input FDD whereas the SRMMAE2 and SRMMAE3 are designed for output FDD. The proposed approach is able to solve the problem of false alarms and slow response to detect that faults have been removed, without actuator excitation. The input and output FDD performance of this novel approach is compared to the MMAE and the IMM approach with the objective of the IMU sensor and ADS FDD of a Cessna Citation II CE-550 aircraft model. The SRMMAE2 and SRMMAE3 are able to identify the output faults while the fault estimation of the augmented MMAE diverges.

The structure of this chapter is shown in Figure 3.1. Section 3.2 gives the preliminaries on the MMAE approach for FDD. Section 3.3 presents the concept of the SRMMAE approach as well as the difference within the multiple model-based approaches. In Section 3.4 the robust FDD of the aircraft sensors is described. The Unscented Multiple-Model Adaptive Estimation (UMMAE) and the Augmented Multiple-Model Adaptive Estimation (AMMAE) which augments the input and output faults as additional states are presented in Section 3.5. In Section 3.6, the SRMMAE1 is proposed for the input FDD while in Section 3.7, the SRMMAE2 and SRMMAE3 are proposed for the output FDD. The performance of the three algorithms is compared to the UMMAE and AMMAE. In Section 3.8, the performance of the three algorithms is compared to the IMM approach. Finally, the conclusions are made in Section 3.9.

3.2. MULTIPLE-MODEL ADAPTIVE ESTIMATION FOR FDD

This section presents some preliminaries related to this research.

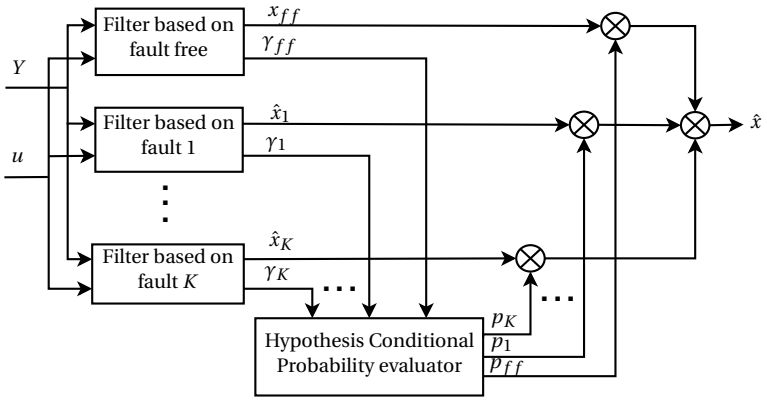


Figure 3.2: Block diagram for the MMAE approach

3.2.1. NONLINEAR SYSTEM MODEL WITH INPUT AND OUTPUT FAULTS

Consider the following nonlinear state-space model with input and output faults

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t),t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{f}^i(t) & (3.1) \\ \boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t),\boldsymbol{u}(t),t) + \boldsymbol{v}(t) + \mathbf{F}(t)\boldsymbol{f}^o(t), \qquad t = t_i, \qquad i = 1,2,\ldots & (3.2) \end{cases}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the system states, $\boldsymbol{u} \in \mathbb{R}^b$ the input, $\boldsymbol{y} \in \mathbb{R}^d$ the measurement. The functions $\boldsymbol{f}^i \in \mathbb{R}^b$ and $\boldsymbol{f}^o \in \mathbb{R}^m$ represent input and output faults respectively. The input noise vector $\boldsymbol{w}$ and output noise vector $\boldsymbol{v}$ are defined as follows:

$$E[\boldsymbol{w}(t)] = \mathbf{0},\ E[\boldsymbol{w}(t)\boldsymbol{w}^T(\tau)] = \mathbf{Q}\delta(t-\tau) \quad (3.3)$$

$$E[\boldsymbol{v}(t_i)] = \mathbf{0},\ E[\boldsymbol{v}(t_i)\boldsymbol{v}^T(t_j)] = \mathbf{R}\delta'_{ij} \quad (3.4)$$

$$E[\boldsymbol{w}(t)\boldsymbol{v}^T(t_i)] = \mathbf{0}\ ,\ t = t_i\ ,\ i,j = 1,2,\ldots \quad (3.5)$$

where $\mathbf{Q}$ and $\mathbf{R}$ are the covariance matrix of the input noise and output noise respectively. δ and δ' are the Dirac function and Kronecker delta function, respectively.

3.2.2. MULTIPLE-MODEL ADAPTIVE ESTIMATION

The MMAE [37] is composed of a bank of Kalman filters operating in parallel. All the filters use a vector of measurements $\boldsymbol{Y}$ explained below and a vector of input $\boldsymbol{u}$, and are based on the same equation of motion while each hypothesizes a different fault scenario. At sample step k, each of the filters produces a state estimate $\hat{\boldsymbol{x}}(k)$ and a vector of innovations $\boldsymbol{\gamma}(k)$. The basic idea is that the Kalman filter which produces the most well-behaved innovation contains the model which matches the true faulty model best. The block diagram for the MMAE is given in Figure 3.2.

A hypothesis test uses the vector of innovations $\boldsymbol{\gamma}(k)$ and the residual covariance matrices $\mathbf{A}(k)$ of the filters in order to assign a conditional probability to each of the filters which matches a particular fault scenario. Let a denote the fault scenarios of the nonlinear system and K the number of the filters of the MMAE. If we define the hypothesis conditional probability $p_i(k)$ as the probability that a is assigned the value a_i for

$i = 1, 2, ..., K$, conditioned on the measurement history up to time step k:

$$p_i(k) = \Pr[a = a_i | \boldsymbol{Y}_k], \quad i = 1, 2, ..., K \tag{3.6}$$

Then the conditional probability can be updated recursively using the following equation:

$$p_i(k) = \frac{f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i, \boldsymbol{Y}_{k-1}) p_i(k-1)}{\sum_{j=1}^{K} f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_j, Y_{k-1}) p_j(k-1)}, \quad i = 1, 2, ..., K \tag{3.7}$$

In Equation (3.7), $\boldsymbol{Y}_{k-1}$ is the measurement history vector which is defined as

$$\boldsymbol{Y}_{k-1} = \{\boldsymbol{y}(0), \boldsymbol{y}(1), .., \boldsymbol{y}(k-1)\},$$

$f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i, \boldsymbol{Y}_{k-1})$ is the density function which is defined as follows:

$$f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i, \boldsymbol{Y}_{k-1}) = \beta_i(k) e^{-(\boldsymbol{\gamma}^T(k)\mathbf{A}_i(k)^{-1}\boldsymbol{\gamma}(k))/2} \tag{3.8}$$

where

$$\beta_i(k) = \frac{1}{(2\pi)^{d/2}|\mathbf{A}_i(k)|^{1/2}} \tag{3.9}$$

In Equation (3.9), d is the dimension of the measurement, the symbol "|...|" denotes the determinant of the innovation covariance matrix $\boldsymbol{A}_i(k)$ which is computed by the Kalman filter at time step k. The filter which matches the fault scenario produces the smallest innovation which is the difference between the estimated measurement and the true measurement. Therefore, the conditional probability of the filter which matches the true fault scenario is the highest among all the filters. After the computation of the conditional probability, the state estimation of the nonlinear system can be generated by the weighted state vector of the K filters.

$$\hat{\boldsymbol{x}}(k) = \sum_{i=1}^{K} \hat{\boldsymbol{x}}_i(k) p_i(k) \tag{3.10}$$

3.2.3. MMAE FOR FDD

The model probability indicates to what extent each model matches the system model; hence, it can be used for FDD in real time. Normally, the alarm for the ith fault is triggered if the associated fault model probability exceeds a certain percentage p_u for a certain amount of time and is not triggered if it is below p_l for a certain amount of time, which is described as follows:

$$p_i(k) \begin{cases} > p_u \Rightarrow \text{alarm for the } i\text{th fault is triggered} \\ < p_l \Rightarrow \text{alarm for the } i\text{th fault is not triggered} \end{cases} \tag{3.11}$$

where p_u and p_l can be chosen to be 0.9 and 0.05 respectively according to Ducard et al. [103].

If the estimation of the fault performed by the ith corresponding filter is denoted as $\hat{f}_i(k)$, then its probability-weighted fault estimation can be denoted as follows:

$$\bar{f}_i(k) = \hat{f}_i(k) p_i(k) \tag{3.12}$$

where $\bar{f}_i(k)$ is the probability-weighted fault estimation.

In this chapter, we call the filter based on the fault-free model as "fault free filter" and the filter based on the faulty model as "fault filter".

3.3. SELECTIVE-REINITIALIZATION MULTIPLE MODEL ADAPTIVE ESTIMATION

As for the multiple model-based approaches, the gains of the filters based on the mismatched models decrease monotonically with time. Consequently, an abrupt fault may not be immediately reflected in a corresponding change in the filter estimates, even if a lower bound on the probabilities is set [99]. Therefore, reinitialization of the elemental filters is important.

For the traditional MMAE, an elemental filter is reset to its initial condition [99] or only restarted when there is an indication of divergence [106]. All the filters operate independently without interaction with each other [104]. This may lead to detection delay or missed detection for subsequent faults [99]. In contrast, the IMM approach reinitializes all the elemental filters at every step using the mixture of the estimates from the filters. This mixing process of the IMM approach weakens its ability to detect the fault despite of the enhancement of its state estimation performance.

The SRMMAE approach proposes a SR scheme which results in an interaction in between. It requires more reinitializations than the traditional MMAE. When there are faults, the SRMMAE keeps reinitializing the fault free filter whereas the traditional MMAE just resets or restarts its filters. The similarity between the SRMMAE and the IMM is that the elemental filters interact with each other. The main differences between these two approaches are as follows:

1. They use different reinitialization frequencies
   The IMM performs the reinitialization at every step whereas the SRMMAE only reinitializes for certain time steps according to different SRMMAE algorithms.

2. They use different reinitialization algorithms
   The IMM reinitializes the filters using their mixed state estimates whereas the SRMMAE does not use the mixed estimates. Furthermore, the IMM reinitializes all the filters whereas the SRMMAE reinitializes the filters selectively.

How to perform the SR and the advantage of performing this SR will be demonstrated in the following sections. Furthermore, the advantages of this SRMMAE compared to the MMAE and IMM will also be demonstrated.

3.4. ROBUST AIRCRAFT SENSOR FDD

This chapter deals with the FDD of aircraft sensors. Normally, there are three categories of sensors on civil airplanes: IMU sensors, ADS and Global Positioning System (GPS). The ADS provide information about e.g. the true airspeed V, angle of attack α and angle of sideslip β. The IMU sensors measure the specific forces A_x, A_y, A_z and rotational rates p, q, r of the aircraft. The GPS receiver provides three dimensional position x, y, z and

ground velocity components u_n, v_n, w_n. Without losing generality, only the faults of the IMU sensors and ADS are considered.

In order to achieve robustness to model uncertainties due to the calculation of the aerodynamic forces and moments, kinematic model [59, 82, 90, 107] is used instead of the aerodynamic equations. This is important since the performance of model-based approaches is sensitive to the model uncertainties. Moreover, this approach also reduces the computational load.

The currently used kinematic equations do not incorporate the effect of the turbulence. However, this can be solved using the MMAE approach by adding an additional filtering stage to achieve robustness in the event of the turbulence [103]. Furthermore, this is beyond the scope of this chapter.

3.4.1. AIRCRAFT MODEL USING THE KINEMATIC EQUATIONS

The kinematic model of the aircraft including the IMU sensor and ADS faults is described as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t\big) + \mathbf{G}(x(t))\boldsymbol{w}(t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{f}^i(t) \tag{3.13}$$

$$\boldsymbol{y}_m(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \boldsymbol{v}(t) + \mathbf{F}(t)\boldsymbol{f}^o(t), \quad t = t_i,\ i = 1, 2, \ldots \tag{3.14}$$

The system equation variables are defined as follows:

$$\boldsymbol{x} = [V\ \alpha\ \beta\ \phi\ \theta\ \psi]^T \tag{3.15}$$

$$\boldsymbol{u}_m = \boldsymbol{u} + \boldsymbol{w} + \boldsymbol{f}^i = [A_x\ A_y\ A_z\ p\ q\ r\ ]^T + \boldsymbol{w} + \boldsymbol{f}^i \tag{3.16}$$

$$\boldsymbol{y}_m = [V_m\ \alpha_m\ \beta_m\ \phi_m\ \theta_m\ \psi_m]^T \tag{3.17}$$

$$\boldsymbol{w} = [w_x\ w_y\ w_z\ w_p\ w_q\ w_r\ ]^T \tag{3.18}$$

$$\boldsymbol{v} = [v_V\ v_\alpha\ v_\beta\ v_\phi\ v_\theta\ v_\psi]^T \tag{3.19}$$

$$\boldsymbol{f}^i = [f_{Ax}\ f_{Ay}\ f_{Az}\ f_p\ f_q\ f_r\ ]^T \tag{3.20}$$

$$\boldsymbol{f}^o = [f_V\ f_\alpha\ f_\beta]^T \tag{3.21}$$

where the input fault $\boldsymbol{f}^i$ is the fault of the IMU sensors. $[f_V,\ f_\alpha,\ f_\beta]^T$ is the fault of the ADS. $\boldsymbol{w}$ and $\boldsymbol{v}$ are the input and output white Gaussian noise, respectively. $\boldsymbol{u}_m$ is the measured input and $\boldsymbol{y}_m$ is the measured output.

The kinematic equations can be derived as follows:

$$\dot{V} = (A_x - g\sin\theta)\cos\alpha\cos\beta + (A_y + g\sin\phi\cos\theta)\sin\beta + (A_z + g\cos\phi\cos\theta)\sin\alpha\cos\beta \tag{3.22}$$

$$\dot{\alpha} = \frac{1}{V\cos\beta}(-A_x\sin\alpha + A_z\cos\alpha + g\cos\phi\cos\theta\cos\alpha + g\sin\theta\sin\alpha) + q - (p\cos\alpha + r\sin\alpha)\tan\beta \tag{3.23}$$

$$\dot{\beta} = \frac{1}{V}[-(A_x - g\sin\theta)\cos\alpha\sin\beta + (A_y + g\sin\phi\cos\theta)\cos\beta - (A_z + g\cos\phi\cos\theta)\sin\alpha\sin\beta] + p\sin\alpha - r\cos\alpha \tag{3.24}$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{3.25}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{3.26}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{3.27}$$

For detailed derivations, please refer to Appendix B. From the above equations, the noise distribution matrix $\mathbf{G}(\boldsymbol{x}(t))$, which is linear in $\boldsymbol{w}(t)$, can be derived as follows:

$$\mathbf{G}(\boldsymbol{x}(t)) = \begin{bmatrix} -c\alpha c\beta & -s\beta & -s\alpha c\beta & 0 & 0 & 0 \\ s\alpha/(Vc\beta) & 0 & -c\alpha/(Vc\beta) & c\alpha t\beta & -1 & s\alpha t\beta \\ c\alpha s\beta/V & -c\beta/V & s\alpha s\beta/V & -s\alpha & 0 & c\alpha \\ 0 & 0 & 0 & -1 & -s\phi t\theta & -c\phi t\theta \\ 0 & 0 & 0 & 0 & -c\phi & s\phi \\ 0 & 0 & 0 & 0 & -s\phi/c\theta & -c\phi/c\theta \end{bmatrix} \tag{3.28}$$

where s(•), c(•) and t(•) denote the trigonometric functions sin(•), cos(•) and tan(•) respectively. The measurement model can be rewritten as follows:

$$\boldsymbol{y}_m = \mathbf{H}\boldsymbol{x} + \boldsymbol{v} + \mathbf{F}\boldsymbol{f}^o \ , \ \mathbf{H} = \mathbf{I}_{6\times 6}, \mathbf{F} = [\mathbf{I}_{3\times 3}\ \mathbf{O}_{3\times 3}]^T \tag{3.29}$$

Remarks:

1. It can be seen from the above-mentioned equations that the IMU sensor measurements serve as the input of the system while the ADS provide the output measurements. Under these conditions, the faults of the IMU sensors are the input faults while the faults of the ADS are the output faults denoted in the model Equation (3.13) and Equation (3.14).

2. In the kinematic equations, the physical actuators such as the elevator, rudder and aileron do not appear in the equations. This is also advantageous since the FDD of the actuators and sensors are separated. The result of the FDD of the IMU sensors and ADS does not affect the FDD of the actuators.

3.4.2. SELECTION OF Q AND R

The **Q** and **R** defined in Equation (3.3) and Equation (3.4) respectively are critical to the performance of the Kalman filter. **R** can be inferred from the output measurement noise. In this chapter, it is obtained from the noise of the ADSs and the sensors which measure the attitude angles. **Q** is usually used to represent modeling errors [87, 108]. As can be seen from the kinematic equations, the only model uncertainty is in the input (i.e. the IMU measurements). Therefore, **Q** can be inferred from the IMU sensor noise.

3.5. THE UMMAE AND AMMAE FOR INPUT AND OUTPUT FDD

This section introduces the UMMAE and AMMAE whose performance will be compared to the proposed approach. They will be applied to the problem described in the last section.

3.5.1. UMMAE FOR FDD

The MMAE approach usually uses the KF or EKF as the elemental filters. However, in this chapter, the UKF is used in order to improve state estimation performance. As such, all the filters in Figure 3.2 are replaced by UKF. For detailed information about the UKF, the reader is referred to Julier et al. [27, 109] and van der Merwe et al. [86].

DISCUSSIONS ON IMPLEMENTATION

The state dynamics presented in Equation (3.1) and Equation (3.13) are modeled as continuous-time processes. The measurement model given in Equation (3.2) and Equation (3.14) are modeled as discrete-time processes. This results in a continuous-discrete system.

The discrete-time UKF [27] combined with Euler approximation is able to perform the estimation of the continuous-discrete time system as done in Julier et al. [27].

Another solution is to use the Continuous-Discrete UKF [110] which is a continuous-discrete form of the UKF designed for continuous-discrete systems.

3.5.2. AMMAE FOR FDD

The MMAE approach is able to perform state estimation and parameter estimation at the same time and can detect and isolate the faults in the system quickly [37]. In this section, the performance of the MMAE with respect to the fault estimation will be discussed.

MMAE FOR FAULT ESTIMATION

The MMAE approach has a satisfactory performance in state estimation. However, if the MMAE approach is used to identify the faults, additional models are needed. Normally, multiple model-based methods use one fault free model and at least one faulty model to estimate the partial faults of each component such as the input and the output of the aircraft model. For example, if there are K faults, then we need at least $2K+1$ models (one fault free model, K total fault models and K partial fault models) to model the total and partial faults [99]. The more models representing different fault values are used, the more accurate the estimation of the faults will be. However, it is computationally intensive. To avoid this, Maybeck and his research team proposed the Moving-Bank MMAE [36, 102] and Hierarchical MMAE [37]. The Hierarchical MMAE divides the FDD into

two levels: the fault detection and fault estimation. After the detection of the fault, the MMAE method enters the next level-fault estimation. Although this method only runs $K+1$ models online, $K(K+1)+1$ models have to be designed offline. Ru [111] used three models ($a = 0, 0.5, 1$, a is the fault values of the model which matches the faulty model) in the second level for each component. However, $3K+1$ models are still required. Ducard [103] proposed to augment the deflection of the actuators of the aircraft in order to eliminate the effect of the biased estimation when using the MMAE method to tackle the fault scenarios that the actuators are locked at nonzero positions.

In this chapter, the augmentation strategy is introduced to both input and output FDD. This approach is called AMMAE. For the model which matches the input faults, the faults of the input are treated as additional states of the filter and the same for the model which matches the output faults. There may be problems when augmenting the output faults as additional states since the estimation of the augmented output faults may not be unbiased. However, it can be solved by the proposed SRMMAE approach, which will be shown in the following section.

AMMAE FOR INPUT FDD

Consider the system model described by Equation (3.1) and Equation (3.2). The dimension of the input fault is b. As for the input FDD, $b+1$ models are needed. b models are the models matching the b input faults and the other model is the fault free model.

The states of the fault free model are denoted as

$$\boldsymbol{x}_{ff}(k) = [x_{ff,1}(k),\ x_{ff,2}(k),\ \dots,\ x_{ff,n}(k)]^T \tag{3.30}$$

The states of the b fault models are

$$\boldsymbol{x}_j(k) = [x_{ff,1}(k),\ x_{ff,2}(k),\ \dots,\ x_{ff,n}(k),\ f_j^i(k)]^T, \quad j = 1,2,\dots,b \tag{3.31}$$

where $f_j^i(k)$ is the jth input fault $\boldsymbol{f}^i$ at time step k. Note that we only augment the input fault which is related to the fault model in the jth model rather than all the input faults.

AMMAE FOR OUTPUT FDD

As for the output FDD, $m+1$ models are needed. m models are the models matching the output faults and the other model is the fault free model.

The states of the fault free model are the same as for the fault free model for the input FDD, as described in Equation (3.30). The states of the m fault models are

$$\boldsymbol{x}_l(k) = [x_{ff,l}(k),\ x_{ff,2}(k),\ \dots,\ x_{ff,n}(k),\ f_l^o(k)]^T, \quad l = 1,2,\dots,m \tag{3.32}$$

where $f_l^o(k)$ is the lth output fault $\boldsymbol{f}^o$ at time step k. Similar to the input FDD, only the output fault related to the lth output fault model will be augmented and estimated.

3.6. THE SRMMAE1 ALGORITHM FOR INPUT FDD

Section 3.6 and Section 3.7 will present the three SR algorithms (the SRMMAE1, SRMMAE2 and SRMMAE3) which are the major improvement of the MMAE. The SRMMAE1

algorithm is designed for input FDD while the SRMMAE2 algorithm and the SRMMAE3 algorithm are two approaches which can both be applied to output FDD.

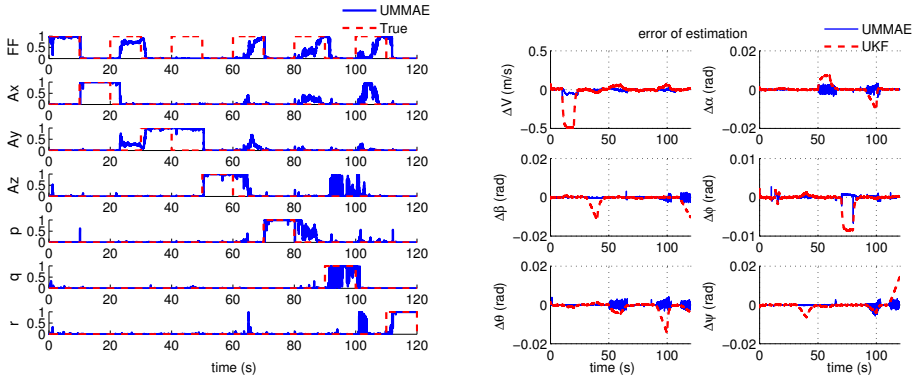
The objective of this section is the input (the IMU sensors) FDD. The aircraft model used for demonstration is the Cessna Citation II CE-550 aircraft which is owned by Delft University of Technology and the Dutch Aerospace Laboratory.

First, the performance of the IMU sensor FDD using the UMMAE and AMMAE will be shown. The drawbacks of these two algorithms leads to the introduction of the SRMMAE1 algorithm.

3.6.1. UMMAE FOR IMU SENSOR FDD

In this simulation, the UMMAE is used to tackle the IMU FDD. A consecutive sequence of faults is generated. All the IMU sensors will be faulty for a certain amount of time. For 10 s $< t <$ 20 s, A_x sensor fails with a bias. For 30 s $< t <$ 40 s, there is a drift fault in the A_y sensor. For 50 s $< t <$ 60 s, a bias occurs in the A_z sensor. For 70 s $< t <$ 80 s, a bias occurs in the roll rate sensor with a value of 0.5/57.3 rad/s. For 90 s $< t <$ 100 s, and 110 s $< t <$ 120 s, there are drift faults in the pitch rate and yaw rate sensor with the same drift rate of 0.002 rad/s^2. The simulation time step is 0.01 s in this chapter.

There are six IMU sensor faults which are all input faults. Using the UMMAE approach, seven filters are required for the six fault models and the fault free model. The result of using the UMMAE is shown in Figure 3.3.



(a) Probabilities of the seven models using the UMMAE

(b) State estimation error compared to the UKF

Figure 3.3: Results of the IMU sensor FDD using the UMMAE approach and the comparison with the UKF

Figure 3.3(a) shows the probability of the seven filters for different fault models while Figure 3.3(b) compares the state estimation error of the UMMAE approach with that of the UKF. In Figure 3.3(a), the label "FF" represents the fault free filter while the others represent the filters matching the fault model of the accelerometers and the rate gyros. The dashed lines represent the true probability of different models while the solid lines show the probability derived by the UMMAE approach.

It can be seen from the figure that the detection of the faults by the UMMAE approach is quick. To detect the fault of the A_x, A_z and p sensors takes 0.3 s, 0.45 s and 0.32 s

respectively. However, it takes longer to detect the fault of the A_y, q and r sensors which is 1.5 s, 1.4 s and 2 s respectively. This is due to the fact that the latter faults are drift faults with a small drift rate which can be detected when the value of the fault reaches a certain level. The drawbacks of using this method are concluded from the figure as follows:

1. False alarms

   When the q sensor is faulty during 90 s $< t <$ 100 s, alarms are also reported by the filter for the A_z sensor fault. This may be caused by the similar effect of the two sensors faults which may both lead to a biased estimation of the pitch angle θ and the angle of attack α. In this case, the UMMAE approach is not able to distinguish the fault of the A_z sensor from that of the q sensor. Other false alarms are also observed such as the false alarm for the A_x sensor during 102 s $< t <$ 108 s when there are no IMU faults in the system.

2. Slow response to detect the removal of the faults

   The A_x sensor is faulty during 10 s $< t <$ 20 s. However, after the fault is removed at $t = 20$ s, the probability of the model for the A_x sensor fault stays at one for another 3.5 s, which means it takes 3.5 s to detect the fact that the A_x sensor fault has been removed. The same problem exists for other fault models, especially the A_y sensor. Its probability stays at one until the next fault (A_z sensor fault) is injected to the system. In this chapter, the detection of the removal of the faults is define as "slow (quick)" if it takes more (less) than 1 s.

3. Difficulty to identify the faults

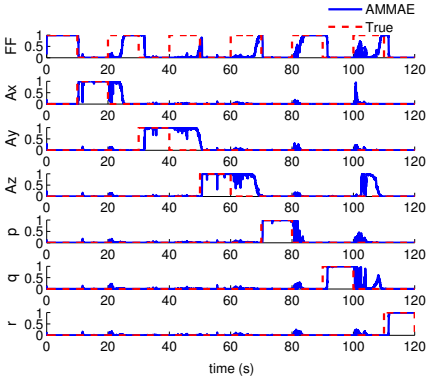
   As mentioned, if the UMMAE is used to identify the fault, additional models to estimate the parameter of the fault have to be designed.

Figure 3.3(b) compares the state estimation error of the UMMAE approach and that of the normal UKF. Obviously, the state estimation error is reduced by the UMMAE approach even in the presence of the faults, which confirms its good state estimation performance. However, it is noticed that the error of estimation experiences significant oscillations during the occurrence of the faults which are caused by the false alarms. Note that the biased estimation of the states can be caused by different sensors. For example, the estimation of the angle of attack can be influenced by both the A_z sensor fault and the the q sensor fault.

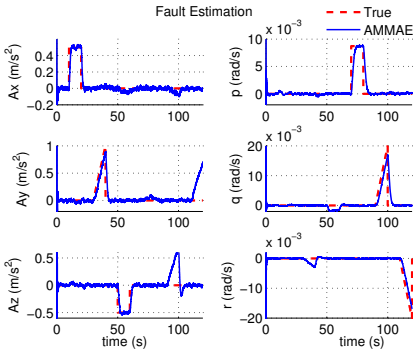
3.6.2. AMMAE FOR IMU SENSOR FDD

From the previous simulation, it can be seen that the UMMAE approach can not be used to perform the FDD directly. In this section, the AMMAE is used to detect and identify the IMU sensor faults. The faults of the IMU sensors are augmented as additional states. By doing this, the faults can be estimated in an unbiased sense. The fault scenario is the same with the previous simulation in order to compare with the performance of the UMMAE approach.

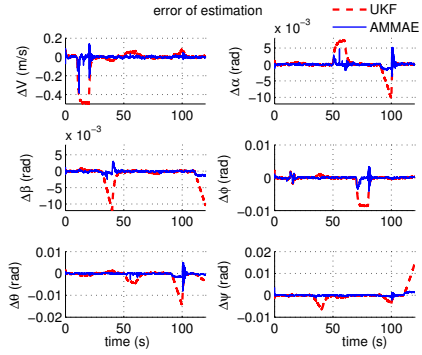
Figure 3.4 shows the performance of the AMMAE approach in comparison with the UKF and the UMMAE approach. Figure 3.4(a) shows the probability of the seven models



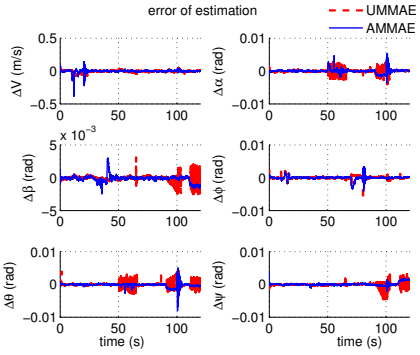
(a) Model probabilities of the seven models



(b) IMU sensor fault estimation of the fault filters



(c) State estimation error compared to the UKF



(d) State estimation error compared to the UMMAE

Figure 3.4: Results of the IMU sensor FDD using the AMMAE approach and the comparison with UKF and UMMAE approach

matching seven fault scenarios. It is seen that the false alarms during the fault have been reduced except that during 100 s < t < 110 s when there are no faults in the IMU sensors, the fault filters for the A_x, A_z and q still give false alarms. In addition, when the q sensor fails during 90 s < t < 100 s, the filter for A_z sensor fault no longer gives false alarms.

More importantly, the shape of the IMU faults can be estimated by the corresponding filters as shown in Figure 3.4(b). Note that faults of one IMU sensor may influence the fault estimation of another IMU sensor. For example, both the filter for A_z and q sensors will give a non-zero fault estimation when either of them fails, which can be seen during 50 s < t < 60 s and 90 s < t < 100 s. However, that is not an issue as long as the probability for the wrong fault estimation is close to zero. Through Equation (3.12), the incorrect fault estimation has no influence on the fault estimation of the AMMAE. From Figure 3.4(c) and Figure 3.4(d), the performance of the AMMAE approach is compared to the normal UKF and the UMMAE approach. Undoubtedly, the performance of the AMMAE is better than the other two methods. The oscillations of the estimation error, which occur using the UMMAE approach, is reduced to a great extent, although some of them still exist in the presence of the faults.

The problem of the slow detection of the removal of the faults, remains unsolved. When every fault is removed, it takes some time for the fault free model to reach the probability of one thus giving false alarms. For example, it takes the fault free model 10 s to reach a model probability of one after the fault of the A_y sensor is removed at t = 40 s. The solution to this problem will be given in the following subsection.

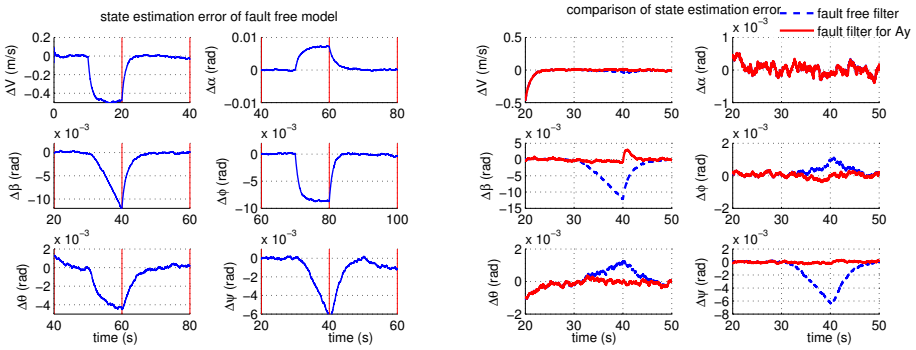
3.6.3. SRMMAE1 FOR IMU SENSOR FDD

It was found in the previous subsection that the AMMAE is not able to quickly detect the removal of the faults after the IMU sensor faults are removed, which causes false alarms.

It can be observed from the probability plots (Figure 3.4(a)) of different models that after each fault is removed, the fault model remains at a high probability which approaches 1 for a certain amount of time. This means that the fault model matches the true model more than the fault free model, which actually is not true. This can be explained by checking the state estimation error of the fault free model given in Figure 3.5(a). It can be seen that after the removal of the fault at t =20 s, 40 s, 60 s, 80 s and 100 s, the estimated state of the filter related to the fault free model can not track the real state immediately.

The state estimation of the fault free filter and the filter matching the A_y sensor fault are considered here without losing generality. The comparison of the results is shown in Figure 3.5(b). Recall that the fault of the A_y sensor occurs at t = 30 s and is removed at t = 40 s, the result for 20 s < t < 50 s is shown. The fault of the A_y sensor mainly influences the estimation of the angle of sideslip β and the yaw angle ψ as can be seen from the estimation of the fault free filter.

Let the subscript "ff" and "A_y" denote the quantities pertaining to the fault free filter and the filter for the A_y sensor FDD, respectively. Let t_f denote the time period when there is an A_y sensor fault and t_r denote the time period which is after the fault is removed and before the fault free filter achieves a comparable performance with the A_y sensor fault filter (which is roughly between t = 40 s and t = 45 s in this case). During t_f, since the fault free filter does not augment and estimate the fault, its state estima-



(a) State estimation error of the fault free filter after the occurrence of the IMU sensor faults

(b) State estimation error of the fault free filter compared to the filter for A_y sensor fault

Figure 3.5: State estimation error of the fault free filter using the AMMAE approach for the IMU sensor FDD

tion $\hat{x}(k+1|k)$ is biased while the fault filter for A_y sensor faults achieves an unbiased estimation. This explains the result shown in the figure during t_f. However, during t_r, when there are no faults in the system, the fault free filter still obtains a biased estimation while the fault filter for A_y sensor faults maintains an unbiased estimation which is demonstrated in the figure. The reason is that the UKF needs time to adapt to the new situation when there are no faults. The a priori estimate error of the filter is

$$\tilde{\boldsymbol{x}}(k+1|k) = \boldsymbol{x}(k+1) - \hat{\boldsymbol{x}}(k+1|k) \tag{3.33}$$

where $\hat{\boldsymbol{x}}(k+1|k)$ is the a priori estimate of the filter. Since $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{x}$ and there are no output faults ($\boldsymbol{f}^o(k+1) = \boldsymbol{0}$) in this simulation, the innovation of each filter is

$$\begin{aligned} \boldsymbol{\gamma}(k+1) &= \boldsymbol{h}(\boldsymbol{x}(k+1)) - \boldsymbol{h}(\hat{\boldsymbol{x}}(k+1|k)) + \boldsymbol{v}(k+1) \\ &= \tilde{\boldsymbol{x}}(k+1) + \boldsymbol{v}(k+1). \end{aligned} \tag{3.34}$$

It follows from $\tilde{\boldsymbol{x}}_{ff}(k+1) > \tilde{\boldsymbol{x}}_{A_y}(k+1)$ and $\boldsymbol{v}_{ff}(k+1) = \boldsymbol{v}_{A_y}(k+1)$ that

$$\boldsymbol{\gamma}_{ff}(k+1) > \boldsymbol{\gamma}_{A_y}(k+1). \tag{3.35}$$

According to Ormsby [101], $\boldsymbol{\gamma}(k+1)$ is the dominant factor in Equation (3.8). It follows

$$f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_{ff},\boldsymbol{Y}_{k-1}) < f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_{A_y},\boldsymbol{Y}_{k-1}) \tag{3.36}$$

where a denotes the fault scenario in the system. Thus, it can be concluded from Equation (3.7) that

$$p_{ff} < p_{A_y}. \tag{3.37}$$

This explains why the probability of the fault model stays high even after the removal of the faults.

To cope with this drawback, we propose to reinitialize the fault-free filter after the fault is removed. However, when and how to reinitialize the fault free model has to be

Algorithm 1 SRMMAE1 Algorithm for Input FDD

At time step k;

1. **Variable assignment**
   $p_{t,i} = p_i, \quad i = 1,2,...,b$
   $\hat{x}_{t,i} = \{\hat{x}_{i,j} \mid j = 1 \text{ to } n+1\}, \quad i = 1,2,...,b$
   $P_{t,i} = \{P_{i,jj} \mid j = 1 \text{ to } n+1\}, \quad i = 1,2,...,b$
2. **Indexes of the filters with the maximal or minimal model probability**
   Find $i_{min,k} \in \{ p_{t,i_{min,k}} = \min(p_{t,i}) \mid i = 1,2,...,b+1 \}$
   Find $i_{max,k} \in \{ p_{t,i_{max,k}} = \max(p_{t,i}) \mid i = 1,2,...,b+1 \}$
3. **Reinitialization**
   if $p_{ff} < p_{t,i_{min,k}}$ **or** $p_{ff} = p_{t,i_{min,k}}$ **then**
   $\hat{\boldsymbol{x}}_{ff} \leftarrow \hat{\boldsymbol{x}}_{t,i_{max,k}}$
   $\mathbf{P}_{ff} \leftarrow \mathbf{P}_{t,i_{max,k}}$
   end if

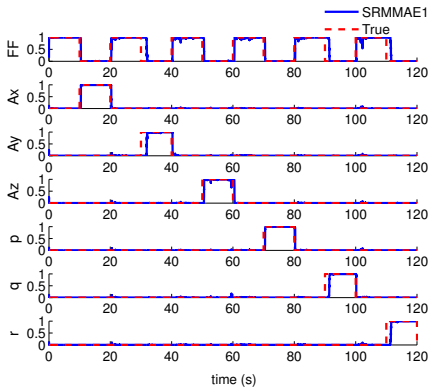
determined. In this section, the first SR algorithm (the SRMMAE1) is proposed which is able to cope with the input FDD. The details are shown in Algorithm 1.

In the SRMMAE1 algorithm, b, described in the system model in Equation (3.1), denotes the dimension of the input faults which is also the number of fault models matching each input fault scenario, n is the dimension of the state as also described in Equation (3.1). The subscript "ff" denotes the parameters of the fault free filter while the subscript "i" describes the parameters of the filter for b different fault filters, e.g., $\hat{\boldsymbol{x}}_{ff}$ and $\mathbf{P}_{ff}$ denote the a posteriori state estimate and the error covariance matrix of the fault-free filter. $\hat{\boldsymbol{x}}_i$ and $\mathbf{P}_i$ ($i = 1,2,...,b$) are the a posteriori state estimates and the error covariance matrices of the fault filters. The variable p_t, $\hat{\boldsymbol{x}}_t$ and $\mathbf{P}_t$ are the sets which contain the model probability, state estimate and covariance of state estimate error of the elemental filters respectively.

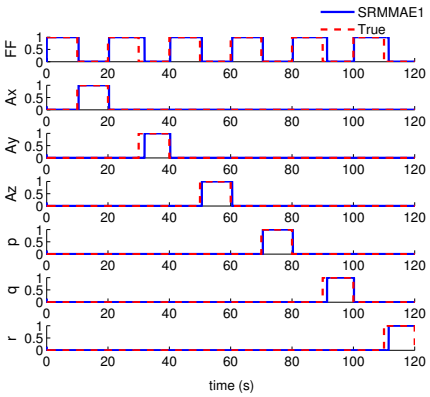
The algorithm proposes to reinitialize the state estimate and the covariance matrix of the fault free filter only when the probability of the fault free model is smaller than or equal to the fault model with the smallest model probability. The reinitialization is done using the a posteriori state estimate and the covariance matrix of the fault filter with the biggest model probability. This algorithm is applied to tackle the input FDD problem.

Figure 3.6 presents the results of the proposed SRMMAE1. From Figure 3.6(a), it can be seen that the improvement by the SRMMAE approach is significant. All the problems mentioned above are solved thus enhancing the robustness of this approach to a great extent. It is apparent that each fault is detected and isolated correctly and quickly. No false alarms are observed from the probability figure. If one notices carefully, there are some probabilities of the fault free model does not approach 1 when there are no IMU sensor faults, such as at time t =42 s and t =111.6 s. This does not influence the FDD result since the alarm of the fault will only be triggered when the probability falls below a certain level for a certain period of time [99]. Moreover, this can be improved by applying an additional filtering stage proposed by Ducard [103]. The result is shown in Figure 3.6(b).

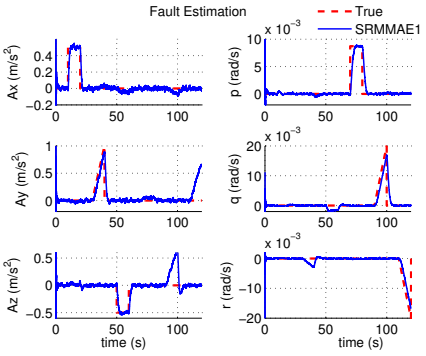
Since the SRMMAE approach preserves the fast detection ability as well as the fault



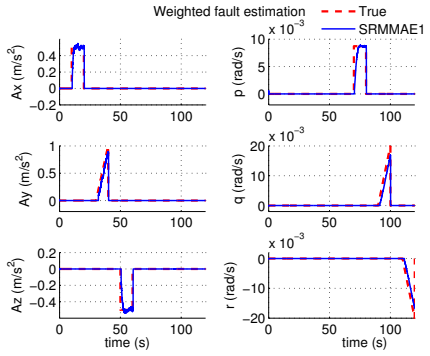
(a) Model probabilities of the seven filters



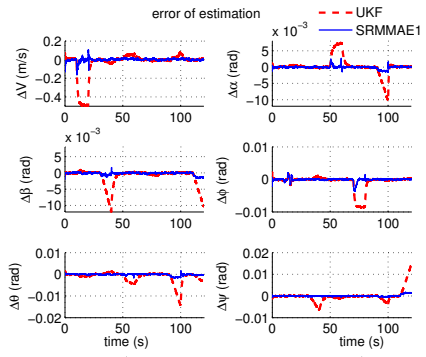
(b) Filtered model probabilities for the seven filters



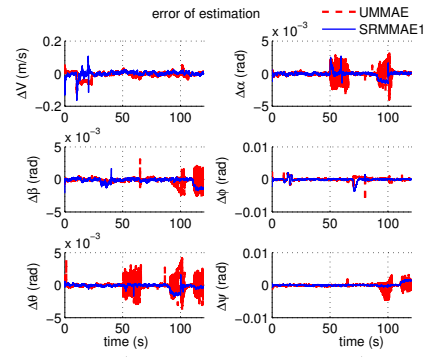
(c) IMU sensor fault estimation of the six fault filters



(d) Probability-weighted IMU sensor fault estimation of the six fault filters



(e) State estimation error compared to the UKF



(f) State estimation error compared to the UMMAE

Figure 3.6: Results of the IMU sensor FDD using the SRMMAE1 approach and the comparison with the UKF and UMMAE approach

estimation ability of the AMMAE approach while introducing the SR algorithm, it can identify the fault more accurately as shown in Figure 3.6(c). The probability-weighted fault estimation of the system using Equation (3.12) is shown in Figure 3.6(d).

The performance of the SRMMAE approach compared to the normal UKF and the UMMAE approach is given in Figure 3.6(e) and Figure 3.6(f), respectively. As can be seen from these two figures, the state estimation of the SRMMAE is close to zero-mean even in the presence of the IMU sensor faults whereas the estimation of the UKF and the UMMAE approach show significant deviations or frequent oscillations.

Remarks: The proposed SRMMAE approach for input FDD does not require input excitation or performing a maneuver to enhance the ability to quickly detect the removal of the faults. As can be seen from the algorithm, the fault free filter is not reinitialized all the time, which is good for the computational load.

3.7. THE SRMMAE2 AND SRMMAE3 FOR OUTPUT FDD

This section will present the other two SR algorithms (the SRMMAE2 and the SRMMAE3). These two algorithms can both be applied to the output FDD.

The objective of this section is the FDD of output faults which are the ADS faults. First, the UMMAE and the AMMAE are applied to the ADS FDD. Then, the SRMMAE1 algorithm is applied which also shows flaw when identifying the output faults. Finally, the SRMMAE2 and the SRMMAE3 are proposed, which solved the problem concerning the estimation of the output faults.

In the simulation, consecutive ADS faults are generated. For 10 s $< t <$ 20 s, the true airspeed sensor V sensor fails with a bias with a value of 1 m/s. For 30 s $< t <$ 40 s, there is a drift fault in the angle of attack α sensor. The drift rate is 10^{-2} rad/s. For 50 s $< t <$ 60 s, a bias with the value of 10^{-2} rad occurs in the angle of sideslip β sensor.

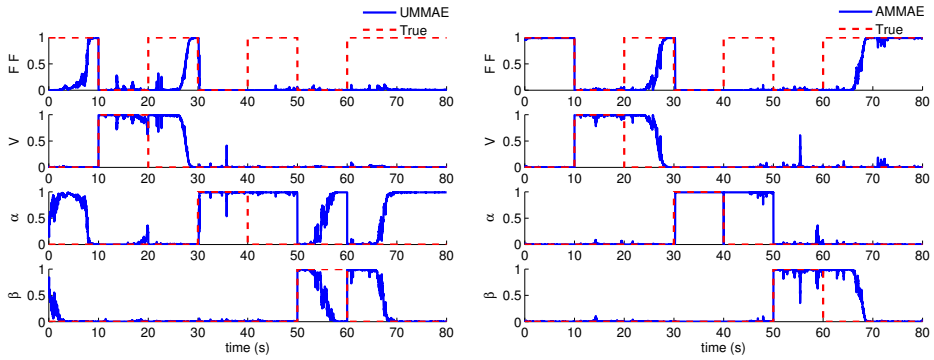
3.7.1. UMMAE & AMMAE FOR ADS FDD

The UMMAE and the AMMAE approach are applied to tackle the ADS FDD. Figure 3.7 shows the result of using the two approaches for the ADS FDD. From the figure, it is apparent that the AMMAE improves the performance of the UMMAE approach. The false alarms given by the α sensor fault filter at 0 s $< t <$ 8 s and 67 s $< t <$ 80 s (there are no faults in the system) disappear. The false alarms given by the α sensor fault filter during 53 s $< t <$ 60 s (The β sensor is faulty) also disappear. However, the inability to quickly detect the removal of the faults of the AMMAE shows again. It takes the AMMAE approximately 8 s, 10 s and 9 s to detect the removal of the V, α and β sensor fault, respectively.

3.7.2. SRMMAE1 FOR ADS FDD

In the previous section, the SRMMAE1 was proposed and solved the problems related to the input FDD problem. In this subsection, it is applied to detect, isolate and identify the ADS faults.

It can be seen from Figure 3.8(a) that the detection performance of the SRMMAE1 is better than that of the UMMAE and AMMAE approach. All the false alarms after the removal of the faults are eliminated successfully, which indicates the effectiveness of the



(a) Model probabilities of the four elemental filters using the UMMAE approach

(b) Model probabilities of the four elemental filters using the AMMAE approach

Figure 3.7: Results of the ADS FDD using the UMMAE and AMMAE approach

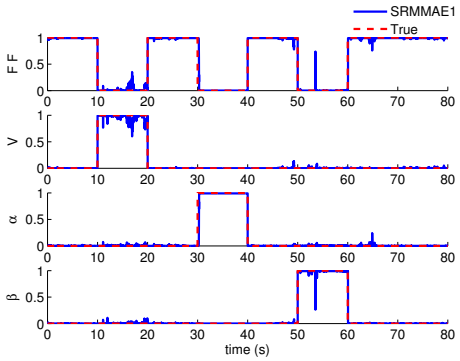
SRMMAE1 algorithm. The detection time of this SRMMAE1 algorithm is quick (less than 1 s). It takes only 0.02 s (2 time steps) and 0.03 s (3 time steps) to detect the fault of the V and the β sensor respectively. As for the fault of the α sensor, it takes longer (0.28 s) since the fault is a drift fault which is a slow-varying fault. The detection of the removal of the faults is also quick, which takes 0.02 s, 0.01 s and 0.02 s respectively. It is interesting to note that the removal of the drift fault is faster than that of the other two faults. This is because that at the end of the drift fault, its value is big due to the accumulation over time.

Figure 3.8(b) presents the information about the fault estimation. It can be seen that the estimation of the α and β sensor fault is acceptable while that of the V sensor fault is not. During 10 s $< t <$ 20 s, when the sensor of the V is faulty, the estimation error of the fault is not zero-mean although it converges to the true value in the end of the V sensor fault. Furthermore, during 40 s $< t <$ 50 s and 60 s $< t <$ 80 s, when there are no faults in the system, the fault estimation of the V sensor even diverges. The same problem is observed in the fault estimation of the β sensor during 20 s $< t <$ 30 s, when all the sensors operate normally. This will be explained in the next subsection.

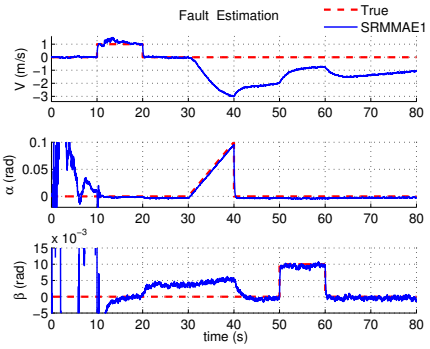
Figure 3.8(c) and Figure 3.8(d) demonstrate the state estimation performance of the SRMMAE1 algorithm compared to the normal UKF and the AMMAE approach. Note that the state estimation of the SRMMAE1 is computed based on the Equation (3.10), so the incorrect fault estimation of the V sensor is not reflected in the state estimation error. The SRMMAE1 exhibits a good state estimation performance even in the presence of the faults. However, there are some offsets in the error of the fault estimation such as the deviation between 30 s $< t <$ 40 s for the estimation of the α sensor fault.

3.7.3. SRMMAE2 FOR ADS FDD

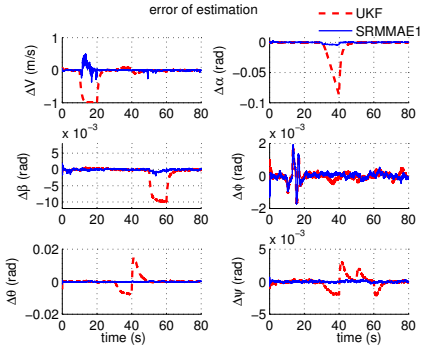
Let f_V, f_α and f_β denote the faults of the V, α and β sensor, respectively. As observed from Figure 3.8(b) in the last simulation, when $f_\alpha \neq 0$ (30 s $< t <$ 40 s), the fault estimation of f_V diverges and it can not converge to zero-mean even when there are no faults



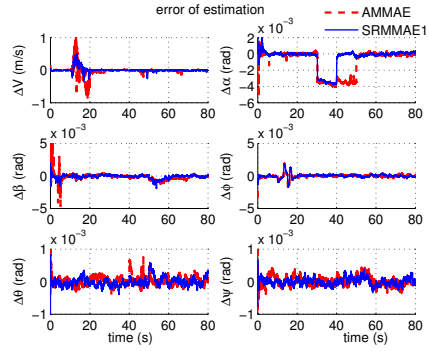
(a) Model probabilities of the four elemental filters



(b) ADS fault estimation of the three fault filters



(c) State estimation error of the SRMMAE1 compared to the UKF



(d) State estimation error of the SRMMAE1 compared to the AMMAE

Figure 3.8: Result of the ADS FDD using the SRMMAE1 approach and the state estimation error compared to the UKF and AMMAE approach

(40 s < t < 50 s and 60 s < t < 80 s). The estimation of f_β also cannot converge to zero-mean during 20 s < t < 30 s. The problem may be caused by the initialization of the filter. To find a solution to the problems using the SRMMAE1 algorithm, the following theorem is needed:

Theorem 1. *Consider the system described by Equation (3.1) and Equation (3.2), where* $\boldsymbol{f}^i = 0$, $\boldsymbol{f}^o \neq 0$. *An Augmented Unscented Kalman Filter (AUKF) is an UKF which augments the fault of the* i*th* ($i = 1, 2, ..., m$) *output* $f_i^o(k+1)$ *as an additional state. The UKF and AUKF are both applied for the state estimation.* $\hat{\boldsymbol{x}}(k+1|k)$ *is the a priori estimate of the states for the two filters,* $\hat{f}_i^o(k+1|k)$ *is the a priori estimate of* $f_i^o(k+1)$ *for the AUKF at time step* k, t_f *denotes the period when* $f^o \neq 0$ *and* t_{nf} *is the time interval when* $\boldsymbol{f}^o = 0$. *During* t_{nf}, *if the state estimate of the UKF is unbiased, the AUKF can achieve an unbiased estimate for both* $x_i(k+1|k)$ *and* $f_i^o(k+1|k)$ *through reinitializations using the UKF.*

Proof. Since $\boldsymbol{f}^i = 0$, although the ith measurement $y_i = \boldsymbol{h}(x_i(k+1)) + f_i^o(k+1) + v_i(k+1)$ is a wrong measurement for $\boldsymbol{h}(x_i(k+1))$, it is a correct measurement for $h(x_i(k+1)) + f_i^o(k+1)$. Therefore, for the AUKF, since the $f_i^o(k+1)$ is augmented as an additional state, the estimate $\boldsymbol{h}(\hat{x}_i(k+1|k)) + \hat{f}_i^o(k+1|k)$ is unbiased. But for the normal UKF, the estimate $h(\hat{x}_i(k+1|k))$ is biased.

During t_{nf}, unbiased state estimates of the UKF can be obtained using an approach such as the SRMMAE1. If the state estimate of the AUKF is replaced by the that of the UKF, then the expectation of ith state estimate of the AUKF

$$E\{\hat{x}_i(k+1|k)\} = x_i(k+1)$$

Since for the AUKF, $E\{\boldsymbol{h}(\hat{x}_i(k+1|k)) + \hat{f}_i^o(k+1|k)\} = \boldsymbol{h}(x_i(k+1)) + f_i^o(k+1)$. It follows that

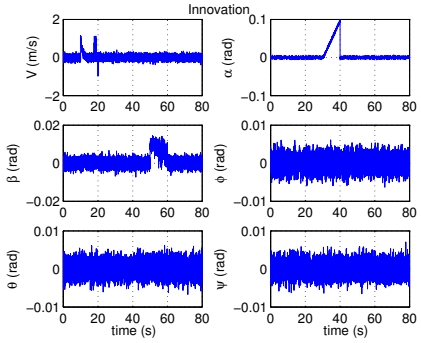
$$E\{\hat{f}_i^o(k+1|k)\} = f_i^o(k+1)$$

Therefore, both $\hat{x}_i(k+1|k)$ and $\hat{f}_i^o(k+1|k)$ of the AUKF are unbiased during t_{nf}. □

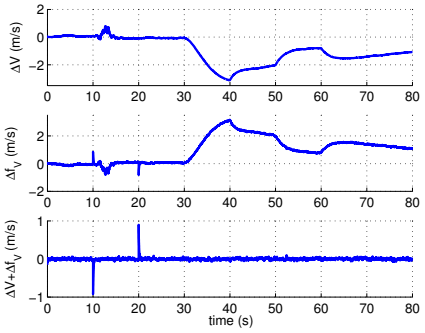
This theorem can be applied to check the performance of the filters. More importantly, it provides a way to achieve an unbiased estimate for the fault filters. The innovation of the fault free filter, using the SRMMAE1 approach for the ADS FDD, is shown in Figure 3.9(a). The fault free filter is reinitialized using the SRMMAE1 approach.

As can be seen from Figure 3.9(a), during the time when there are no faults (0 s < t < 10 s, 20 s < t < 30 s, 40 s < t < 50 s and 60 s < t < 80 s), the fault free filter achieves an unbiased estimate of V, α and β. This confirms that the SRMMAE1 algorithm is able to guarantee that the fault free filter achieves an unbiased estimation when there are no faults. From Figure 3.9(b), it can be seen that for the filter for the V sensor faults, although the estimate of the V and the fault f_V are biased, the estimate of $V + f_V$ is unbiased.

Therefore, according to Theorem 1, the fault free filter can be used to reinitialize the fault filters when there are no faults, which makes the estimate of V of the fault filter unbiased. Since the estimate of $V + f_V$ of the fault filter for the V sensor is unbiased, the estimate of V and f_V are both unbiased according to the theorem.



(a) Innovation of the fault free filter



(b) Estimation error of V, f_V and $V + f_V$ of the filter matching the V sensor faults

Figure 3.9: Results of the ADS FDD obtained using the SRMMAE1 approach

To reduce the number of reinitializations, the fault filters are reinitialized by the fault free filter only at the end of a fault free period or at the beginning of a fault. This means that the fault filter for the V sensor will be reinitialized by the fault free filter at $t = 30$ s and $t = 50$ s. The problem of the V sensor fault filter at $t = 40$ s and $t = 60$ s can be solved by reinitializations using the fault filter with the maximum model probability, which is based on the idea in the SRMMAE1 algorithm. The problem of the fault filter for the β sensor can be also solved in the same way.

Based on the above idea, more reinitializations are added to the SRMMAE1 algorithm. This leads to the SRMMAE2 algorithm given in Algorithm 2. In this algorithm, m is the dimension of the output fault described in Equation (3.2). $\hat{x}_i$ and P_i $(i = 1, 2, ..., m)$ are the a posteriori state estimates and the error covariance matrices of the m fault filters. The other variables have the same meanings as in Algorithm 1. This algorithm is applied to the ADS FDD problem.

Figure 3.10 shows the results of using the SRMMAE2 algorithm to detect, isolate and identify the ADS faults. The model probability and the filtered model probability of the four filters are given in Figure 3.10(a) and Figure 3.10(b), respectively. It can be seen that all the false alarms in Figure 3.7(a) and Figure 3.7(b) are eliminated. The detection time of this approach is the same as the result of the SRMMAE1 algorithm.

Figure 3.10(c) presents the fault estimation of each output using the SRMMAE2 algorithm. It can be seen from the figure that all the faults are estimated correctly during 10 s $< t <$ 20 s, 30 s $< t <$ 40 s and 50 s $< t <$ 60 s when the system is faulty. Furthermore, when there are no faults, all the fault estimation is zero-mean, different from the result of using the SRMMAE1 algorithm. This can be seen from Figure 3.10(d). The fault estimation using the SRMMAE1 algorithm diverges after a fault occurs at $t = 30$ s while that of using the SRMMAE2 algorithm maintains an unbiased estimation. The fault estimation problem with the β sensor during 20 s $< t <$ 30 s also disappears. The probability-weighted fault estimation of the algorithm using Equation (3.12) is given in Figure 3.10(e). The state estimation performance is shown in Figure 3.10(f) compared

Algorithm 2 SRMMAE2 Algorithm for Output FDD

At time step k;

1. **Variable assignment**

$p_{t,1} = p_{ff}$; $p_{t,i+1} = p_i$, $i = 1,2,...,m$

$\hat{x}_{t,1} \leftarrow \{\hat{x}_{ff,j} \mid j = 1 \text{ to } n\}$

$P_{t,1} \leftarrow \{P_{ff,j,j} \mid j = 1 \text{ to } n\}$

$\hat{x}_{t,i+1} = \{\hat{x}_{i,j} \mid j = 1 \text{ to } n+1\}$, $i = 1,2,...,m$

$P_{t,i+1} = \{P_{i,j,j} \mid j = 1 \text{ to } n+1\}$, $i = 1,2,...,m$

2. **Indexes of the filters with the maximal or minimal model probability**

Find $i_{min,k} \in \{ p_{t,i_{min,k}} = \min(p_{t,i}) \mid i = 1,2,...,m+1 \}$

Find $i_{max,k} \in \{ p_{t,i_{max,k}} = \max(p_{t,i}) \mid i = 1,2,...,m+1 \}$

3. **Reinitialization**

if $(k = 1)$ **or** $(k > 1$ **and** $i_{max,k-1} = 1$ **and** $i_{max,k} \neq 1)$ **then**

for i=1 to m **do**

$\hat{x}_{i,j} \leftarrow \hat{\boldsymbol{x}}_{ff}$, $j = 1$ to n

$P_{i,j,j} \leftarrow \mathbf{P}_{ff}$, $j = 1$ to n

end for

end if

4. **Additional reinitialization**

if $p_{ff} = p_{t,i_{min,k}}$ **then**

$\hat{x}_{ff} \leftarrow \hat{x}_{t,i_{max,k}}$

$P_{ff} \leftarrow P_{t,i_{max,k}}$

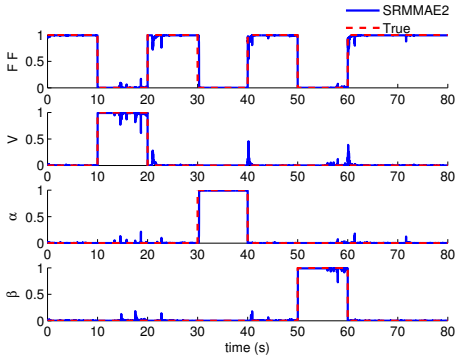
for i=1 to m **do**

$\hat{x}_{i,j} = \hat{\boldsymbol{x}}_{t,i_{max,k}}$, $j = 1$ to $n+1$

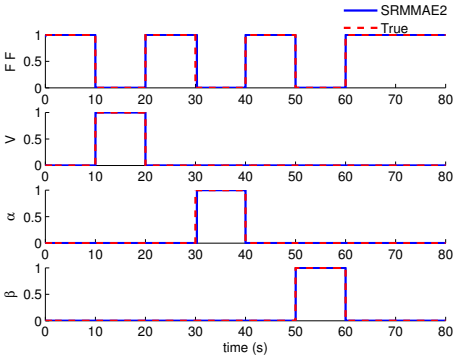
$P_{i,j,j} = \mathbf{P}_{t,i_{max,k}}$, $j = 1$ to $n+1$

end for

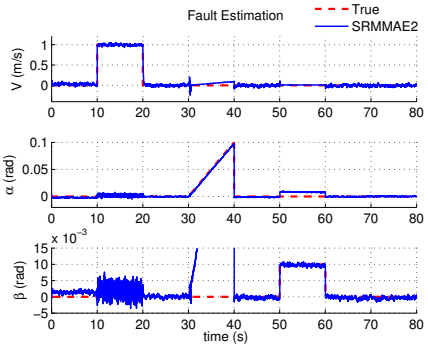
end if



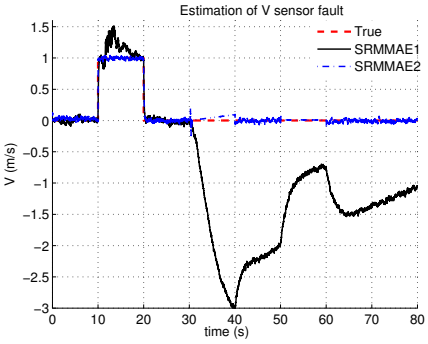
(a) Model probabilities of the four elemental filters using the SRMMAE2 approach



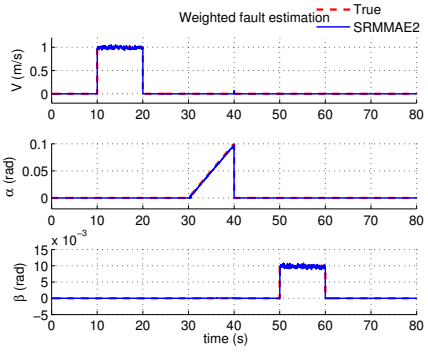
(b) Filtered model probabilities for the four filters using the SRMMAE2 approach



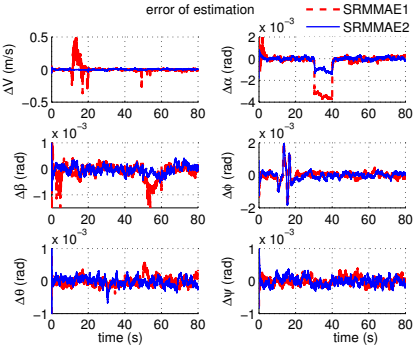
(c) ADS fault estimation of the three fault filters using the SRMMAE2 approach



(d) Fault estimation of the V sensor fault using the SRMMAE2 approach compared to the SRMMAE1

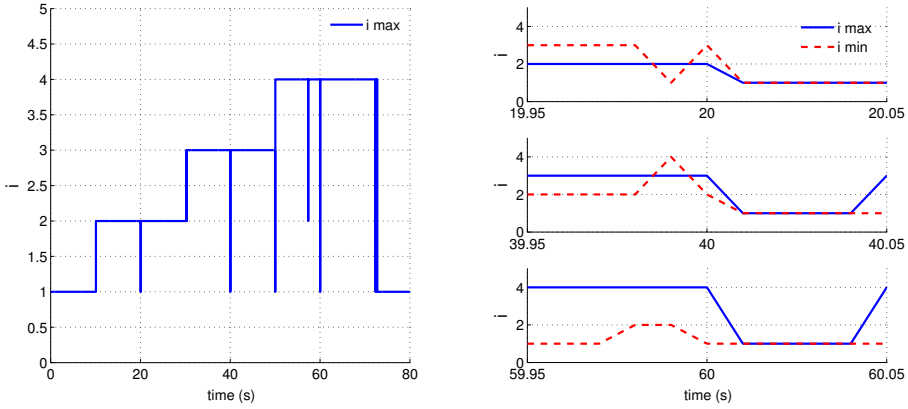


(e) Probability-weighted ADS fault estimation of the three fault filters using the SRMMAE2 approach



(f) State estimation error of the SRMMAE2 approach compared to the SRMMAE1

Figure 3.10: Result of the ADS FDD using the SRMMAE2 approach and the comparison with the SRMMAE1 approach



(a) Indexes of the maximum model probability using the AMMAE

(b) Indexes of the maximum and minimum model probability using the AMMAE, zoomed-in

Figure 3.11: Indexes of the maximum and minimum model probability among the four elemental filters using the AMMAE for the ADS FDD

to algorithm 1. It can be seen that SRMMAE2 algorithm reduces the estimation error of the V, α and β estimated by the SRMMAE1 algorithm during 10 s $< t <$ 20 s, 30 s $< t <$ 40 s and 50 s $< t <$ 60 s, respectively. It achieves zero-mean state estimation even in the presence of the output faults, which confirms the effectiveness of this algorithm.

3.7.4. SRMMAE3 FOR ADS FDD

In this subsection, the third SR algorithm is designed for the output FDD when the magnitude of the faults is not small. The definition of "small" is different according to different applications. In this chapter, we define the bias fault for the V sensor as small if its value is no bigger than 1 m/s. For the α and β sensor, the bias and drifting rate of the fault which is no bigger than 0.01 rad and 0.01 rad/s respectively is defined as small. In this simulation, the bias fault of the V sensor is 10 m/s, the drifting rate of the α sensor fault is 0.01 rad/s and the bias fault of the β sensor is 0.1 rad. Therefore, the faults are big compared to the previous simulations.

Let i denote the different filters. Specifically, $i = 1$ refers to the fault free model while $i = 2, 3, \ldots, m+1$ refer to the model matching the faults of the V, α and β sensor, respectively. Let i_{max}, i_{min} denote the indexes corresponding to the model with the maximum and minimum model probability respectively. When the AMMAE is applied to detect and identify the output faults, i_{max} usually displays a situation as shown in Figure 3.11(a). Figure 3.11(b) presents zoomed-in figures of i_{max} and i_{min} during 19.95 s $< t <$ 20.05 s, 39.95 s $< t <$ 40.05 s and 59.95 s $< t <$ 60.05 s respectively.

As Figure 3.11(b) shows, both i_{max} and i_{min} drop to one once the faults are removed at 20, 40 and 60 s. This occurs when the model probabilities of all the models are equal to the minimum probability limited by the MMAE. This indicates that the faults have

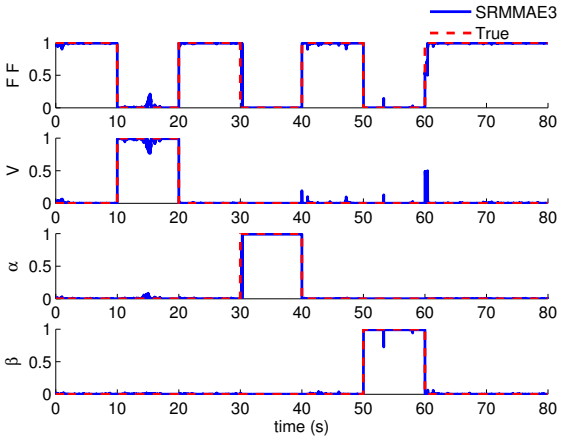


Figure 3.12: Model probabilities of the four filters using the SRMMAE3 algorithm for the ADS FDD

been removed. Therefore, we only need to reinitialize the fault free filter at this time step using the filter with the maximum model probability in the last time step. By doing this, the required reinitializations are less than those required by the SRMMAE2. The third SR algorithm is given in Algorithm 3.

Algorithm 3 SRMMAE3 for output FDD

At time step k;

1. **Variable assignment**
   Same as in Algorithm 2
2. **Indexes of the filters with the maximal or minimal model probability**
   Same as in Algorithm 2
3. **Reinitialization**
   Same as in Algorithm 2
4. **Additional reinitialization**
   if $k > 1$ **and** $i_{max,k} = i_{min,k}$ **and** $i_{max,k-1} \neq i_{min,k-1}$ **and** $i_{max,k-1} \neq 1$ **then**
   $\hat{\boldsymbol{x}}_{ff} \leftarrow \hat{x}_{t,i_{max,k-1}}$
   $\mathbf{P}_{ff} \leftarrow P_{t,i_{max,k-1}}$
   for i=1 to m **do**
   $\hat{x}_{i,j} = \hat{x}_{t,i_{max,k-1}}, \quad j = 1 \text{ to } n+1$
   $P_{i,j,j} = P_{t,i_{max,k-1}}, \quad j = 1 \text{ to } n+1$
   end for
   end if

The performance of Algorithm 3 is shown in Figure 3.12. As can be seen, the performance is comparable to that of the SRMMAE2.

Remarks: The SRMMAE2 and SRMMAE3 are two approaches which can both be applied to output FDD. The only difference between them is in the fourth step of the algorithm as can be seen from Algorithm 3. The fourth step of SRMMAE3 requires n_f (the

number of output faults) reinitializations which are much less than those required by the fourth step of SRMMAE2. However, recall that the faults in this simulation are bigger than the previous ones. For small faults, SRMMAE3 may not be effective. If the subtle fault has a little influence on the innovation, the sign to trigger the SR may not appear.

3.8. COMPARISON WITH THE IMM APPROACH

3.8.1. ADS AND IMU FDD USING THE IMM APPROACH

In this section, the proposed SRMMAE approach will be compared with the IMM approach [35, 99, 105]. The IMM approach is used to detect and isolate the ADS and IMU fault respectively. The situation is the same as in the simulations in Section 3.6 and Section 3.7.

Figure 3.13 shows the result of the ADS FDD using the IMM approach. As can be seen from Figure 3.13(a), the fault detection performance is good. The detection time of three sensor faults is 0.02 s, 0.57 s and 0.02 s respectively, which is comparable with that using the SRMMAE approach. Furthermore, there are no false alarms, which means there is no need to filter the model probability. Figure 3.13(b) shows the error of the state estimation by the IMM approach. The error is close to zero-mean even in the presence of the faults which indicates the state estimation performance of the IMM approach.

Figure 3.13(c) presents the model probabilities when the IMM approach is used to detect and isolate the IMU faults. The performance is bad. There are mainly three problems:

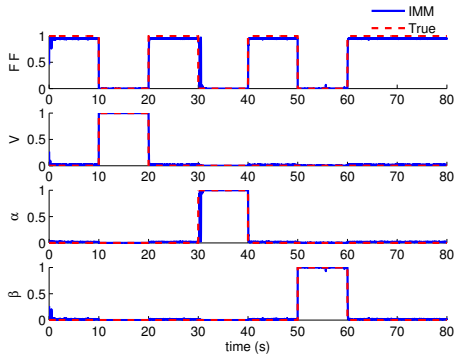
1. There are many false alarms. The model for the A_y fault continues to give false alarms when there are no faults in this sensor.

2. The model probability cannot reach one when the corresponding model is the true model. Whether for the fault free model or the fault models, the model probability is less than one when the model is the true model.

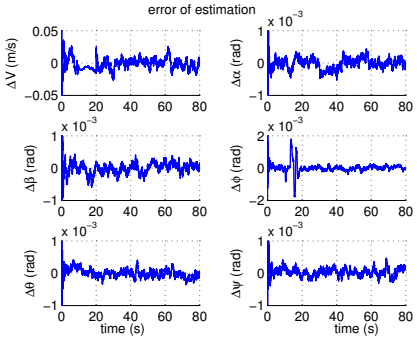
3. The performance of detecting the drift faults (the fault occurs in the accelerometer A_y, the gyro q and r) is even worse. The detection time of the pitch rate sensor is around 5 s and the model probability does not stay at one. The detection of the yaw rate sensor fault is missed.

Similar problems have been found by others who are working on input FDD using the IMM approach [104]. The IMM approach performs poorly when dealing with the IMU sensor faults for several reasons:

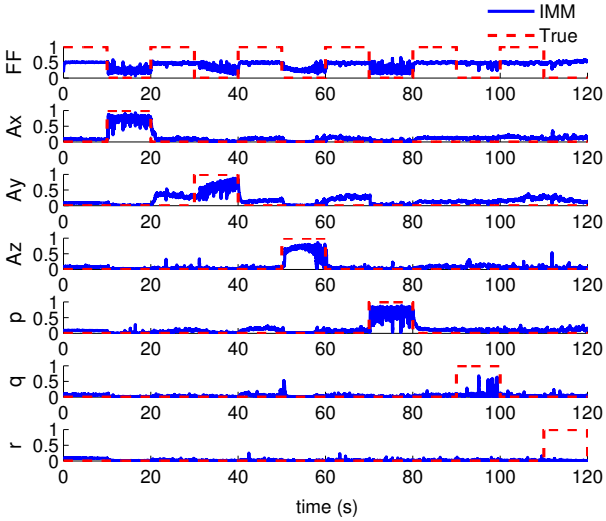
1. Since the ADS faults are the output faults $\boldsymbol{f}^o(k+1)$, the information of the ADS fault at time step $k+1$ can be reflected directly in the innovation $\boldsymbol{\gamma}(k+1)$. In this chapter, the IMU sensor faults are the input faults $\boldsymbol{f}^i(k)$, so the faulty information of the IMU at time step k will be integrated before it has an influence on the innovation in the next step $\boldsymbol{\gamma}(k+1)$. This can be derived from the system model described by Equation (3.13) and Equation (3.14). If there are input and output faults in the



(a) Model probabilities of the four elemental filters using the IMM for ADS FDD



(b) State estimation error using the IMM for ADS FDD



(c) Model probabilities of the seven elemental filters using the IMM for the IMU sensor FDD

Figure 3.13: IMU sensor and ADS FDD result using the IMM approach

system, the innovation is:

$$\begin{aligned}\boldsymbol{\gamma}(k+1) &= \boldsymbol{y}(k+1) - \boldsymbol{h}(\hat{\boldsymbol{x}}(k+1)) \\ &= \boldsymbol{h}\Big[\boldsymbol{x}(k) + \int_k^{k+1} \big(\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{f}^i(t)\big)dt\Big] \\ &+ \boldsymbol{v}(k+1) + \boldsymbol{f}^o(k+1) - \boldsymbol{h}(\hat{\boldsymbol{x}}(k+1))\end{aligned}$$

This property makes input faults more difficult to detect than output faults.

2. The magnitude of the IMU sensor faults are much smaller than those of the ADS faults. This makes the IMU sensor faults more difficult to detect.

3. The most crucial reason is that the IMM approach has a tendency to enhance the state estimation ability which weakens its performance of fault detection. All the models in the IMM filter interact with each other at every step. The states of all the models mix with each other which makes the differences among them smaller step by step. Hence, the probabilities of these models will also approach each other. This becomes more obvious when the fault of the system is small. In this case, the innovation of the elemental filters will become more similar to each other, leading to similar model probabilities which causes the missed detection of slow-varying faults (such as the drift fault).

3.8.2. COMPARISON BETWEEN THE IMM AND THE SRMMAE APPROACH

This subsection will compare the IMM with the proposed SRMMAE approach and present the advantages and disadvantages of these two approaches.

The similarity between these two approaches is that the reinitialization they perform both lead to interaction within the elemental filters. The main differences of the reinitialization as well as the performance comparison are summarized as follows:

1. The IMM approach reinitializes all the filters with its mixed state estimates. This mixing process of the IMM approach weakens its ability to detect the fault despite of the enhancement of its state estimation performance. In contrast, the SRMMAE approach includes three SR algorithms which mainly use the states of the fault filters to reinitialize the fault free filters or use the states of the fault free filter to reinitialize the fault filters. Since it does not use the mixed states and does not reinitialize all the filters, it is still able to maintain the ability to detect and identify small input faults while the IMM is unable to detect small input faults.

2. The IMM approach reinitializes each filter at every time step while the SRMMAE approach only reinitializes for certain time steps according to the different algorithms. In the fourth step of the SRMMAE3 algorithm, only 3 steps of reinitialization are needed. Therefore, the computational load of the IMM is higher than the SRMMAE approach.

To sum up, the SRMMAE approach reinitializes the filter selectively using three SR algorithms to maintain the state estimation performance and enhance its fault detection ability while the IMM uses the mixed state estimates to reinitialize every filter at every

step which enhances its state estimation ability whereas weakens its fault detection ability.

3.9. CONCLUSIONS

This chapter is concerned with improving the FDD performance of the multiple-model based approach. There are two popular multiple-model adaptive estimation algorithms: the MMAE and the IMM. The MMAE approach is able to detect the faults quickly. However, there are some problems with using this approach. This chapter proposed a novel SRMMAE approach which improves the performance of the MMAE significantly. First, a state augmentation strategy is introduced to both input and output FDD. Then, three SR algorithms are designed in order to cope with input and output FDD. The proposed approach is tested with the example of the IMU and ADS FDD of a Cessna Citation II aircraft model, which demonstrates its performance in both state estimation and FDD.

The SRMMAE approach is compared to the IMM approach both in effectiveness and efficiency. As for output FDD, the performance is comparable to each other whereas in terms of input FDD, the SRMMAE approach outperforms the IMM.

After the diagnosis of the sensor faults using the SRMMAE approach, the correct sensor measurements are fed back to the flight control computer to generate control laws. Therefore, the proposed approach is able to enhance the safety of the aircraft in the presence of sensor faults. It can be extended to the FTC system, which is the future plan. Besides, how to maintain the fault detection performance of the SRMMAE approach using fewer reinitialization steps is also an interesting topic.

In conclusion, the proposed SRMMAE approach is a good candidate for input and output FDD based on its excellent performance on both state estimation and fault diagnosis. Therefore, it can enhance the reliability and safety of the aircraft.

4

DOUBLE-MODEL ADAPTIVE FAULT DETECTION AND DIAGNOSIS APPLIED TO REAL FLIGHT DATA

In the previous chapter, the ADS FDD was solved using the Selective-Reinitialization Multiple-Model Adaptive Estimation approach which provides fast fault detection. However, this approach is computationally intensive since it runs a number of models online. When dealing with simultaneous faults, the number of models increase and lead to more computational load. In this chapter, a novel Double-Model Adaptive Estimation approach is proposed to solve the problem. First the novel approach is introduced in detail. Then, it is applied to the ADS FDD and is compared to existing approaches such as the Selective-Reinitialization Multiple-Model Adaptive Estimation approach.

The flight data used in this chapter is available at: https://www.researchgate.net/profile/Peng_Lu15/publications?pubType=dataset.

Parts of this chapter are based on:

P Lu, L Van Eykeren, E van Kampen, C C de Visser, Q P Chu. Double-model Adaptive Fault Detection and Diagnosis Applied to Real Flight Data. *Control Engineering Practice*, (36), 39-57, 2015. [92].

The existing multiple model-based estimation algorithms for FDD require the design of a model set, which contains a number of models matching different fault scenarios. To cope with partial faults or simultaneous faults, the model set can be even larger. A large model set makes the computational load intensive and can lead to performance deterioration of the algorithms. In this chapter, a novel DMAE approach for output FDD is proposed, which reduces the number of models to only two, even for the FDD of partial and simultaneous output faults. Two SR algorithms are proposed which can both guarantee the FDD performance of the DMAE. The performance is tested using a simulated aircraft model with the objective of ADS FDD. Another contribution is that the ADS FDD using real flight data is addressed. Issues related to the FDD using real flight test data are identified. The proposed approaches are validated using real flight data of the Cessna Citation II aircraft, which verified their effectiveness in practice.

4

4.1. INTRODUCTION

Presently, FDD is important to achieve fault-tolerance [11]. For flight control systems, sensor or actuator faults may cause serious problems. Thus, quick detection and isolation of these faults is highly desirable [9]. During the last few decades, many approaches have been proposed for aircraft actuator and sensor FDI[12, 13, 66]. Some recent advances and trends can be found in Zolghadri [14] and Goupil [9]. One recent European project, Advanced Fault Diagnosis for Sustainable Flight Guidance and Control (ADDSAFE), aims to develop FDI methods for aircraft flight control systems [112]. Within this project, a number of model-based FDI methods were tested and evaluated, refer to Varga and Ossmann [113], Van Eykeren and Chu [114], Henry and Cieslak et al. [115], Alwi and Edwards [116], Chen and Patton et al. [117], Vanek and Edelmayer et al. [118], Hecker and Pfifer [119] and Marcos [23]. However, few of these papers [114] consider the FDD of the ADS. The ADS measure the air data information which is critical to the pilot and to the flight control system. They are usually mounted to the outside of the fuselage. Therefore, they can be affected by the environment in which the aircraft is flying. Faults of the ADS are contributing factors which have led to several aircraft accidents. For civil aircraft, the final report of the Air France Flight 447 accident stated that erroneous airspeed measurements from the pitot probes were a contributing factor [73]. An example for military aircraft is the cause of the crash of a B-2 Bomber; it was found that moisture in the port transducer units caused a large bias to the ADS [73]. These are only two examples of recent air disasters caused by failures of the ADS system. Therefore, the FDD of the ADS is important. Recently, Freeman et al. [25] models the faults of the ADS using the physical air data relationships and experimental wind tunnel data. The present chapter deals with the detection and diagnosis of the ADS faults.

One of the most effective approaches for the FDD is the multiple-model-based approach [99]. The basic idea of performing FDD using the multiple model (MM) approach is: a model set must be created that contains models corresponding to different fault conditions of the monitored system. In addition to fault models, the model set usually includes the nominal model. There are two principal stochastic architectures based on the MM approach [13]: the MMAE and the IMM. Both approaches require the design of a model set which represents or covers all possible system models at any time [99].

The MMAE [34, 37, 93, 101] algorithm runs a bank of filters in parallel, termed “ele-

mental filters". Each filter is based on a model matching a particular fault mode of the system. If the set of models used by the MM approach does not change, it is referred to as a fixed model set. When coping with partial faults or simultaneous faults, one disadvantage of the MM approach based on a fixed model set is that the number of models needed to cover all expected failures can be large, making implementation of a single MM estimator impractical [104] for real time FDD. In order to cope with this situation, Maybeck and his research team proposed a hierarchical structure [37] and a moving-bank MMAE algorithm [36, 106] to reduce the number of required filters online. The hierarchical MMAE is designed for the FDD of single- and dual-failure hypotheses. Although this approach runs $K+1$ (K is the number for single fault FDD) models online, $K(K+1)+1$ models have to be designed. When it is used for FDD of dual-failure scenarios, it is assumed that the second fault occurs two or more seconds later which allows enough time for the first fault to be detected before the second is inserted [93]. In addition, it can not cope with three or more simultaneous faults. The moving-bank MMAE is also designed to avoid the potentially large number of element filters needed for an MMAE bank. It uses less filters to identify the parameters related to the K basic element models. However, more than $K+1$ models are still required only for single FDD. Ducard et al. [103] proposed to augment the faults of the input as additional states, which reduces the number of the models to only $K+1$. More recently, Lu et al. [107] proposed to use SR algorithms to solve the problem when the output faults are augmented to reduce the size of the model set.

The IMM [35, 99, 105] is another type of multiple-model-based approach. Its difference from the MMAE lies in the fact that element filters in the IMM interact with each other, which leads to a better state estimation performance. Both the MMAE and the IMM display deteriorated performance in case where the model set does not contain a model corresponding to the true system [120]. The model set can become very large when dealing with multiple faults. Li et al. [105] proposed a variable structure IMM with an adaptive model set. The expected-mode augmentation [121] is used to adaptively change the model set. However, to cope with partial faults, an extra feature needs to be added to the IMM. Ru et al. [104] used a maximum likelihood estimator to estimate the extent of the faults after the detection of a fault using the IMM. In addition, to cope with two or more simultaneous faults, the number of expected modes will increase which makes the model set larger.

The drawbacks of a large model set are as follows: firstly, it brings a high computational load which increases with the number of models. This is also the reason why many approaches were proposed to reduce the model set; secondly, a large model set could lead to models being similar to one another in terms of input-output behavior, which could lead to performance deterioration of the MM approaches [99, 120].

In this chapter, a novel approach called DMAE is proposed for the output FDD. This approach reduces the number of the models in the model set to only two, regardless of K and whether there are partial faults or simultaneous output faults. The two models used in this approach are the no-fault model and the fault model. The states of the fault model are augmented by the K output fault scenarios rather than one fault scenario. Augmenting the output faults as states may lead to reconstructibility problems; this is solved by using the SR scheme. Two SR algorithms are proposed to guarantee the per-

formance of the DMAE approach. The elemental filters are designed based on an UKF [27], which uses the direct nonlinear model without linearization and can achieve higher order accuracy. The performance of the two proposed double-model-based approaches are compared to the multiple-model-based approach as well as to each other. The example is the FDD of the ADS faults of a Cessna Citation II CE-500 aircraft model. In order to demonstrate the performance, different fault scenarios are considered, including a single fault and simultaneous faults, small faults and big faults, bias faults and drift faults.

A second contribution of this chapter is the application of the proposed approach to the ADS FDD using real flight test data. It should be noted that few papers [97] dealing with aircraft sensor FDD use real flight data. Berdjag et al. [97] used real flight data to detect the faults in the inertial data. But they did not use it to detect the ADS faults. To the best of the authors' knowledge, the present chapter is one of the first few studies concerning ADS FDD use real flight test data.

Real flight data are more difficult to deal with mainly due to the model uncertainties such as the model inaccuracies and unmodelled dynamics, measurement biases and disturbances. The important issues that arise when dealing with real flight data are identified and solutions are provided. Calibrations are performed in order to reduce the influence of model uncertainties. The proposed approaches are validated using real flight test data of the Cessna Citation II aircraft. The IMU sensors and ADS of the aircraft contains biases but are not calibrated. The command for pilots was frequency sweep input, which was not designed for sensor calibration but for aerodynamic model identification purposes. Under these conditions, the proposed approaches are still able to successfully detect and reconstruct the ADS faults that are injected into this real flight data, which verifies the effectiveness of the approach. Therefore, the proposed approaches can potentially be applied to the real practice.

The structure of this chapter is as follows: Section 4.2 gives the preliminaries. Section 4.3 introduces the DMAE approach. The performance of the proposed approach is demonstrated using simulated data in Section 4.4. In Section 4.5, the issues related to the FDD with application to real flight data are dealt with and the performance of the approach is tested using real flight data. Section 4.6 concludes the chapter.

4.2. PRELIMINARIES

This section introduces the preliminaries which are the basis of the approaches used in this chapter.

4.2.1. NONLINEAR SYSTEM MODEL WITH OUTPUT FAULTS

Consider the following nonlinear state-space model with output faults

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \mathbf{G}(t)\boldsymbol{w}(t) & (4.1) \\ \boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \boldsymbol{v}(t) + \boldsymbol{f}^o(t), \qquad t = t_i, \qquad i = 1, 2, \ldots & (4.2) \end{cases}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the system states, $\boldsymbol{u} \in \mathbb{R}^b$ the input, $\boldsymbol{y} \in \mathbb{R}^l$ the measurement. The matrix $\mathbf{G}$ is the noise distribution matrix. The function $\bar{\boldsymbol{f}}$ and $\boldsymbol{h}$ are nonlinear functions. The function $\boldsymbol{f}^o$ represents output faults. $\boldsymbol{w}$ and $\boldsymbol{v}$ represent input and output noises.

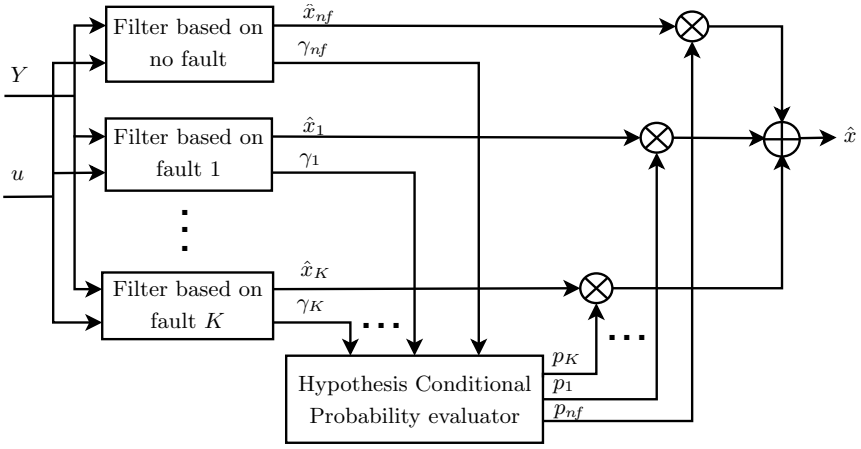


Figure 4.1: Block diagram of the MMAE approach

4.2.2. MULTIPLE-MODEL ADAPTIVE ESTIMATION

The proposed approach in this chapter is mainly based on the framework of the MMAE approach. Therefore, the background of the MMAE approach is given.

The MMAE [37] is composed of a bank of KFs operating in parallel. All the filters use a vector of measurements $\boldsymbol{Y}$ explained below and a vector of input $\boldsymbol{u}$, and are based on the same equations of motion, while each hypothesizes a different fault scenario. At sample step k, each of the filters produces a state estimate $\hat{\boldsymbol{x}}(k)$ and a vector of innovation $\boldsymbol{\gamma}(k)$. The basic idea is that the KF which produces the most well-behaved innovation contains the model which matches the true faulty model best [93]. The block diagram for the MMAE is given in Figure 4.1. The "nf" in the figure means no fault.

A hypothesis test uses the innovation $\boldsymbol{\gamma}(k)$ and the residual covariance matrix $\mathbf{A}(k)$ of the filters in order to assign a conditional probability to each of the filters which matches a particular fault scenario. Let a denote the fault scenarios of the nonlinear system and K the number of the filters of the MMAE. If we define the hypothesis conditional probability $p_i(k)$ as the probability that a is assigned a_i for $i = 1,2,...,K$, conditioned on the measurement history up to time step k:

$$p_i(k) = \Pr[a = a_i | \boldsymbol{Y}(k) = \boldsymbol{Y}_k], \quad i = 1,2,...,K \tag{4.3}$$

The conditional probability will be updated recursively using the following equation:

$$p_i(k) = \frac{f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i,\boldsymbol{Y}_{k-1})\,p_i(k-1)}{\sum_{j=1}^{K} f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_j,\boldsymbol{Y}_{k-1})\,p_j(k-1)}, \quad i = 1,2,...,K \tag{4.4}$$

where $\boldsymbol{Y}_{k-1}$ is defined as $\boldsymbol{Y}_{k-1} = \{\boldsymbol{y}(0), \boldsymbol{y}(1), .., \boldsymbol{y}(k-1)\}$. $f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i,\boldsymbol{Y}_{k-1})$ is the probability density function which is defined as follows:

$$f_{\boldsymbol{y}(k)|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}(k)|a_i,\boldsymbol{Y}_{k-1}) = \beta_i(k)\,\exp\{-\boldsymbol{\gamma}^T(k)\boldsymbol{A}_i^{-1}(k)\boldsymbol{\gamma}(k)/2\} \tag{4.5}$$

where

$$\beta_i(k) = \frac{1}{(2\pi)^{l/2}|\boldsymbol{A}_i(k)|^{1/2}} \tag{4.6}$$

In Equation (4.6), l is the dimension of the measurement, the symbol "|...|" denotes the determinant of the covariance matrix $\boldsymbol{A}_i(k)$ which is computed by the KF at time step k. The filter which matches the fault scenario produces the smallest innovation which is the difference between the estimated measurement and the true measurement. Therefore, the conditional probability of the filter which matches the true fault scenario is the highest among all the filters. After the computation of the conditional probability, the state estimation of the nonlinear system $\hat{\boldsymbol{x}}(k)$ can be generated by the weighted state estimate $\hat{\boldsymbol{x}}_i(k)$ of the K filters:

$$\hat{\boldsymbol{x}}(k) = \sum_{i=1}^{K} \hat{\boldsymbol{x}}_i(k)\, p_i(k). \tag{4.7}$$

4.2.3. MMAE FOR FDD

The model probability indicates to what extent each model matches the system model; hence, it can be used for the FDD in real time. Normally, a fault alarm is triggered if the associated fault model probability exceeds a certain percentage described as follows:

$$p_i(k) \begin{cases} > p_m \Rightarrow \text{trigger the fault alarm for the } i\text{th model} \\ < p_m \Rightarrow \text{No } i\text{th fault} \end{cases} \tag{4.8}$$

where the threshold p_m is 0.9 in this research. Fisher et al. [122] and Ducard et al. [103] suggest a fault is detected if the conditional probability of the elemental filter exceeds 0.9 for a certain amount of time while Maybeck [37] claims a fault is detected if the conditional probability is above 0.9 for 10 sample periods or exceeds 0.95. A lower threshold would detect the faults faster but may also increase the false alarm rate [122]. In addition, in order to avoid the possibility that recursive computation of the conditional probability (Equation (4.4)) stays at zero as soon as the conditional probability reaches zero, the conditional probability $p_i(k)$ is bounded by a lower bound of 0.001 [37, 93, 103, 122].

If the fault is estimated by the K filters, then the probability-weighted fault estimation of the ith filter is:

$$\bar{f}_i(k) = \hat{f}_i(k)\, p_i(k),\ i = 1, 2, ..., K \tag{4.9}$$

where $\hat{f}_i(k)$ is the fault estimation of the ith filter and $\bar{f}_i(k)$ is the probability-weighted fault estimation of the ith filter.

4.3. THE DOUBLE-MODEL ADAPTIVE ESTIMATION APPROACH

This section presents the DMAE approach. As mentioned, the DMAE approach uses a framework similar to that of the MMAE. This section will first introduce the differences with the MMAE. Then two new SR algorithms will be proposed which are essential to the DMAE approach.

4.3.1. DIFFERENCES FROM THE MMAE FRAMEWORK

The biggest difference between the proposed DMAE and the MMAE is that the DMAE reduces the number of models in the model set to two: the no-fault model and the augmented fault model. Therefore, $K = 2$ for all the K in Section 4.2. The MMAE and the IMM approach requires at least $K+1$ models to cope with multiple faults and more models to deal with simultaneous faults. The MMAE approach requires $3K+1$ models to identify the partial faults. This is the biggest difference which is also the main advantage of the proposed approach compared to other multiple-model-based approaches.

Let subscript "nf" denote the quantities related to the no-fault model and "af" denote those related to the augmented fault model. It is assumed that there are m ($m \leq l$) output faults without losing generality. As for the MMAE approaches in Ducard et al. [103] and Lu et al. [107], considering single FDD, the states of the no-fault filter are:

$$\boldsymbol{x}_{nf}(k) = [x_1(k),\ x_2(k),\ \dots,\ x_n(k)]^T \tag{4.10}$$

The states of the ith fault filter are:

$$\boldsymbol{x}_{af_i}(k) = [x_1(k),\ x_2(k),\ \dots,\ x_n(k),\ f_i^o(k)]^T,\ i = 1,2,\dots,m$$

where $f_i^o(k)$ is the fault vector of the ith fault filter.

As for the DMAE approach, the states of the no-fault filter are the same as those of the no-fault filter using MMAE approaches (Equation (4.10)). However, the fault filter of the DMAE approach augments all the output faults as additional states instead of only one fault as implemented by the MMAE approach. Then, the state vector of the fault filter matching the fault model of the DMAE approach is as follows:

$$\boldsymbol{x}_{af}(k) = [x_1(k),\ x_2(k),\ \dots,\ x_n(k),\ f_1^o(k),\ f_2^o(k),\ \dots,\ f_m^o(k)]^T \tag{4.11}$$

The DMAE approach is able to detect and isolate the faults in the system more efficiently than the traditional MMAE approach. This can be explained by the computational complexity analysis of the DMAE approach in the following subsection.

Another advantage of the DMAE approach compared to the MMAE approach is that it reduces the false alarm rate. A large model set could lead to similar elemental filters, which results in performance degradation [99, 105, 120] such as false alarms. A double-model set could reduce performance degradation, which will also be shown in the simulation section.

4.3.2. COMPUTATIONAL COMPLEXITY COMPARISON

This subsection compares the computational complexity of the MMAE and DMAE approach, assuming there are multiple faults which do not occur simultaneously. The UKF is used for analysis.

As for the MMAE, the most intensive computation complexity of the no-fault filter is $\mathcal{O}(n^3)$ while that of the fault filter is $\mathcal{O}(m(n+1)^3)$. In terms of DMAE approach, the most intensive complexity of the no-fault filter is $\mathcal{O}(n^3)$ whereas that of the fault filter is $\mathcal{O}((n+m)^3)$. It is straightforward to compare the complexity of the fault filters of these two approaches. It can be verified that as long as $n > m > 2$ or $n \geq m > 4$, the following

inequality holds:

$$(n+m)^3 < m(n+1)^3 \tag{4.12}$$

In the simulation section, $n = 6$, $m = 3$, which satisfies the above inequality. Therefore, the computation complexity is reduced. Note that if $m > n$, the complexity needs to be further analyzed.

4.3.3. SELECTIVE REINITIALIZATION ALGORITHMS

The DMAE approach uses a modified framework of the multiple-model-based approach. Reinitialization is important to such an approach [99]. Lu et al. [107] proposed a SRMMAE approach which consists of a SR scheme different from the reinitialization scheme of the traditional MMAE and IMM approach. The SRMMAE, which is an improved version of the MMAE, is more effective than the IMM when coping with small input FDD. In this chapter, two SR algorithms for the DMAE are proposed.

The objective of the two SR algorithms is to achieve an unbiased estimation for both the no-fault filter and the fault filter in the absence and presence of output faults through certain reinitializations. The proposed SR algorithms have the same structure:

1. Determine whether there are output faults in the system.
2. If there are no output faults, perform reinitialization for the fault filter using the state estimate of the no-fault filter.
3. If there are output faults, keep performing reinitialization for the no-fault filter using the state estimate of the fault filter.

The details of the first SR algorithm (SR1) are given in Algorithm 4. This algorithm keeps reinitializing one filter using the other filter. However, it is still different from the reinitialization of the IMM approach since SR1 does not reinitialize all the filters whereas the IMM reinitializes all the filters simultaneously.

The details of the second SR algorithm (SR2) are given in Algorithm. 5. This SR2 is an extension of the SR used for output FDD in Lu et al. [107].

In the algorithms, $\hat{\boldsymbol{x}}^0_{nf}$ ($\hat{\boldsymbol{x}}^0_{af}$) and $\hat{\boldsymbol{x}}_{nf}$ ($\hat{\boldsymbol{x}}_{af}$) denote the state estimate of the no-fault (augmented fault) filter before and after the reinitialization, respectively. $\mathbf{P}^0_{nf}$ ($\mathbf{P}^0_{af}$) and $\mathbf{P}_{nf}$ ($\mathbf{P}_{af}$) denote the covariance of state estimate error of the no-fault (augmented fault) filter before and after the reinitialization, respectively. $\hat{\boldsymbol{x}}_t$, $\boldsymbol{p}_t$ and $\mathbf{P}_t$ are the vectors which contain the state estimate, model probability and the covariance matrix of state estimation error of the no-fault filter and the fault filter respectively. $i_{max,k}$ is the index of the model with the maximum model probability at time step k. $\boldsymbol{x}_0$ and $\mathbf{P}_0$ are parameters which are used for the initialization of the filter.

The first two steps of the two algorithms are the same. The objective of these two steps is to assign the two models in a set $\{\boldsymbol{p}_t,\ \boldsymbol{x}_t,\ \mathbf{P}_t\}$ and find the index of the model with the maximum probability $p_{t,i_{max,k}}$ in the model set. It should be noted that the dimension of the state of the no-fault filter is n while that of the fault filter is $n+m$.

The main similarity between the two algorithms is that they both keep reinitializing the no-fault filter using the fault filter in the presence of output faults (the 4th step). The reason is explained in the following paragraph.

Algorithm 4 SR1 for Output FDD

At time step k;

1. **Variables assignment**

$$p_{t,1} = p_{nf}\,; \quad p_{t,2} = p_{af}$$
$$\hat{x}_{t,1} = \hat{x}^0_{nf,j}\,; \quad \hat{x}_{t,2} = \hat{x}^0_{af,j}\,, \quad j = 1 \text{ to } n$$
$$P_{t,1} = P^0_{nf,jj}\,; \quad P_{t,2} = P^0_{af,jj}\,, \quad j = 1 \text{ to } n$$

2. **Index of the filter with the maximal model probability**

Find $i_{max,k} \in \{\, p_{t,i_{max,k}} = \max(\, p_{t,i}) \mid i = 1,2 \,\}$

3. **Reinitialization for the fault filter**

if $i_{max,k} = 1$ **then**

$$\hat{x}_{af,j} = \hat{x}_{t,i_{max,k}}, \quad j = 1 \text{ to } n$$
$$P_{af,jj} = P_{t,i_{max,k}}, \quad j = 1 \text{ to } n$$
$$\hat{x}_{af,j} = \boldsymbol{x}_0, \quad j = n+1 \text{ to } n+m$$
$$P_{af,jj} = \mathbf{P}_0, \quad j = n+1 \text{ to } n+m$$

end if

4. **Reinitialization for the no-fault filter**

if $i_{max,k} = 2$ **then**

$$\hat{\boldsymbol{x}}_{nf} = \hat{x}_{t,i_{max,k}}$$
$$\mathbf{P}_{nf} = P_{t,i_{max,k}}$$

end if

Algorithm 5 SR2 for output FDD

At time step k;

1. **Variables assignment**

the same as in Algorithm 4

2. **Indexes of the filters with the maximal model probability**

the same as in Algorithm 4

3. **Reinitialization for the fault filter**

if $k = 1$ **or** ($k > 1$ **and** $i_{max,k-1} = 1$ **and** $i_{max,k} \neq 1$) **then**

$$\hat{x}_{af,j} = \hat{x}_{t,1}, \quad j = 1 \text{ to } n$$
$$P_{af,jj} = P_{t,1}, \quad j = 1 \text{ to } n$$
$$\hat{x}_{af,j} = \boldsymbol{x}_0, \quad j = n+1 \text{ to } n+m$$
$$P_{af,jj} = \mathbf{P}_0, \quad j = n+1 \text{ to } n+m$$

end if

4. **Reinitialization for the no-fault filter**

the same as in Algorithm 4

As demonstrated in Lu et al. [107], the fault filter is able to maintain an unbiased state and fault estimate in the presence of output faults through a certain reinitialization (i.e., the third step). Therefore, it can be assumed that in the presence of output faults, the state estimate and fault estimate of the fault filter can achieve an unbiased estimate. If the no-fault filter is not reinitialized each time step in the presence of faults, the filter will follow the measurement which contains the faults. This means that the innovation

$\boldsymbol{\gamma}_{nf}(k+1)$ of the no-fault filter is still small even though the state estimate is biased. Since $\boldsymbol{\gamma}(k+1)$ is the dominant term in Equation (4.5), the probability of the no-fault filter and the fault filter will be similar to each other especially when the output faults are small. Therefore, false alarms may be obtained especially in the presence of small faults. This is consistent with the simulation result shown in Figure 4.3(c). The smaller the magnitude of the faults, the more false alarms there will be, which is also shown in Figure 4.4(c).

If the no-fault filter is reinitialized using the fault filter, then the expectation of the state estimate of the no-fault filter is

$$E\{\hat{\boldsymbol{x}}_{nf}(k+1)\} = \boldsymbol{x}(k+1); \tag{4.13}$$

According to Equation (4.2), the innovation of the no-fault filter is

$$\begin{aligned} E\{\boldsymbol{\gamma}_{nf}(k+1)\} &= E\{\boldsymbol{y}(k+1)\} - E\{\boldsymbol{h}(\hat{\boldsymbol{x}}_{nf}(k+1))\} \\ &= \boldsymbol{f}^{o}(k+1). \end{aligned} \tag{4.14}$$

Since the innovation of the fault filter is zero-mean, its expectation is $E\{\boldsymbol{\gamma}_{af}(k+1)\} = \mathbf{0}$. Therefore, it is obvious that in the presence of output faults, the innovation of the no-fault filter is always larger than that of the fault filter under the condition that the no-fault filter is reinitialized by the fault filter. Therefore, it follows:

$$f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_{nf},\boldsymbol{Y}_{k-1}) < f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_{af},\boldsymbol{Y}_{k-1}), \tag{4.15}$$

which leads to

$$p_{nf} < p_{af}. \tag{4.16}$$

As can be seen, the probability of the no-fault filter will always be smaller than that of the fault filter. Therefore, false alarms are less likely to appear.

The two SR algorithms differ from each other in the third step, which is the reinitialization for the fault filter in the absence of output faults. The third step of the SR2 is able to maintain an unbiased state and fault estimate, as demonstrated by Lu et al. [107]. The third step of the SR1 proposes to perform the reinitialization as long as there are no output faults ($i_{max,k} = 1$). This requires more reinitializations compared to the SR2. The advantages and drawbacks of using these two different reinitialization strategies will be shown in the later sections.

For the reinitialization of the fault filter, the first n components of the state estimate ($\hat{x}_{af,j},\ j = 1:n$) of the fault filter are replaced by those of the no-fault filter. For the latter m components ($\hat{x}_{af,j},\ j = n+1:n+m$), a normal reinitialization such as the initialization of the KF can be applied.

By now, the complete block diagram of the DMAE approach can be given in Figure 4.2. The DMAE approach combined with the SR1 is termed as the DMAE1 approach and the DMAE approach combined with the SR2 is termed as the DMAE2 approach. The SR algorithms are vital to the DMAE approach. In this chapter, the DMAE approach is referred to the double-model-based approach with the SR algorithms. If there is no SR, the approach is called Double-Model Adaptive Estimation-No Selective Reinitialization (DMAE-NSR). The performance of the DMAE-NSR will also be shown to demonstrate the importance of the SR algorithms.

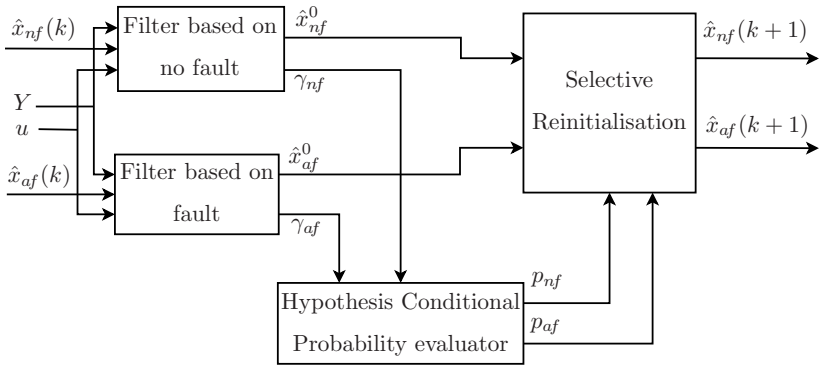


Figure 4.2: Block diagram for the DMAE approach

4

4.3.4. THE DMAE FOR FDD

The fault detection using the DMAE approach is the same as using the MMAE approach (Equation (4.8)). A fault alarm indicates a fault or faults in the system. However, which sensor or sensors are faulty needs to be further determined.

Since the fault filter of the DMAE approach is augmented with the m output faults $\boldsymbol{f}^o(k+1)$, the faults of each output are estimated by the DMAE. As mentioned by Edwards et al. [19], fault estimation is a powerful alternative to the detection of a fault when the location of the fault effect on the system is known. Furthermore, the fault information provided by the fault estimation approach can be directly used to isolate all faults [19]. This chapter makes use of a probability-weighted fault estimation to achieve fault isolation.

Let $\hat{f}^o_i$ denote the fault estimation of the ith output fault which is obtained by the fault filter. The probability-weighted fault estimation of the ith output can be determined using

$$\bar{f}^o_i = \hat{f}^o_i \cdot p_{af}, \qquad i = 1,2,...,m \tag{4.17}$$

where $\bar{f}^o_i$ is the probability-weighted estimation of the ith output fault. This probability-weighted fault estimation is used to isolate all faults through a threshold crossing. The threshold for isolating the faults is denoted as $\boldsymbol{T}_o$. It will be demonstrated that in the presence of little or no uncertainties, $\boldsymbol{T}_o$ can be readily selected. The FDD performance of the DMAE approaches will be presented in the following section.

4.4. PERFORMANCE OF THE DMAE

This section will compare the performance of the two proposed DMAE approaches to multiple-model-based approach. The fault detection example concerns the detection and estimation of consecutive and simultaneous faults of the aircraft ADS.

The structure of this section is as follow: Section 4.4.1 presents the kinematic model and the measurement model which are used for the FDD of the aircraft ADS. Section 4.4.2 compares the performance of the two DMAE approaches with multiple-model-based approach on the FDD of multiple faults. The performance of the two DMAE approaches with respect to simultaneous FDD is compared in Section 4.4.3. The sensitivity analysis

of the DMAE approaches is presented in Section 4.4.4. Finally, Section 4.4.5 summaries the performance comparison between the two DMAE approaches.

4.4.1. AIRCRAFT ADS FDD

The kinematic model of the aircraft including ADS faults is described as

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}\left(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t\right) + \mathbf{G}(\boldsymbol{x}(t))\,\boldsymbol{w}(t) \tag{4.18}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \boldsymbol{v}(t) + \boldsymbol{f}^o(t), \quad t = t_i,\ i = 1,2,\ldots \tag{4.19}$$

4

where $\boldsymbol{u}_m \in \mathbb{R}^b$ is the measured input, the other variables were defined in Equations (4.1) and (4.2). All the variables are dependent on time t. Note that for reasons of readability, t is omitted. The system equation variables are specifically described as follows:

$$\boldsymbol{x} = [V\ \alpha\ \beta\ \phi\ \theta\ \psi]^T \tag{4.20}$$

$$\boldsymbol{u}_m = \boldsymbol{u} + \boldsymbol{\lambda} + \boldsymbol{w} = [A_x\ A_y\ A_z\ p\ q\ r\ ]^T + \boldsymbol{\lambda} + \boldsymbol{w} \tag{4.21}$$

$$\boldsymbol{y} = [V_m\ \alpha_m\ \beta_m\ \phi_m\ \theta_m\ \psi_m]^T \tag{4.22}$$

$$\boldsymbol{\lambda} = [\lambda_x\ \lambda_y\ \lambda_z\ \lambda_p\ \lambda_q\ \lambda_r\ ]^T \tag{4.23}$$

$$\boldsymbol{w} = [w_x\ w_y\ w_z\ w_p\ w_q\ w_r\ ]^T \tag{4.24}$$

$$\boldsymbol{v} = [v_V\ v_\alpha\ v_\beta\ v_\phi\ v_\theta\ v_\psi]^T \tag{4.25}$$

$$\boldsymbol{f}^o = [f_V\ f_\alpha\ f_\beta\ 0\ 0\ 0]^T \tag{4.26}$$

where $[f_V\ f_\alpha\ f_\beta]^T$ are the faults of the ADS, i.e. f_V, f_α and f_β are the faults in the velocity sensor, angle of attack sensor and angle of sideslip sensor, respectively. The input u_m is the IMU measurement which measures the specific forces (A_x, A_y and A_z) and angular rates (roll rate p, pitch rate q and yaw rate r) of the aircraft. y is the output measurement which measures the air data information (true airspeed V, angle of attack α and angle of sideslip β) and Euler angles (roll angle ϕ, pitch angle θ and yaw angle ψ). $\boldsymbol{\lambda}$ is the bias in the IMU measurement. It is assumed that $\boldsymbol{\lambda} = \mathbf{0}$ in this section. In reality, this may not be true. The situation when $\boldsymbol{\lambda} \neq \mathbf{0}$ will be dealt with in Section 4.5. The input noise $\boldsymbol{w}$ and output noise $\boldsymbol{v}$ are assumed to be white Gaussian noise sequences. When this is not the case, there is a little inconsistency with the application of the theory [103]. However, it is still reasonable to assume the noises are Gaussian-distributed, especially when the aircraft dynamics are slow [103].

The kinematic equations are as follows:

$$\dot{V} = (A_x - g\sin\theta)\cos\alpha\cos\beta + (A_y + g\sin\phi\cos\theta)\sin\beta + (A_z + g\cos\phi\cos\theta)\sin\alpha\cos\beta \tag{4.27}$$

$$\dot{\alpha} = \frac{1}{V\cos\beta}(-A_x\sin\alpha + A_z\cos\alpha + g\cos\phi\cos\theta\cos\alpha + g\sin\theta\sin\alpha) + q - (p\cos\alpha + r\sin\alpha)\tan\beta \tag{4.28}$$

$$\dot{\beta} = \frac{1}{V}[-(A_x - g\sin\theta)\cos\alpha\sin\beta + (A_y + g\sin\phi\cos\theta)\cos\beta - (A_z + g\cos\phi\cos\theta)\sin\alpha\sin\beta] + p\sin\alpha - r\cos\alpha \tag{4.29}$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{4.30}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{4.31}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{4.32}$$

where g is the gravity constant. The measurement model is:

$$\boldsymbol{y} = \mathbf{H}\boldsymbol{x} + \boldsymbol{v} + \boldsymbol{f}^o \ , \ \mathbf{H} = \mathbf{I}_{6\times 6} \tag{4.33}$$

The objective of this chapter is to detect and identify f_V, f_α and f_β. The aircraft model is a simulated aircraft model of Cessna Citation II aircraft.

SELECTION OF Q AND R

The covariance matrix of $\boldsymbol{w}$ and $\boldsymbol{v}$ are defined as **Q** and **R** respectively. These matrices are critical to the performance of the filter. **R** can be inferred from the output measurement noise. As can be seen from the kinematic equations, it is obtained from the noise of the ADS and the sensors which measure the attitude angles. **Q** is usually used to represent modeling errors. If the influence of disturbances is assumed to be limited, then the only model uncertainty is in the input, i.e. the IMU measurements . Therefore, **Q** can be inferred from the IMU sensor noise which is defined in Equation (4.21) and Equation (4.24). The sensitivity of this **Q** selection in the presence of turbulence can be found in Section 4.4.4. For more detailed information about the implementation of the **Q** and **R** in the filter, the reader is referred to Mulder et al. [90].

4.4.2. MULTIPLE FAULT DETECTION AND DIAGNOSIS

This section will compare the performance of the proposed approaches with that of the multiple-model-based approaches. Since the SRMMAE2 in Lu et al. [107] is an improved version of the traditional MMAE approach, and its performance is better than the IMM approach when coping with small input faults, it is selected to compare with the two DMAE approaches. Furthermore, to demonstrate the importance of the SR algorithms, the DMAE-NSR approach is also compared to the two DMAE approaches.

For multiple faults simulated in this chapter, the faults occur between 10 s $< t <$ 20 s, 30 s $< t <$ 40 s and 50 s $< t <$ 60 s. The sampling rate in this chapter is 100 Hz. Two different scenarios are designed for the performance comparison.

SCENARIO 1

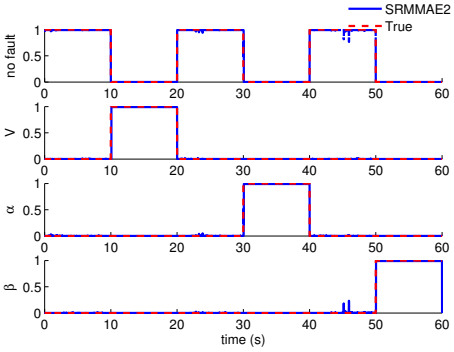
In this scenario, consecutive ADS faults are generated. For 10 s $< t <$ 20 s, the true airspeed V sensor fails with a bias of 10 m/s. For 30 s $< t <$ 40 s, there is a drift fault in the angle of attack α sensor. The drift rate is 10^{-1} rad/s. For 50 s $< t <$ 60 s, a bias with the value of 10^{-1} rad occurs in the angle of sideslip β sensor. Note that the drift fault is difficult to detect and identify due to its time-varying properties. The fault type and magnitude are shown by the solid line in Figure 4.3(b).

The SRMMAE2, DMAE-NSR and DMAE1 and DMAE2 approach are all used to detect and identify these consecutive faults. The results are shown in Figure 4.3. In the figures, the annotation "true" means the true model probability of elemental filters. The results of the DMAE2 approach are not shown, which will be explained.

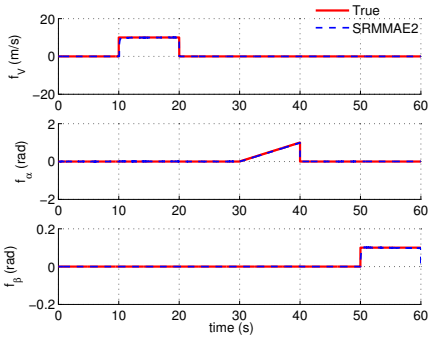
The model probabilities and probability-weighted fault estimation of the elemental filters using the SRMMAE2 approach are shown in Figures 4.3(a) and 4.3(b) respectively. Note there are four elemental filters using this approach. As can be seen, the performance is good despite some glitches in the probabilities. E.g., during 45 s $< t <$ 45.5 s, the model probability plot of the β sensor fault filter displays some probabilities which are larger than the lower probability bound (0.001). However, this does not trigger the alarm since the magnitude is smaller than the threshold p_m (0.9).

The results using the DMAE-NSR are shown in Figures 4.3(c) and 4.3(d). As can be seen from Figure 4.3(c), there are false alarms during 20 s $< t <$ 30 s and 40 s $< t <$ 50 s. During 10 s $< t <$ 20 s (there are output faults), the model probability plot of the fault filter displays several probabilities which are smaller than p_m (0.9). The reason why this occurs is explained in Section 4.3.3 (the motivation of performing reinitialization when there are output faults) and is explained in more detail in the following. When there are output faults, the covariance of the innovation of the no-fault filter $\boldsymbol{\gamma}_{nf}$ is smaller than the true covariance. This makes the no-fault filter trust the measurement (which contains output faults) more. Consequently, the innovation of the no-fault filter $\boldsymbol{\gamma}_{nf}$ will also be small. Therefore, according to Equations (4.4) and (4.5), the probability of the no-fault filter p_{nf} can be bigger than that of the fault filter p_{af}. This explains why in the presence of output faults, the model probability of the fault filter p_{af} can fall below the threshold p_m (0.9) and may even be smaller than that of the no-fault filter p_{nf} (at t = 14.16 s and t = 15.1 s). The situation will become worse when $\boldsymbol{f}^o$ is small, as can be seen from Figure 4.4(a). During 10 s $< t <$ 16 s, the model probability of the no-fault filter is several times significantly larger than 0.1 (which means the corresponding model probability of the fault filter p_{af} is smaller than 0.9). During 16 s $< t <$ 20 s, the model probability of the no-fault filter p_{nf} exceeds that of the fault filter p_{af} during the whole period except at t = 16.65 s and t = 17.37 s.

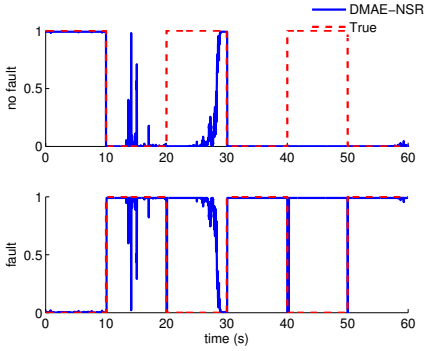
For the false alarms during 20 s $< t <$ 30 s and 40 s $< t <$ 50 s, detailed explanations can be found in Lu et al. [107] and only brief explanations are followed. During these two periods, the output faults are removed. The no-fault filter needs time to adapt to the no-fault situation. This situation will become worse when the faults are smaller, which explains why the probability of the no-fault model p_{nf} remains zero during the whole no-fault period (40 s $< t <$ 50 s). The estimation of f_V, f_α and f_β using the DMAE-NSR approach is shown in Figure 4.3(d). The estimation of f_α and f_β is acceptable while that of the f_V deviates from the true value.



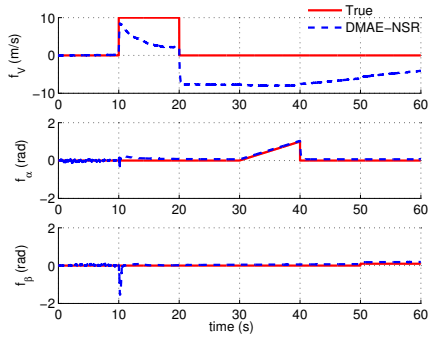
(a) Model probabilities of the four elemental filters using the SRMMAE2



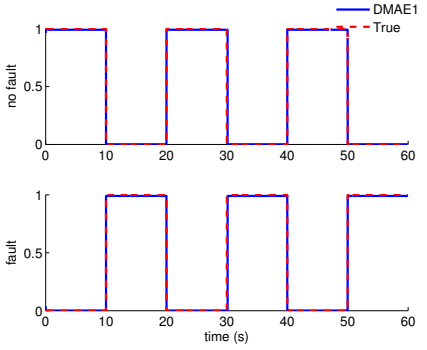
(b) Probability-weighted fault estimation of the elemental filters using the SRMMAE2



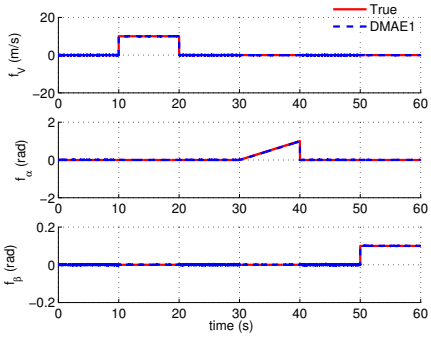
(c) Model probabilities of the no-fault filter and the fault filter using the DMAE-NSR



(d) Fault estimation of the elemental filters using the DMAE-NSR



(e) Model probabilities of the no-fault filter and the fault filter using the DMAE1



(f) Fault estimation of the fault filter using the DMAE1

Figure 4.3: Result of consecutive ADS FDD using the proposed DMAE1 approach and its comparison with the SRMMAE2 and DMAE-NSR approach (Scenario 1)

The model probability and the fault estimation of the proposed DMAE1 approach are shown in Figures 4.3(e) and 4.3(f), respectively. The performance is satisfactory. The model probability of the fault filter p_{af} approaches 1 when there are output faults and approaches the lower probability bound (0.001) when there are no output faults. No glitches are observed from the figure. Its fault estimation performance is as good as that of the SRMMAE2 approach. However, the DMAE1 approach only uses two models while the SRMMAE2 uses four models. The SRMMAE2 approach needs additional filtering to remove the spikes which is not necessary for the DMAE1 approach. This confirms the fact that more models may lead to performance degradation [99, 120]. The model probabilities using the DMAE2 approach are the same as those of the DMAE1 approach. The fault estimation using the DMAE2 approach is the same as that of the probability-weighted estimation using the SRMMAE2 approach since they use the same reinitialization scheme. Therefore, the results using the DMAE2 approach are omitted.

It has been shown that the performance of the DMAE approach is better than the SRMMAE2 approach. The computational load of the DMAE approach is also less intensive than multiple-model-based approach. Therefore, the performance comparison in the following simulations will be performed between the DMAE approaches.

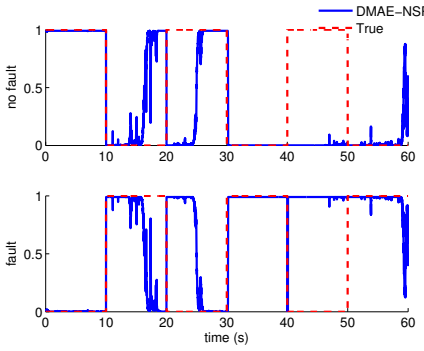
SCENARIO 2

In this scenario, the magnitude of the faults are smaller than those in Scenario 1, which can be seen from Figure 4.4(b). The drift rate of the α sensor is 2×10^{-2} rad/s, the bias of the V sensor is 1 m/s whereas that of the β sensor is $\pi/180$ rad. The results are shown in Figure 4.4.

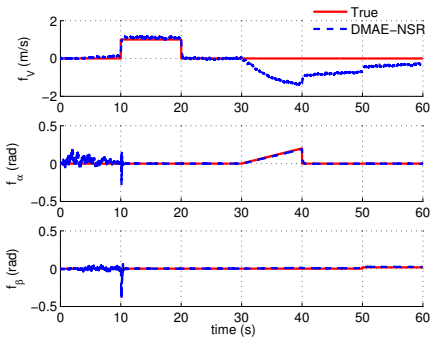
To emphasize the importance of the SR algorithms, the result using the DMAE-NSR is also shown in Figures 4.4(a) and 4.4(b). The amount of false alarms using the DMAE-NSR approach increases due to smaller faults. The estimation of f_V using the DMAE-NSR shows a divergence problem in the case there is another fault in a different sensor (f_α).

The results using the DMAE approaches are shown in Figures 4.4(c)- 4.4(f). As can be seen from the figures, the superiority of the DMAE approaches is more significant when the faults are smaller. Even though the faults are smaller, the DMAE approaches still maintain a satisfactory performance. The detection delay of f_α is larger than that of the f_V and f_β using the two approaches. This is because the drift fault is so small that it takes time to be detected. However, the detection response is still satisfactory. It takes the DMAE approaches less than 0.75 s to detect and identify such a small drift fault.

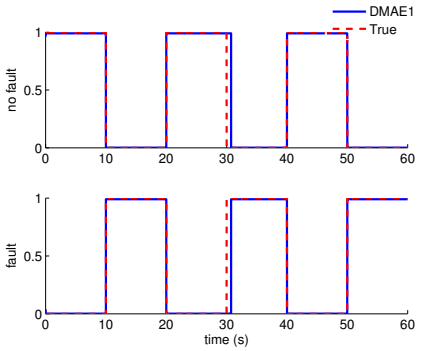
The estimation of f_V, f_α and f_β using the two approaches is shown in Figures 4.4(d) and 4.4(f) respectively. It can be noted that when there are no faults ($0\,\text{s} < t < 10\,\text{s}$, $20\,\text{s} < t < 30\,\text{s}$ and $40\,\text{s} < t < 50\,\text{s}$), the variances of the fault estimate using the DMAE1 approach are bigger than those using the DMAE2 approach, while in the presence of fault, their performance is almost the same. The reason is that they use different reinitialization strategies when there are no faults and the same reinitialization strategy in the presence of faults. When there are no faults, the DMAE1 keeps reinitializing the fault filter using the no-fault filter. It can be inferred that the variance of the fault estimate is the same as that of the corresponding measurement noise covariance. In this case, the variance of the V sensor fault estimate is the same as that of the V measurement noise



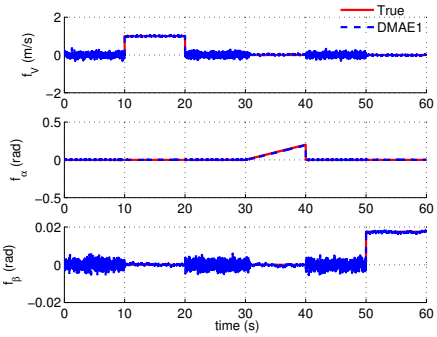
(a) Model probabilities of the elemental filters using the DMAE-NSR



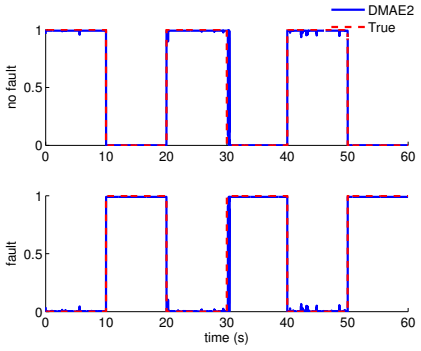
(b) Fault estimation of the elemental filters using the DMAE-NSR



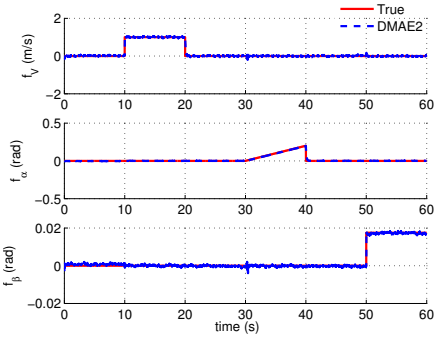
(c) Model probabilities of the no-fault filter and the fault filter using the DMAE1



(d) Fault estimation of the fault filter using the DMAE1

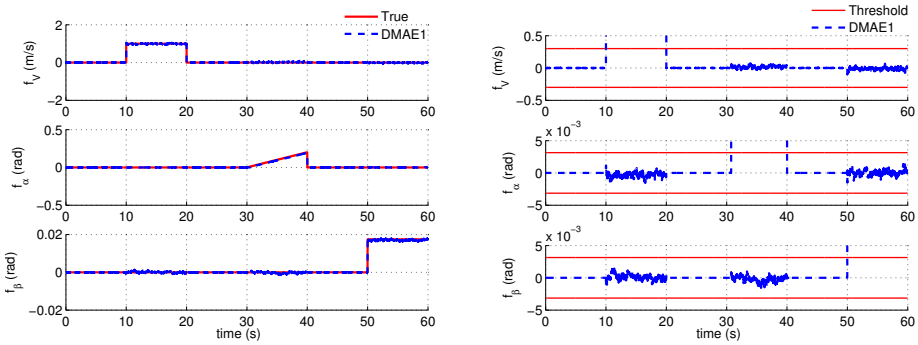


(e) Model probabilities of the no-fault filter and the fault filter using the DMAE2



(f) Fault estimation of the fault filter using the DMAE2

Figure 4.4: Result of consecutive ADS FDD using the DMAE1 and DMAE2 approach and their comparison with the DMAE-NSR approach (Scenario 2)



(a) Probability-weighted fault estimation of the fault filter using the DMAE1

(b) Fault isolation using probability-weighted fault estimation of the DMAE1

Figure 4.5: More results of using the DMAE1 approach (Scenario 2)

covariance. The variance of the fault estimate using the DMAE2 approach is not affected by that of the measurement noise covariance.

Using Equation (4.17), the probability-weighted estimation of f_V, f_α and f_β using the DMAE1 approach is computed and shown in Figure 4.5(a). This probability-weighted fault estimation is applied to isolate the faults, which is shown in Figure 4.5(b). The absolute value of the threshold $\boldsymbol{T}_o$ used to isolate the faults is: $[0.3, 0.18\pi/180, 0.18\pi/180]$. Note that the probability-weighted estimate of f_α during 30 s $< t <$ 30.75 s approaches 0 due to Equation (4.17). At $t = 30.75$ s, when the probability of the fault filter p_{af} exceeds 0.9 (Figure 4.4(c)), the estimate of f_α has already crossed its corresponding threshold ($0.18\pi/180$) while that of the f_V and f_β still remains in the threshold bound. Therefore, the fault detection and isolation can be achieved simultaneously. The result using the DMAE2 approach is similar to that of the DMAE1 approach and is omitted.

Remark: The benefit of FDI using the probability-weighted fault estimation can be seen from Figure 4.5(b). In the absence of faults (0 s $< t <$ 10 s, 20 s $< t <$ 30 s and 40 s $< t <$ 50 s), the probability-weighted estimation of f_V, f_α and f_β approaches zero due to Equation (4.17). Therefore, the possibility of a false detection, when there are no faults, is limited. When there are faults, due to the DMAE approaches, the fault estimate remains unbiased and is not influenced by the measurement noise. This also facilitates the selection of $\boldsymbol{T}_o$. Under these circumstances, $\boldsymbol{T}_o$ can be designed based on fault-free simulations. The selection of $\boldsymbol{T}_o$ using real flight data will be mentioned in Section 4.5.

4.4.3. SIMULTANEOUS FAULT DETECTION AND DIAGNOSIS

In this section, simultaneous FDD is considered. If normal MMAE or SRMMAE2 is used for detecting simultaneous faults, it is normally assumed that there are some delays between the two simultaneous faults [93, 103]. However, the DMAE approach in this chapter does not require this assumption. Furthermore, since the MMAE and the IMM require more than $m+1$ models to represent different fault situations, it increases the computational load significantly. Therefore, in this section, only the performance of the

two proposed approaches is compared.

For the simultaneous faults considered in this chapter, the faults occur in all the ADS during 10 s $< t <$ 20 s and 30 s $< t <$ 40 s. Two different scenarios are designed to demonstrate and compare the performance of the DMAE approaches.

SCENARIO 3

In order to test the performance of detecting smaller faults, the faults in this section are all bias faults. It should be noted that other types of faults can also be detected, which will also be shown in later sections. The type and the magnitude of the faults can be found in Figure 4.6(d). All the magnitudes of the faults are relatively small in this simulation. The absolute value of the f_V is 0.5 m/s while the absolute value of the f_α and f_β sensor are both $0.5\pi/180$ rad. The results of simultaneous FDD of the DMAE approaches are shown in Figure 4.6.

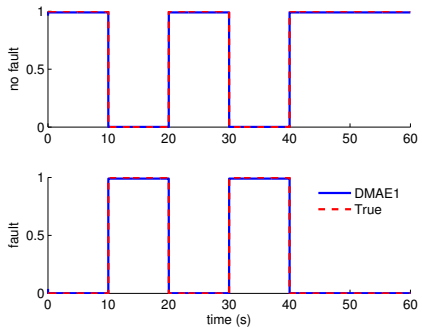
The model probabilities of the DMAE1 and DMAE2 of detecting the simultaneous faults of the ADS are shown in Figures 4.6(a) and 4.6(b) respectively. As can be seen, both approaches perform well. Even though the faults are small, all the faults are detected almost instantaneously. It takes 0.03 s to detect the simultaneous faults.

The estimation of f_V, f_α and f_β using the two approaches is shown in Figures 4.6(c) and 4.6(d) respectively. Note again, when there are no faults, the variances of the fault estimate using the DMAE1 approach are bigger than those using the DMAE2 approach while the variances of the fault estimation using the DMAE2 approach are not affected. This is not an issue for the DMAE1 approach as long as the faults can be detected. The probability-weighted estimation of f_V, f_α and f_β using the two approaches is calculated through Equation (4.17) and the result is shown in Figures 4.6(e) and 4.6(f) respectively. As can be seen, the performance of the two approaches is comparable. However, there may be problems when the fault magnitude is so small that it can be comparable to the standard deviation of the fault estimate, which will be shown in the following section.

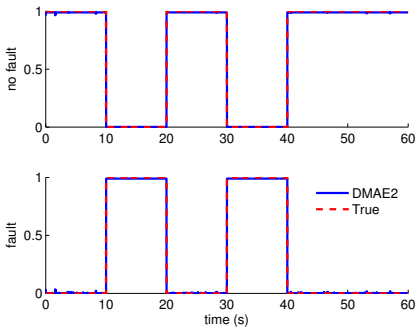
SCENARIO 4

In the previous simulation, the performance of the two DMAE approaches with respect to the simultaneous FDD are compared. In this section, the advantage of using the reinitialization strategy in the DMAE2 approach over the DMAE1 approach will be shown. The fault scenario is similar to Scenario 3, but the fault magnitudes are smaller. The absolute value of the V sensor faults is 0.5 m/s while the absolute value of the α and β sensor faults are both $0.2\pi/180$ rad. The results are shown in Figure 4.7.

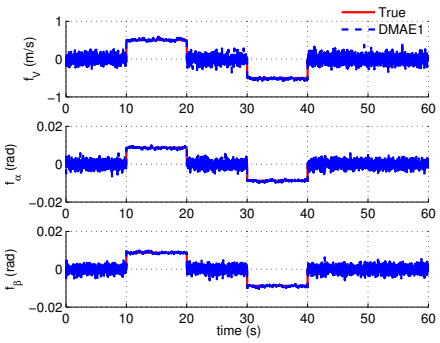
The model probabilities of the two approaches are shown in Figures 4.7(a) and 4.7(b) respectively. It can be seen that the detection of simultaneous faults using the DMAE1 approach is completely missed while the DMAE2 approach still maintains a satisfactory performance. The reason is that the DMAE1 approach keeps reinitializing the fault filter using the no-fault filter, which makes the standard deviation of the fault estimate comparable or even larger than the faults in this simulation. This makes the faults undetectable and demonstrates the advantage of using the reinitialization strategy in the DMAE2 approach, which maintains the differences between the state estimate and the fault estimate. The probability-weighted estimation of f_V, f_α and f_β using the DMAE1 and DMAE2 is shown in Figures 4.7(c) and 4.7(d) respectively. The probability-weighted fault estimation of the DMAE1 approach is zero due to Equation (4.17) while that of the



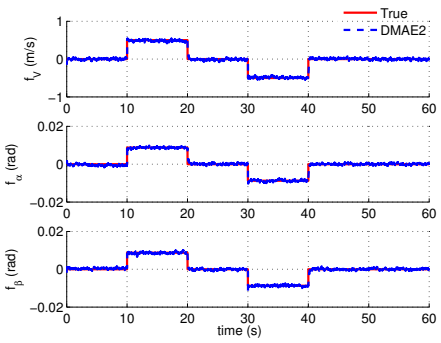
(a) Model probabilities of the no-fault filter and the fault filter using the DMAE1



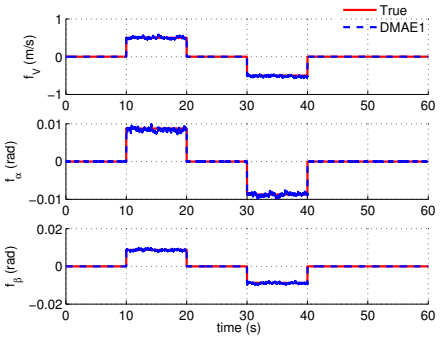
(b) Model probabilities of the no-fault filter and the fault filter using the DMAE2



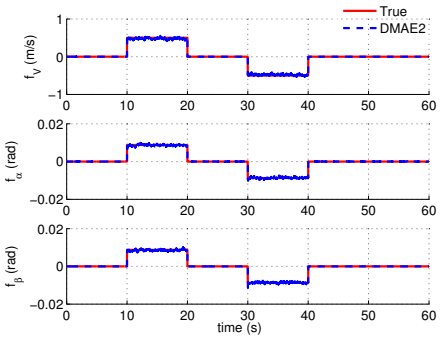
(c) Fault estimation of the fault filter using the DMAE1



(d) Fault estimation of the fault filter using the DMAE2

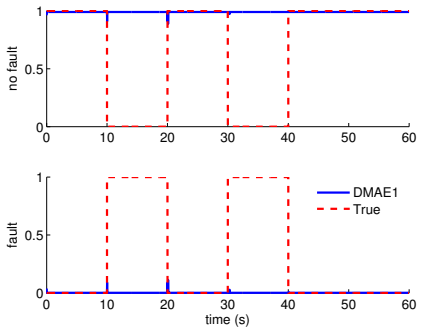


(e) Probability-weighted fault estimation of the fault filter using the DMAE1

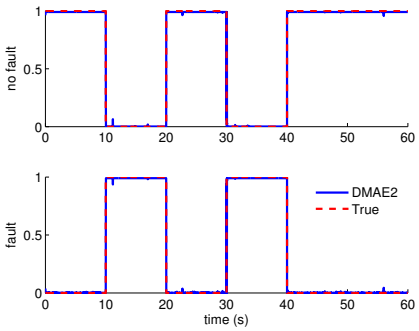


(f) Probability-weighted fault estimation of the fault filter using the DMAE2

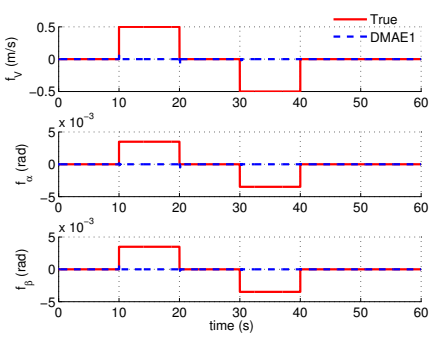
Figure 4.6: Results of simultaneous ADS FDD using the DMAE1 and DMAE2 approach (Scenario 3)



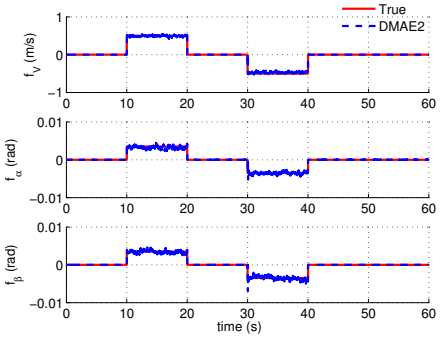
(a) Model probabilities of the no-fault filter and the fault filter using the DMAE1



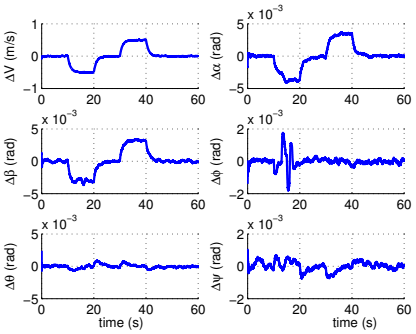
(b) Model probabilities of the no-fault filter and the fault filter using the DMAE2



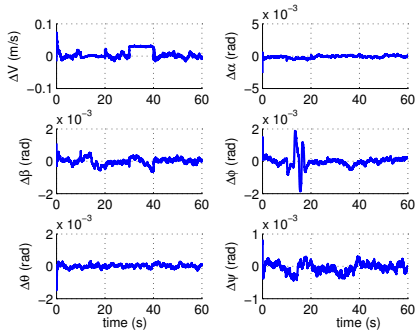
(c) Probability-weighted fault estimation of the fault filter using the DMAE1



(d) Probability-weighted fault estimation of the fault filter using the DMAE2

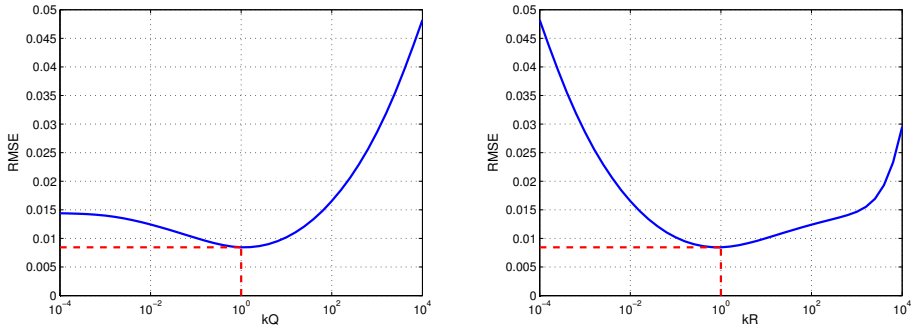


(e) State estimation error using the DMAE1



(f) State estimation error using the DMAE2

Figure 4.7: Result of simultaneous ADS FDD using the DMAE1 and DMAE2 approach (Scenario 4)



(a) Averaged root mean square error using the UKF when $10^{-4} \leq k_Q \leq 10^4$, the minimum is achieved at $k_Q = 1$

(b) Averaged root mean square error using the UKF when $10^{-4} \leq k_R \leq 10^4$, the minimum is achieved at $k_R = 1$

Figure 4.8: Performance of the UKF when k_Q and k_R varies

DMAE2 approach reconstructs the fault in an unbiased sense. The state estimation error of the two approaches is shown in Figures 4.7(e) and 4.7(f) respectively. It can be also seen that the state estimation error of the DMAE2 approach is unbiased as expected, while that of the DMAE1 approach shows a large error, especially in the presence of faults.

4.4.4. SENSITIVITY ANALYSIS

This section first analyses the sensitivity of the filter (UKF) used by the DMAE approaches. If there are no disturbances, the model described in Equations (4.18) and (4.19) can be considered to be exact. Under this circumstance, the sensitivity can be analyzed based on two situations: sensitivity to $\mathbf{Q}$ and sensitivity to $\mathbf{R}$.

Let $\mathbf{Q}_r$ and $\mathbf{R}_r$ denote the true process and measurement noise covariance matrix. The relationship between $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{Q}_r$, $\mathbf{R}_r$ can be denoted as follows:

$$\mathbf{Q} = k_Q \mathbf{Q}_r, \qquad \mathbf{R} = k_R \mathbf{R}_r \tag{4.34}$$

where k_Q and k_R are two coefficients. For linear systems, the estimation error covariance of the KF is minimum if $k_Q = 1$ and $k_R = 1$. For nonlinear systems, if the filter requires linearization, the optimal performance may not be necessarily achieved when $k_Q = 1$ and $k_R = 1$ due to the linearization errors. In this chapter, the UKF, which does not require linearization, is used. The sensitivity of the UKF to $\mathbf{Q}$ and $\mathbf{R}$ can be analyzed by varying k_Q and k_R respectively. The averaged root mean square error (RMSE) of the estimation using the UKF, when $10^{-4} \leq k_Q \leq 10^4$ or $10^{-4} \leq k_R \leq 10^4$, is shown in Figures 4.8(a) and 4.8(b), respectively.

As can be seen from the figures, in both situations, the minimum error is obtained when $k_Q = 1$ and $k_R = 1$. This indicates that the linearization error caused by the UKF is limited. Furthermore, it can be seen from the figure that either $k_Q \neq 1$ or $k_R \neq 1$ will cause an increase in the filter estimation error. This indicates the importance of the selection

of **Q** and **R**.

The above analysis is based on the assumption that there are no disturbances. In the presence of disturbances, the sensitivity of the two DMAE approaches, rather than that of the UKF, is shown below. The simulation situation is the same as in Section 4.4.2. In this chapter, only turbulence is considered as a disturbance, while other types of atmospheric disturbances, such as wind shears, are considered to be future work. A Dryden wind model is included in the simulation and the corresponding wind speeds are shown in Figure 4.9. The **Q** and **R** are selected in the same manner as in Section 4.4.1. The results of the DMAE1 and DMAE2 approach are shown in Figures 4.10(a) and 4.10(b), respectively.

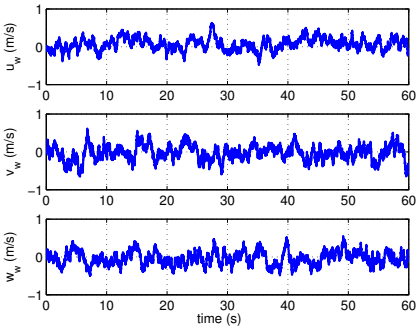


Figure 4.9: Wind speeds. u_w, v_w and w_w represent the wind speed in the north, east and down direction, respectively.

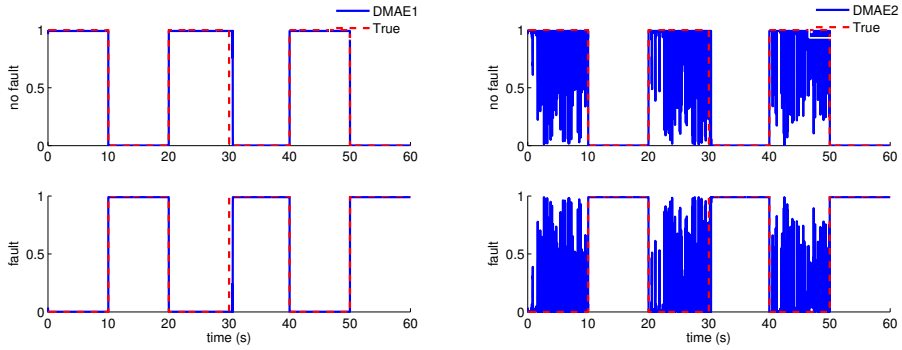
As can be seen from the figures, in the presence of faults (10 s < t < 20 s, 30 s < t < 40 s and 50 s < t < 60 s), both the performance of the DMAE approaches are satisfactory. However, in the absence of faults, the performance of the DMAE2 approach suffers from degradation whereas the DMAE1 approach maintains its satisfactory performance. The reason is that in the presence of faults, both the approaches keep performing reinitialization which reduces the influence of disturbance. In the absence of faults, the DMAE1 still keeps reinitializing whereas the DMAE2 does not, which makes the DMAE2 sensitive to disturbances.

4.4.5. COMPARISON BETWEEN THE DMAE1 AND DMAE2

The performance comparison between the DMAE1 and DMAE2 is summarized as follows:

1. Computational load: The DMAE1 approach is more computationally intensive since it performs the reinitialization strategy at every time step.

2. Detection range: The DMAE2 can detect smaller faults compared to the DMAE1 approach when there are no or little uncertainties and disturbances. The DMAE2 does not perform the reinitialization strategy at every time step, which maintains the differences between the two models.



(a) Model probabilities of the no-fault filter and fault filter using the DMAE1 in the presence of turbulence

(b) Model probabilities of the no-fault filter and fault filter using the DMAE2 in the presence of turbulence

Figure 4.10: Performance comparison of the DMAE1 and DMAE2 approach in the presence of turbulence

3. Sensitivity to disturbances: The DMAE1 approach is less sensitive to disturbances since it keeps performing the reinitialization while the DMAE2 is more sensitive to disturbances.

The performance comparison of the two DMAE approaches in this section is based on the simulated data of the aircraft. The performance of the two approaches when dealing with real flight test data will be validated and compared in the following section.

4.5. FAULT DETECTION AND DIAGNOSIS USING REAL FLIGHT DATA

The previous simulations are based on simulated data of an aircraft. This chapter intends to narrow down the gap between the academic and the real application by applying the proposed approaches to real flight data. It should be noted that a majority of FDD related research uses simulated data rather than real data, since the use of real data presents a number of challenges. The reasons are given below.

4.5.1. ISSUES RELATED TO FDD USING THE REAL FLIGHT DATA

There are many differences between using real flight data and simulated data. First of all, most studies dealing with aircraft sensor FDD usually use the aerodynamic model of the aircraft. The aerodynamic model can contain model uncertainties due to the calculation of the aerodynamic forces and moments. An accurate aerodynamic model is difficult and expensive to obtain. Hence, the performance of the model-based FDD approaches which mainly rely on the accuracy of the model may suffer from performance degradation.

There are also uncertainties in the measurements. Although calibrated by the manufacturer, there remain biases and inaccuracies in real life measurements. E.g., the biases of the IMU sensors can be temperature dependent, and as such cannot be completely

removed by calibration. The angle of attack α and angle of sideslip β in this chapter are measured by vanes which are mounted on the nose boom of the aircraft. However, the α-vane measures a combination of the true angle of attack, a fuselage-upwash induced angle of attack, a kinematically induced angle of attack and a vertical wind component [90, 91]. The β-vane also measures a combination of the true angle of sideslip, a fuselage-induced angle of sideslip and a lateral wind component. Therefore, the measured angle of attack and angle of sideslip also have to be calibrated in order to get the information about true angle of attack and angle of sideslip.

The noise level of the measurement has to be determined. Normally, the manufacturer provides this information. However, sometimes the noise level of the sensors may not be same with those specified by the manufacturer [91]. In this chapter, the noise levels in the accelerometers are more than an order of magnitude higher than those specified by the manufacturer, which increases the uncertainty.

There are also different update rates related to different measurements. As for the IMU sensors, the update rates are usually high whereas for the Global Positioning System (GPS) sensors, the update rate may be low. In this chapter, the update of the IMU and ADS sensors is 100 Hz whereas that of the GPS sensors is 1 Hz. In addition, the update rate of the magnetic compass which measures ψ is 10 Hz. The specifications of sensors are listed in Table 4.1 [91]. In the table, "n/a" means the information is not available.

Table 4.1: Specifications of the sensors

Variable	Unit	Standard Deviations	Sensor
V	[m/s]	n/a	Pitot-static probe
A_x, A_y, A_z	[m/s^2]	$2{\cdot}10^{-2}$	Q-Flex 3100 accelerometer (IMU)
α, β	[rad]	$3.5{\cdot}10^{-3}$	vane
ϕ, θ	[rad]	$8.7\cdot10^{-3}$	Sperry VG-14H vertical gyro
ψ	[rad]	n/a	gyrosyn compass
p, q, r	[rad/s]	$2{\cdot}10^{-3}$	LITEF μFORS rate gyro (IMU)

4.5.2. REAL FLIGHT DATA FDD WITHOUT CALIBRATION

The real flight data in this chapter is taken from flight tests performed with the Cessna Citation II aircraft, which is owned by the Delft University of Technology and the Dutch Aerospace Laboratory. The photo of the aircraft is given in Figure 4.11. The measurements of the flight test are shown in Figure 4.13(a). The fault scenario is multiple FDD, which means multiple faults are injected to the flight test data. It should be noted that the FDD performance of the DMAE approaches remains the same whether the faults occur during the flight or are injected to post flight data. This is due to the fact that they solely rely on the kinematic equations, which are independent of the actuator deflections.

The shape and magnitude of the faults can be found in the solid lines in Figure 4.12(b). For 10 s $< t <$ 20 s, the V sensor fails with a bias of 3 m/s. For 30 s $< t <$ 40 s, a bias with

Figure 4.11: Cessna Citation II aircraft, owned by Delft University of Technology and the Dutch Aerospace Laboratory

the value of $\pi/180$ rad occurs in the α sensor. For 50 s $< t <$ 60 s, a bias with the value of $\pi/180$ rad occurs in the β sensor.

In this section, sensor biases are not calibrated and the raw data is used. The result of the FDD is shown in Figure 4.12.

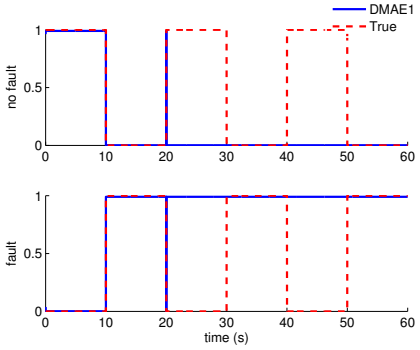
From Figures 4.12(a) and 4.12(b), it can be seen that the DMAE1 approach can detect and estimate the first fault (f_V) quickly and accurately. However, after the detection of the first fault, it keeps giving alarms. As can also be seen from the fault estimation figure, there are additional drift faults in the fault estimate. This is due to the biases of the sensors. The DMAE1 approach also treats the biases as faults, which results in giving alarms. The estimation of f_V, f_α and f_β using the DMAE2 approach is quite similar to that of the DMAE1 approach except in the beginning. The fault estimate of the DMAE2 approach, as shown in Figure 4.12(d), is not zero in the first 10 seconds while that of the DMAE1 approach is zero-mean. The non-zero-mean fault estimate of the DMAE2 approach leads to the alarms in the first ten seconds which is shown in Figure 4.12(c).
Remarks:

1. The poor FDD performance of the DMAE approaches is caused by the biases in the sensors which are inevitable in practice. However, the biases which exist before the start of the flight are not considered by the FDD system. A better performance can be obtained if the sensors are calibrated, which will be shown in Section 4.5.4.

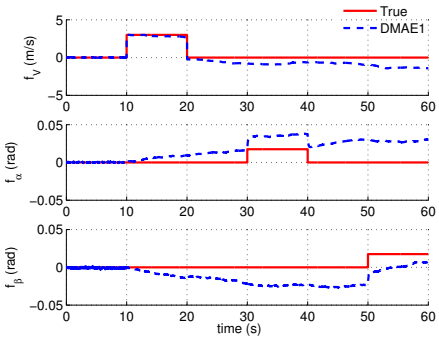
2. From the FDD point of view, the biases in the sensors can be interpreted as measurement uncertainties. In this case, the DMAE1 approach is better than the DMAE2 approach. Since the DMAE1 approach keeps reinitializing at every step, it reduces the influence of uncertainties. Therefore, the DMAE1 approach does not give alarms in the first ten seconds when the influence of the biases are still small.

4.5.3. CALIBRATION OF THE REAL FLIGHT DATA

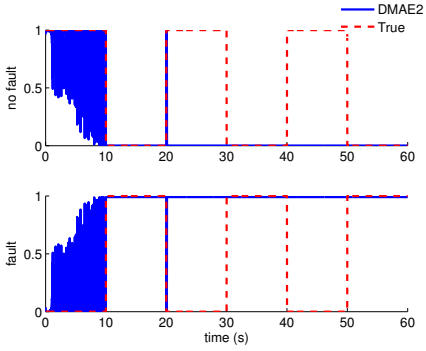
In practice, the calibration is usually done during certification. Normally, different maneuvers can be designed in order to calibrate the sensors [90]. The maneuvers in the



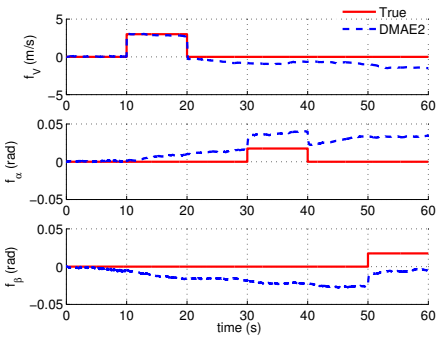
(a) Model probabilities of the no-fault filter and the fault filter using the DMAE1



(b) Fault estimation of the fault filter using the DMAE1



(c) Model probabilities of the no-fault filter and the fault filter using the DMAE2



(d) Fault estimation of the fault filter using the DMAE2

Figure 4.12: Consecutive ADS FDD for the real flight data using the DMAE1 and DMAE2 approach

flight test data used in this chapter were not designed for sensor calibration but for aerodynamic model identification purposes. This limits the accuracy of sensor calibration. For detailed methods for calibration, the reader is referred to Mulder et al. [90].

The process model used for calibration is the model given in Equation (4.18). In this section, the bias in the IMU sensors $\boldsymbol{\lambda} \neq \mathbf{0}$ and will be calibrated. The measurement

model for calibration is as follows [91]:

$$V_m = V + \upsilon_V; \tag{4.35}$$

$$\alpha_m = C_{\alpha 0} + (1 + C_{up})\alpha + \frac{x_\alpha q}{V} + \upsilon_\alpha \tag{4.36}$$

$$\beta_m = C_{\beta 0} + (1 + C_{si})\beta - \frac{x_\beta r}{V} + \upsilon_\beta \tag{4.37}$$

$$\phi_m = \phi + \upsilon_\phi \tag{4.38}$$

$$\theta_m = \theta + \upsilon_\theta \tag{4.39}$$

$$\psi_m = \psi + \upsilon_\psi \tag{4.40}$$

where $C_{\alpha 0}$ and $C_{\beta 0}$ are the zero shift of the α sensor and β sensor respectively, C_{up} and C_{si} are the upwash and sidewash coefficients. It should be noted that this is a relatively simple model to calibrate the sensors. The objective of calibration is to estimate

$$[\lambda_x,\ \lambda_y,\ \lambda_z,\ \lambda_p,\ \lambda_q,\ \lambda_r,\ C_{\alpha 0},\ C_{\beta 0},\ C_{up},\ C_{si}]^T. \tag{4.41}$$

The calibration is performed using the same method as in Mulder et al. [90]. The parameters in Equation (4.41) are augmented as additional states and are estimated using the UKF. The real data from the sensors are denoted by the solid lines in Figure 4.13(a). The calibration results are shown in Figure 4.13.

The state estimation using the filter is presented in Figure 4.13(b). The parameter estimation result related to the IMU sensor and ADS are shown in Figures 4.13(c) and 4.13(d), respectively. As can be seen from Figure 4.13(c), the biases of the rate gyros (λ_p, λ_q and λ_r) approach zero while those of the accelerometers (λ_x, λ_y and λ_z) are 0.05, 0.06 and -0.055 m/s^2, respectively. It can be noticed from Figure 4.13(d) that C_{up} is not a constant, but is a time varying parameter. However, for simplicity, this chapter uses a constant C_{up}.

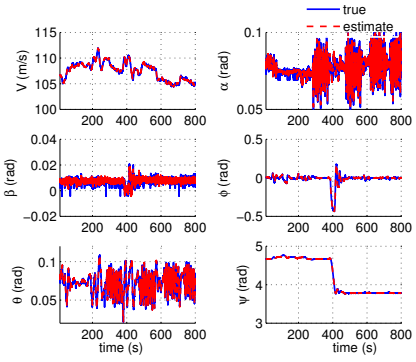
4.5.4. REAL FLIGHT DATA FDD WITH CALIBRATION

In this section, the DMAE approaches will be applied to FDD of the real flight data using the calibrated parameters from the previous subsection. It will be demonstrated that through calibration, the performance of the approaches can be improved.

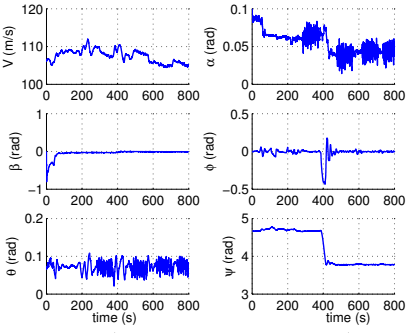
MULTIPLE FDD APPLIED TO THE REAL FLIGHT DATA

This subsection tests the performance of detecting multiple faults using the DMAE approaches. The fault scenario is the same as that of the FDD without calibration in Section 4.5.2. It should be noted that although the sensors are calibrated, the quality of the sensors in this chapter is not as good as those onboard commercial transport aircraft. On the other hand, the calibration is coarse. Therefore, the threshold for isolation $\boldsymbol{T_o}$, which was used when ideal sensors are assumed in Section 4.4.2, needs to be increased. The design of $\boldsymbol{T_o}$ is based on fault-free simulations. When using the real flight data, $\boldsymbol{T_o} = [1, 0.008, 0.008]$. The magnitude of $\boldsymbol{T_o}$ depends on the quality and calibration of the sensors.

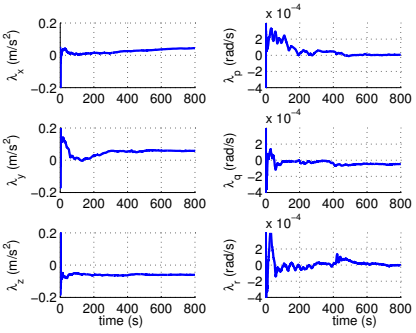
The results obtained using the two DMAE approaches are given in Figure 4.14. As can be seen from Figures 4.14(a) and 4.14(b), the performance of the DMAE1 approach



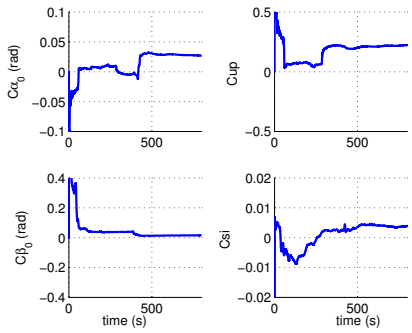
(a) The true measurement of the real flight test and its estimation by the UKF



(b) The state estimation using the UKF



(c) IMU bias estimation using the UKF



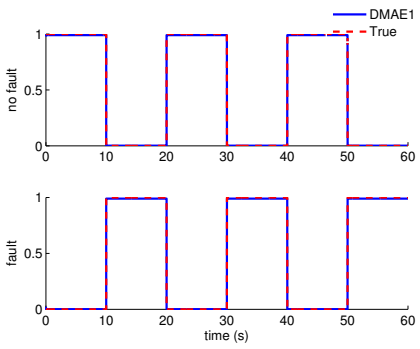
(d) ADS bias parameter estimation using the UKF

Figure 4.13: Flight path reconstruction using the real flight data and the sensor calibration

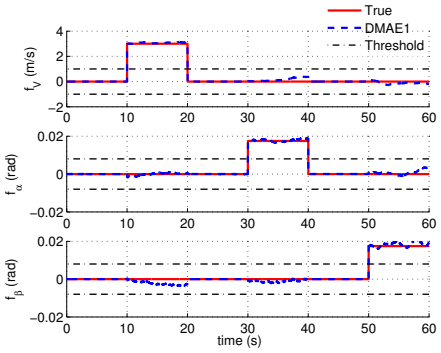
is satisfactory. There are no false alarms and all the faults in the ADS are estimated in an unbiased sense despite some offsets caused by the imperfect calibration. E.g., when there is a V sensor fault during 10 s $< t <$ 20 s, the probability-weighted estimate of f_α and f_β is larger than 0.001, which is caused by the imperfect calibration. The fault detection and isolation can be achieved simultaneously. The detection delays of f_V, f_α and f_β are all 1 time step, which demonstrates its good performance. Also notice that f_α and f_β with a bias of $\pi/180$ rad can be detected and estimated, which shows its ability to be applied in practice.

In contrast, the performance of the DMAE2 approach is not as good as that of the DMAE1 approach. The fault estimation performance is satisfactory, as seen in Figure 4.14(d). The model probabilities of the DMAE2 approach are given in Figure 4.14(c). As can be seen, in the presence of faults, the detection performance is good. However, when there are no faults (during 0 s $< t <$ 10 s, 20 s $< t <$ 30 s and 40 s $< t <$ 50 s), some glitches are present in the model probabilities although they do not trigger the alarm (their magnitude is smaller than 0.9). The calibration in this chapter is coarse which leads to un-

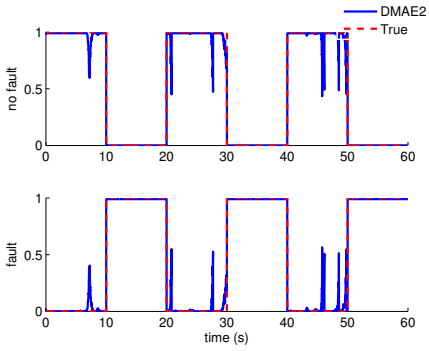
certainties. Therefore, the performance of the DMAE2 approach is degraded. The FDD performance of the DMAE2 approach can be improved by a better sensor calibration. However, this is not the focus of this chapter.



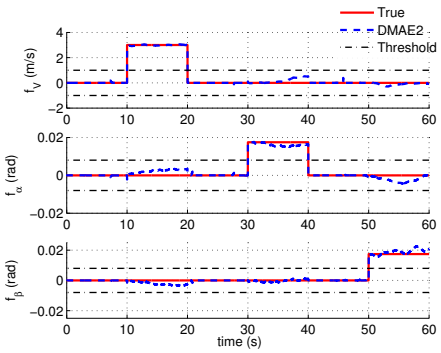
(a) Model probabilities of the no-fault filter and the fault filter using the DMAE1



(b) Probability-weighted fault estimation of the fault filter using the DMAE1



(c) Model probabilities of the no-fault filter and the fault filter using the DMAE2



(d) Probability-weighted fault estimation of the fault filter using the DMAE2

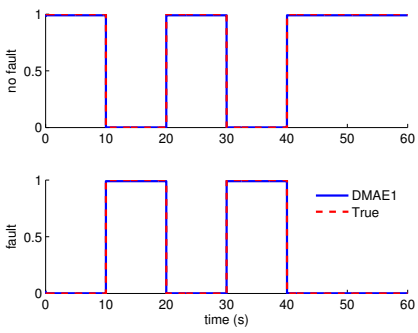
Figure 4.14: Consecutive ADS FDD of the real flight data using the DMAE1 and DMAE2 approach

SIMULTANEOUS FDD APPLIED TO THE REAL FLIGHT DATA

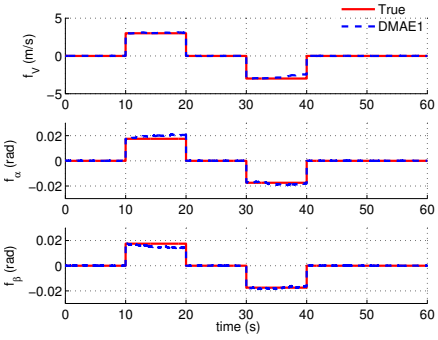
This section tests the FDD performance of simultaneous faults using the DMAE approaches. The absolute value of f_V is 3 m/s while the absolute value of f_α and f_β is $\pi/180$ rad. The results are given in Figure 4.15.

The result using the DMAE1 approach continues to present satisfactory FDD performance as shown in Figures 4.15(a) and 4.15(b). All the faults are detected within less than 0.02 s. The fault estimate is also satisfactory despite some small offsets.

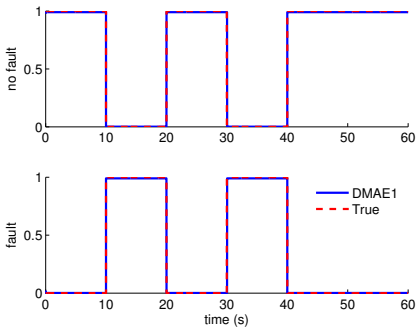
The probability-weighted estimation of f_V, f_α and f_β using the DMAE2 approach is similar to that using the DMAE1 approach as shown in Figure 4.15(b). The model probability of the fault filter using the DMAE2 approach also shows some glitches when



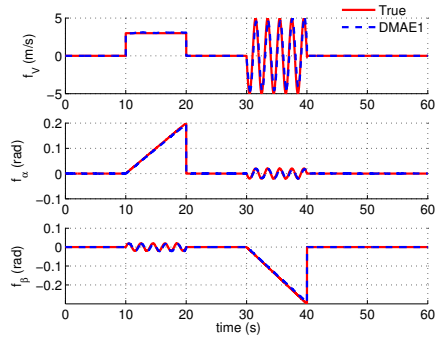
(a) Model probabilities of the no-fault filter and the fault filter using the DMAE1 in the presence of simultaneous bias faults



(b) Probability-weighted fault estimation of the fault filter using the DMAE1 in the presence of simultaneous bias faults



(c) Model probabilities of the no-fault filter and the fault filter using the DMAE1 in the presence of different types of faults



(d) Probability-weighted fault estimation of the fault filter using the DMAE1 in the presence of different types of faults

Figure 4.15: Simultaneous ADS FDD of the real flight data using the DMAE1 approach

there are no faults and is similar to the result shown in Figure 4.14(c). Therefore, the results of the DMAE2 are omitted here.

To demonstrate the ability of detecting different type of faults, another type of fault, i.e., the oscillatory fault [39] is included. This kind of fault is mainly caused by electronic components in fault mode generating spurious sinusoidal signals [39]. For 10 s $< t <$ 20 s, the V sensor fails with a bias with a value of 3 m/s, there is a drift fault in the α sensor fails with a drift rate of 2×10^{-2} rad/s, the oscillatory fault of the β sensor is $0.02\sin(0.01\pi t)$ rad. For 30 s $< t <$ 40 s, an oscillatory fault $(-5\sin(0.01\pi t)$ rad) occurs in the V sensor, another oscillatory fault $(-0.02\sin(0.01\pi t)$ rad) occurs in the α sensor while there is a drift fault in the β sensor with a drift rate of -3×10^{-2} rad/s.

As can be seen from Figures 4.15(c) and 4.15(d), the DMAE1 approach can also estimate different types of faults. There are no false alarms and the probability-weighted fault estimation performance is good. This demonstrates its ability to detect and identify various type of faults.

4.6. CONCLUSIONS

This chapter has proposed a practical FDD system for aircraft ADS. Two DMAE approaches are proposed which reduce the number of models in the model set of the multiple-model-based approach to two. This reduces the computational load significantly especially when there are simultaneous faults. The performance of the DMAE approach is guaranteed by the two SR algorithms. It is shown by simulation that in the absence of model uncertainties, the DMAE2 approach, which does not perform the reinitialization at every step, outperforms the DMAE1 approach. Specifically, the DMAE2 approach can detect smaller faults and the computational load is smaller compared to the DMAE1 approach.

This chapter also addresses the aircraft sensor FDD using real flight data. Important issues related to use of real flight data are addressed. It is shown that both approaches can be applied to the real flight FDD by first performing sensor calibration. In the presence of model uncertainties, the DMAE1 approach outperforms the DMAE2 approach since the DMAE1 approach performs the reinitialization at every step which reduces the influence of model uncertainties. Recall that the sensor calibration in this chapter is rough due to the limited source of maneuvers in the real flight data. The obtained result is satisfactory, which verifies their feasibility for real application.

The fact that the DMAE approaches reduce the computational load of the MM-based approach while still guaranteeing fast detection and accurate estimation of the faults, makes them applicable and suitable for real application. In the future, the DMAE approach can be incorporated in the control system to enhance the safety of aircraft.

II

Sensor Fault Diagnosis and Fault-Tolerant Control in the Presence of Model Uncertainties and Disturbances

5

AIRCRAFT INERTIAL MEASUREMENT UNIT FAULT IDENTIFICATION WITH APPLICATION TO REAL FLIGHT DATA

In Part I (Chapters 2, 3 and 4), IMU and ADS FDD approaches, which consider the model uncertainties, were proposed. However, effects of external disturbances such as atmospheric turbulence were not fully considered. In this chapter, a first attempt to cope with aircraft sensor FDD in the presence of external disturbances is made. First, the model used for IMU FDD in the presence of turbulence is introduced. The state and fault estimation is dealt with by the Iterated Optimal Two-Stage Extended Kalman Filter. The performance of the proposed approaches is validated using both simulated and real flight test data.

The software of the work in this chapter is available at: https://www.researchgate.net/profile/Peng_Lu15/publications?pubType=dataset.

Parts of this chapter are based on:

P Lu, L Van Eykeren, E van Kampen, C C de Visser, Q P Chu. Aircraft Inertial Measurement Unit Fault Identification with Application to Real Flight Data. *Journal of Guidance, Control and Dynamics*, 38(12), 2467-2475, 2015. [123].

5.1. INTRODUCTION

FDD is an important problem in aerospace engineering [11]. For flight control systems, sensor or actuator faults may cause serious problems. Therefore, quick detection and isolation of these faults is highly desirable. During the last few decades, many approaches have been proposed for FDD [12, 13, 66, 67]. Some recent advances and trends can be found in Zolghadri [14] and Goupil [9, 124]. Presently, the sensor FDD onboard the aircraft is mainly based on hardware redundancy. Despite the hardware redundant setup, recent airliner accidents have indicated that sensor malfunctions can result in critical failures [73]. One recent example of an IMU failure can be found in the accident investigation of a Qantas Airbus A330-303 Flight QF72. In this case, one of the air data inertial reference units was providing erroneous data on many parameters to other aircraft systems, which led to a flight upset and 11 occupants seriously injured [73]. Therefore, fault detection and identification of the IMU faults is important to the safety of the aircraft.

Aircraft IMU fault identification has been dealt with by a number of researchers [97, 116, 125]. Most research uses the aerodynamic model of the aircraft. However, the calculation of the aerodynamic forces and moments may contain uncertainties. Alwi and Edwards [125] use Linear Parametric Varying models which may explicitly contain information about the aerodynamic coefficient variations over the flight envelope. They use the same approach to cope with yaw rate sensor faults in the ADDSAFE [124] benchmark problem [116]. There are also studies using robust FDD approach [95, 113] to detect actuator faults which can cope with the model uncertainties. Alternatively, the kinematic model (KM) of the aircraft can be used instead of the aerodynamic model. It should be noted that few researchers [83, 92, 114, 126] use the KM to cope with aircraft sensor fault detection. These studies show that using the KM can reduce the influence of model uncertainties caused by the calculation of the aerodynamic forces and moments. Van Eykeren and Chu [114] use a Sliding Mode Differentiator [127] to estimate the IMU faults. However, it requires the selection of a Lipschitz constant [114, 127]. Lu et al. [83] use a Selective-Reinitialization Multiple-Model Adaptive Estimation approach to estimate the faults in the IMU. However, the computational load is intensive when dealing with simultaneous faults. Furthermore, both of them [83, 114] assume the absence of disturbances. More recently, Lu et al. [92] used the KM to detect the air data sensor faults of real flight test data. In [92], the air data measurements are the output of the KM. In the present chapter, the IMU measurements are the input, which leads to an input fault identification problem.

In this study, an IOTSEKF, which improves the performance of the OTSEKF [33, 128, 129] when dealing with nonlinear systems, is applied to estimate both the system states and the IMU faults. The simulation data is taken from the ADMIRE benchmark model [130]. The computational complexity of the IOTSEKF is compared to an Augmented Extended Kalman Filter (AEKF).

The second contribution is that the influence of disturbances such as turbulence is considered. The original KM (KM1) is modified by proposing a novel KM (KM2) with different state and measurement vectors. It is shown that KM2 is less affected by varying wind such as turbulence. The performance of the IMU sensor fault identification using KM1 and KM2 is compared. Simulation results demonstrate the effectiveness of using

KM2 even in the presence of turbulence.

The third contribution of this chapter lies in the fact that it uses real flight data to validate the proposed approaches for IMU fault identification. Few researchers [97, 131] identify the IMU sensor faults using real flight data. Berdjag et al. [70, 97] detect oscillatory faults [39] using three inertial aircraft sensors. Their approach is validated on a normalized real flight data set. Hu and Seiler [131] assess the false alarm probability of a simple, model-based UAV fault detection system. Freeman et al. [132] use the same UAV for the detection of aileron faults. The present chapter uses real flight test data to validate the performance of the proposed approaches on identifying the faults which are injected into the IMU sensor data. The fault types not only include bias and drift, but oscillatory faults as well. The real flight data is taken from a Cessna Citation II aircraft. The results show that the proposed approaches are able to identify the faults in the IMU sensors, which verifies that they can be applied in practice.

The structure of this chapter is as follows: Section 5.2 presents the novel KM which includes the influence of the IMU sensor faults. In Section 5.3, the IMU fault identification using KM2 is dealt with by the IOTSEKF. The performance is also compared to that using KM1 in the absence and presence of turbulence. The performance of the IOTSEKF using KM1 and KM2 is further validated using real flight test data in Section 5.4. Finally, the conclusions are presented in Section 5.5.

5.2. NOVEL AIRCRAFT KINEMATIC MODEL WITH IMU SENSOR FAULTS

The original KM [73, 90], which is called "KM1" in this chapter, is used for flight path reconstruction. Van Eykeren and Chu [126], and Lu et al. [83] used KM1 to detect and identify aircraft sensor faults. However, KM1 does not incorporate the influence of turbulence. In order to cope with that, a new KM, called "KM2", is proposed in this section.

5.2.1. KM2 WITH IMU SENSOR FAULTS

The general model of the aircraft using KM2 including IMU sensor faults is described as follows:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), \boldsymbol{f}^i(t), t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) \tag{5.1}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) \tag{5.2}$$

$$\boldsymbol{y}_m(t) = \boldsymbol{y}(t) + \boldsymbol{v}(t), \quad t = t_i, \ i = 1, 2, \ldots \tag{5.3}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the system states, $\boldsymbol{u}_m \in \mathbb{R}^b$ the measured input, $\boldsymbol{y}_m \in \mathbb{R}^m$ the measured output. $\boldsymbol{w}$ and $\boldsymbol{v}$ are assumed to be white Gaussian noise sequences. The function $\boldsymbol{f}^i$ represents the input faults. The system equation variables are defined as

follows:

$$\boldsymbol{x} = [u^B_{\text{GS}}\ v^B_{\text{GS}}\ w^B_{\text{GS}}\ \phi\ \theta\ \psi]^T \tag{5.4}$$

$$\boldsymbol{u}_m = [A_{xm}\ A_{ym}\ A_{zm}\ p_m\ q_m\ r_m] = [A_x\ A_y\ A_z\ p\ q\ r\ ]^T + \boldsymbol{w} + \boldsymbol{f}^i \tag{5.5}$$

$$\boldsymbol{y}_m = [u_{GSm}\ v_{GSm}\ w_{GSm}\ \phi_m\ \theta_m\ \psi_m]^T \tag{5.6}$$

$$\boldsymbol{w} = [w_x\ w_y\ w_z\ w_p\ w_q\ w_r\ ]^T \tag{5.7}$$

$$\boldsymbol{f}^i = [f_{Ax}\ f_{Ay}\ f_{Az}\ f_p\ f_q\ f_r]^T \tag{5.8}$$

$$\boldsymbol{v} = [v_{u_{GS}}\ v_{v_{GS}}\ v_{w_{GS}}\ v_\phi\ v_\theta\ v_\psi]^T \tag{5.9}$$

where the u^B_{GS}, v^B_{GS} and w^B_{GS} are the ground speed components in the body reference frame. A_x, A_y, A_z are the specific forces, ϕ, θ and ψ are the Euler angles and p, q, r are the rotational rates of the aircraft. It can be noticed that $\boldsymbol{u}_m$ is the measurement of the IMU and $\boldsymbol{f}^i$ are the faults in the IMU sensors. f_{Ax}, f_{Ay} and f_{Az} represent the faults in the accelerometers while f_p, f_q and f_r represent the faults in the rate gyros.

The process model of KM2 is given as follows:

$$\dot{u}^B_{\text{GS}} = (A_{xm} - f_x) + v^B_{\text{GS}}(r_m - f_r) - w^B_{\text{GS}}(q_m - f_q) - g\sin\theta \tag{5.10}$$

$$\dot{v}^B_{\text{GS}} = (A_{ym} - f_y) - u^B_{\text{GS}}(r_m - f_r) + w^B_{\text{GS}}(p_m - f_p) + g\cos\theta\sin\phi \tag{5.11}$$

$$\dot{w}^B_{\text{GS}} = (A_{zm} - f_z) + u^B_{\text{GS}}(q_m - f_q) - v^B_{\text{GS}}(p_m - f_p) + g\cos\theta\cos\phi \tag{5.12}$$

$$\dot{\phi} = (p_m - f_p) + (q_m - f_q)\sin\phi\tan\theta + (r_m - f_r)\cos\phi\tan\theta \tag{5.13}$$

$$\dot{\theta} = (q_m - f_q)\cos\phi - (r_m - f_r)\sin\phi \tag{5.14}$$

$$\dot{\psi} = (q_m - f_q)\frac{\sin\phi}{\cos\theta} + (r_m - f_r)\frac{\cos\phi}{\cos\theta} \tag{5.15}$$

The measurement model of KM2 is

$$u_{GSm} = \mathbf{T}_{eb}[u^B_{\text{GS}}, v^B_{\text{GS}}, w^B_{\text{GS}}]^T + v_{u_{GS}} \tag{5.16}$$

$$v_{GSm} = \mathbf{T}_{eb}[u^B_{\text{GS}}, v^B_{\text{GS}}, w^B_{\text{GS}}]^T + v_{v_{GS}} \tag{5.17}$$

$$w_{GSm} = \mathbf{T}_{eb}[u^B_{\text{GS}}, v^B_{\text{GS}}, w^B_{\text{GS}}]^T + v_{w_{GS}} \tag{5.18}$$

$$\phi_m = \phi + v_\phi \tag{5.19}$$

$$\theta_m = \theta + v_\theta \tag{5.20}$$

$$\psi_m = \psi + v_\psi \tag{5.21}$$

where u_{GSm}, v_{GSm} and w_{GSm} are the measurements for the ground velocity expressed in the earth reference frame. $\mathbf{T}_{eb}$, which can be found in Appendix A, is the transformation matrix from the body frame to the earth frame.

If KM1 is used, then the system variables are defined as follows [85]:

$$\boldsymbol{x}' = [u_a\ v_a\ w_a\ \phi\ \theta\ \psi]^T \tag{5.22}$$

$$\boldsymbol{y}'_m = [V_m\ \alpha_m\ \beta_m\ \phi_m\ \theta_m\ \psi_m]^T \tag{5.23}$$

$$\boldsymbol{v}' = [v_V\ v_\alpha\ v_\beta\ v_\phi\ v_\theta\ v_\psi]^T \tag{5.24}$$

where u_a, v_a, w_a are the airspeed components defined in the body-fixed reference frame. V_m, α_m and β_m are the measurements of the true airspeed, angle of attack and angle of sideslip. The detailed equations can be found in Appendix B.

The advantages of using KM2 over KM1 for IMU fault identification is explained in the following. Let $\boldsymbol{V}_g^B$, $\boldsymbol{V}_a^B$ and $\boldsymbol{V}_w^B$ denote the ground speed, air speed and wind speed components expressed in the body frame respectively. $\boldsymbol{\omega}$ and $\boldsymbol{A}$ denote the angular rates and specific forces respectively. g is the gravitational acceleration. The derivative of $\boldsymbol{V}_g^B$ is as follows:

$$\dot{\boldsymbol{V}}_g^B = \dot{\boldsymbol{V}}_a^B + \dot{\boldsymbol{V}}_w^B = \boldsymbol{A} + \mathbf{T}_{be}\boldsymbol{g} - \boldsymbol{\omega} \times \boldsymbol{V}_g^B \tag{5.25}$$

where $\mathbf{T}_{be} = \mathbf{T}_{eb}^T$. Since $\boldsymbol{V}_g^B = \boldsymbol{V}_a^B + \boldsymbol{V}_w^B$, the derivative of $\boldsymbol{V}_a^B$ is as follows:

$$\dot{\boldsymbol{V}}_a^B = \boldsymbol{A} + \mathbf{T}_{be}\boldsymbol{g} - \boldsymbol{\omega} \times \boldsymbol{V}_a^B - (\dot{\boldsymbol{V}}_w^B + \boldsymbol{\omega} \times \boldsymbol{V}_w^B) \tag{5.26}$$

If the wind is constant, i.e. $\frac{d\boldsymbol{V}_w^I}{dt_I} = \mathbf{0}$, then $\frac{d\boldsymbol{V}_w^B}{dt_I} = \mathbf{0}$. The subscript I means the inertial reference frame. Since $\frac{d\boldsymbol{V}_w^B}{dt_I} = \dot{\boldsymbol{V}}_w^B + \boldsymbol{\omega} \times \boldsymbol{V}_w^B = \mathbf{0}$, Equation (5.26) reduces to

$$\dot{\boldsymbol{V}}_a^B = \boldsymbol{A} + \mathbf{T}_{be}\boldsymbol{g} - \boldsymbol{\omega} \times \boldsymbol{V}_a^B \tag{5.27}$$

KM1 is based on Equation (5.27) while KM2 is based on Equation (5.25). Apparently, when the wind is not constant, Equation (5.27) is no longer exact. Therefore, KM1 is influenced by non-constant wind such as turbulence while KM2 is not.

Remarks: Another advantage using this KM2 is that the fault identification of the IMU sensors and ADS is separated. In this situation, the assumption that there are no faults in the ADS, which is made when using KM1, can be removed if KM2 is used since the ADS measurements are not required by the IMU fault identification.

5.3. IMU SENSOR FI USING KM2

This section uses the proposed KM2 to identify the faults in the IMU sensors ($\boldsymbol{f}^i$). First, the fault scenario of the IMU sensors is shown. Then, the IOTSEKF, which is an improved version of the OTSEKF [33], is introduced to estimate $\boldsymbol{f}^i$. The performance of the IOTSEKF using KM2 is compared to that using the KM1 to demonstrate the advantages of using KM2.

5.3.1. FAULT SCENARIO USING THE SIMULATED DATA

The simulated measurement data is taken from the ADMIRE aircraft benchmark model. This simulation data is used in this section to validate the performance of the proposed approaches while the real flight test data is used in Section 5.4. The noise variances and the update rates of the sensors are given in Table 5.1. The update rate of the GPS receivers was slow a decade ago. However, recently, rates of 10, 20, 50 or even 100 Hz have been used for various purposes [133]. In addition, the GPS receiver is only used to measure the speed in this research. In this section, the update rate of the GPS measurements is 10 Hz.

Table 5.1: Update rates and standard deviations of the noises in the sensors

Measurement	Update rate	Standard deviation	Unit
V_m	100 Hz	0.01	[m/s]
u_{GSm}, v_{GSm}, w_{GSm}	10 Hz	0.01	[m/s]
α_m, β_m	100 Hz	$0.01\pi/180$	[rad]
ϕ_m, θ_m, ψ_m	100 Hz	$0.01\pi/180$	[rad]
A_{xm}, A_{ym}, A_{zm}	100 Hz	0.01	[m/s^2]
p_m, q_m, r_m	100 Hz	$0.01\pi/180$	[rad/s]

In this chapter, all the IMU sensors fail simultaneously during certain a period. Furthermore, different types of faults are considered, including bias faults and drift faults, as well as oscillatory faults. The detailed fault scenario in this section, including the fault type and magnitude, is presented in Table 5.2. The fault scenario is arbitrarily chosen to test the performance of the proposed approach.

5

5.3.2. ITERATED OPTIMAL TWO-STAGE KALMAN FILTER

This section first presents the equations of the IOTSEKF for the estimation of $\boldsymbol{f}^i$. Then, the computational complexity comparison between the IOTSEKF and an AEKF is presented.

THE IOTSEKF

The OTSEKF can estimate the bias of the system in an optimal minimum variance error sense [33]. It is composed of a modified bias-free filter and a bias filter [33]. However, its performance may be degraded when dealing with nonlinear systems [85]. In order to cope with that, an IOTSEKF is proposed. The state estimate and its covariance matrix using the IOTSEKF are given as follows:

$$\hat{\boldsymbol{x}}_{k|k} = \bar{\boldsymbol{x}}_{k|k} + \mathbf{V}_k \boldsymbol{f}^i{}_{k|k} \tag{5.28}$$

$$\mathbf{P}^x_{k|k} = \mathbf{P}^{\bar{x}}_{k|k} + \mathbf{V}_k \mathbf{P}^{\boldsymbol{f}^i}_{k|k} \mathbf{V}_k^T \tag{5.29}$$

The bias-free filter of the IOTSEKF is as follows:

$$\boldsymbol{\eta}_1 = \bar{\boldsymbol{x}}_{k|k-1} = \Phi_{k-1} \bar{\boldsymbol{x}}_{k-1|k-1} + \bar{\boldsymbol{u}}_{k-1} \tag{5.30}$$

$$\mathbf{P}^{\bar{x}}_{k|k-1} = \Phi_{k-1} \mathbf{P}^{\bar{x}}_{k-1|k-1} \Phi^T_{k-1} + \bar{\mathbf{Q}}_{k-1} \tag{5.31}$$

where Φ_{k-1} is calculated as follows:

$$\Phi_{k-1} = e^{\mathbf{F}_{k-1}\Delta t} = \sum_n^{\infty} \frac{\mathbf{F}^n_{k-1}(\Delta t)^n}{n!}, \; \Gamma_{k-1} = \int_{t_{k-1}}^{t_k} \Phi_{k-1} \mathbf{G}(\boldsymbol{x}_{k-\bar{1}|k-1}) dt, \tag{5.32}$$

$$\mathbf{F}_{k-1} = \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), \boldsymbol{f}^i(t), t)}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k-1|k-1}}, \qquad \mathbf{H}_k = \frac{\partial \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t)}{\partial \boldsymbol{x}}|_{\boldsymbol{x}=\boldsymbol{\eta}_1}. \tag{5.33}$$

Table 5.2: Fault Scenario

Time interval	Sensor	Fault type	Fault magnitude	Fault unit
$10\text{ s} < t < 30\text{ s}$	A_{xm}	bias	1	$[\text{m/s}^2]$
	A_{ym}	drift	$0.1t$	$[\text{m/s}^2]$
	A_{zm}	oscillatory	$2\pi\sin(0.5\pi t)$	$[\text{m/s}^2]$
	p_m	bias	$0.5\pi/180$	[rad/s]
	q_m	oscillatory	$\pi\sin(0.5\pi t)/180$	[rad/s]
	r_m	bias	$\pi/180$	[rad/s]
$40\text{ s} < t < 60\text{ s}$	A_{xm}	oscillatory	$-\pi\sin(0.5\pi t)/180$	$[\text{m/s}^2]$
	A_{ym}	drift	$-0.1t$	$[\text{m/s}^2]$
	A_{zm}	bias	-2	$[\text{m/s}^2]$
	p_m	bias	$-0.5\pi/180$	[rad/s]
	q_m	bias	$-0.5\pi/180$	[rad/s]
	r_m	oscillatory	$-\pi\sin(0.5\pi t)/180$	[rad/s]

and

$$\tilde{\mathbf{Q}}_{k-1} = \int_{t_{k-1}}^{t_k} \Phi_{k-1}\mathbf{G}(\bar{x}_{k-1|k-1})\mathbf{Q}\mathbf{G}^T(\bar{\boldsymbol{x}}_{k-1|k-1})\Phi_{k-1}^T d\tau$$
$$\mathbf{Q} = E\{\boldsymbol{w}(t)^T\boldsymbol{w}(\tau)\}, \qquad \mathbf{R} = E\{\boldsymbol{v}(t_i)^T\boldsymbol{v}(t_j)\} \tag{5.34}$$

The iteration part of the bias-free filter is as follows:

1. Kalman gain calculation:

$$\tilde{\mathbf{R}}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}\mathbf{H}_k^T + \mathbf{R} \tag{5.35}$$
$$\mathbf{K}_k^{\bar{x}} = \mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}\mathbf{H}_k^T\tilde{\mathbf{R}}_k^{-1} \tag{5.36}$$

2. Measurement update

$$\boldsymbol{\eta}_2 = \bar{\boldsymbol{x}}_{k|k-1} + \mathbf{K}_k^{\bar{\boldsymbol{x}}}\big(\boldsymbol{y}_k - \boldsymbol{h}(\boldsymbol{\eta}_1) - \mathbf{H}_k(\bar{\boldsymbol{x}}_{k|k-1} - \boldsymbol{\eta}_1)\big) \tag{5.37}$$

Define $\epsilon := \frac{||\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1||}{||\boldsymbol{\eta}_2||}$ and ϵ_0 the desired parameter to stop the iteration, if $\epsilon > \epsilon_0$, repeat step 1 and step 2. After each iteration, $\boldsymbol{\eta}_1 := \boldsymbol{\eta}_2$.

3. New time update

After the iteration, the time update is obtained as follows:

$$\bar{\boldsymbol{x}}_{k|k} = \boldsymbol{\eta}_2 \tag{5.38}$$
$$\mathbf{P}_{k|k}^{\bar{\boldsymbol{x}}} = (\mathbf{I} - \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{H}_k)\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}(\mathbf{I} - \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{H}_k)^T + \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{R}(\mathbf{K}_k^{\bar{\boldsymbol{x}}})^T \tag{5.39}$$

The bias filter of the IOTSEKF, which is used to estimate the faults, is as follows:

$$\boldsymbol{f}^i{}_{k|k-1} = \boldsymbol{f}^i{}_{k-1|k-1} \tag{5.40}$$

$$\boldsymbol{f}^i{}_{k|k} = \boldsymbol{f}^i{}_{k|k-1} + \mathbf{K}_k^{\boldsymbol{f}^i}(\boldsymbol{y}_k - \mathbf{H}_k\bar{x}_{k|k-1} - \mathbf{S}_k\boldsymbol{f}^i{}_{k|k-1}) \tag{5.41}$$

$$\mathbf{P}^{\boldsymbol{f}^i}_{k|k-1} = \mathbf{P}^{\boldsymbol{f}^i}_{k-1|k-1} + \mathbf{Q}^{\boldsymbol{f}^i}_{k-1} \tag{5.42}$$

$$\mathbf{K}_k^{\boldsymbol{f}^i} = \mathbf{P}^{\boldsymbol{f}^i}_{k|k-1}\mathbf{S}_k^T(\mathbf{H}_k\mathbf{P}^{\bar{x}}_{k|k-1}\mathbf{H}_k^T + \mathbf{R} + \mathbf{S}_k\mathbf{P}^{\boldsymbol{f}^i}_{k|k-1}\mathbf{S}_k^T)^{-1} \tag{5.43}$$

$$\mathbf{P}^{\boldsymbol{f}^i}_{k|k} = (\mathbf{I} - \mathbf{K}_k^{\boldsymbol{f}^i}\mathbf{S}_k)\mathbf{P}^{\boldsymbol{f}^i}_{k|k-1} \tag{5.44}$$

and

$$\bar{\boldsymbol{u}}_k = (\bar{\mathbf{U}}_{k+1} - \mathbf{U}_{k+1})\boldsymbol{f}^i{}_{k|k} \tag{5.45}$$

$$\bar{\mathbf{Q}}_k = \Gamma_k\mathbf{Q}\Gamma_k^T - \mathbf{Q}_k^{x\boldsymbol{f}^i}\bar{\mathbf{U}}_{k+1}^T - \mathbf{U}_{k+1}(\mathbf{Q}_k^{x\boldsymbol{f}^i} - \bar{\mathbf{U}}_{k+1}\mathbf{Q}_k^{\boldsymbol{f}^i})^T \tag{5.46}$$

$$\bar{\mathbf{U}}_k = \Phi_{k-1}\mathbf{V}_{k-1} + \Gamma_{k-1} \tag{5.47}$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{U}_k \tag{5.48}$$

where $\mathbf{U}_k$ and $\mathbf{V}_k$ are the two-stage blending matrices which are given by

$$\mathbf{U}_k = \bar{\mathbf{U}}_k + (\mathbf{Q}^{x\boldsymbol{f}^i}_{k-1} - \bar{\mathbf{U}}_k\mathbf{Q}^{\boldsymbol{f}^i}_{k-1})(\mathbf{P}^{\boldsymbol{f}^i}_{k|k-1})^{-1} \tag{5.49}$$

$$\mathbf{V}_k = \mathbf{U}_k - \mathbf{K}_k^{\bar{x}}\mathbf{S}_k \tag{5.50}$$

It can be noticed that Equations (5.35)-(5.39) are different from those of the OTSEKF [33]. The computational load of the IOTSEKF is more intensive than the OTSEKF due to the iteration of Equations (5.35)-(5.37). However, application of this iteration is only used in the first ten time steps [73]. Therefore, the increased computational load is limited. The advantage of the IOTSEKF and performance comparison of the IOTSEKF and the OTSEKF can be found in Lu and van Kampen [85].

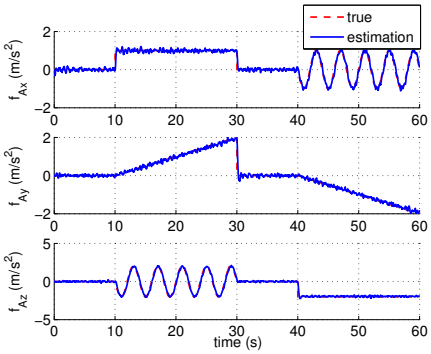
COMPUTATIONAL COMPLEXITY COMPARISON

The AEKF [134] can also estimate $\boldsymbol{f}^i$ by augmenting $\boldsymbol{f}^i$ as states. The state vector of the AEKF is

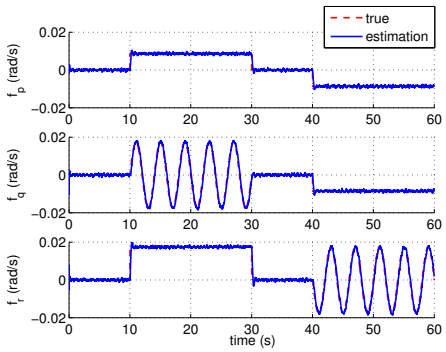
$$\boldsymbol{x}_a = [u^B_{\text{GS}}\ v^B_{\text{GS}}\ w^B_{\text{GS}}\ \phi\ \theta\ \psi\ \ f_{Ax}\ f_{Ay}\ f_{Az}\ f_p\ f_q\ f_r]^T \tag{5.51}$$

The dimension of the state vector using the AEKF is $(n+b)$. The dimension of the bias-free filter is n and that of the bias filter is b using the IOTSEKF.

Since the computational load of the KF is approximately related with the cube of the state dimension [129], the most intensive computationally complexity of the AEKF is $\mathcal{O}((n+b)^3)$ while that of the IOTSEKF is $\mathcal{O}(n^3+b^3)$. It is straightforward to see that the IOTSEKF is less computational complex than the AEKF. For the case in this chapter, $n=b$. The most intensive complexity of the AEKF is $\mathcal{O}(8n^3)$ while that of the IOTSEKF is $\mathcal{O}(2n^3)$.



(a) True and estimated f_{A_x}, f_{A_y} and f_{A_z} using KM2



(b) True and estimated f_p, f_q and f_r using KM2

Figure 5.1: IMU sensor FI of simulated aircraft model using KM2 in the absence of turbulence

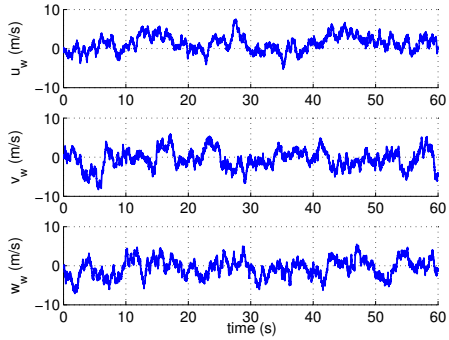


Figure 5.2: Speed of the turbulence in the north, east and down directions

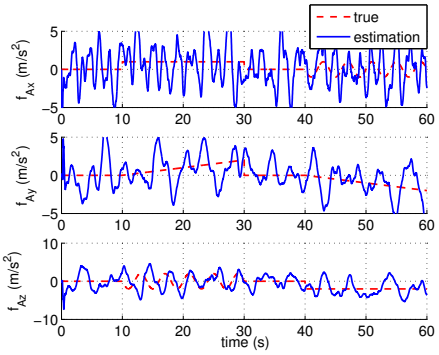
5.3.3. IMU SENSOR FI USING KM2 AND IOTSEKF IN THE ABSENCE OF TURBULENCE

In this subsection, the performance of using KM2 in the absence of turbulence is tested. The results of using KM2 are shown in Figure 5.1. It can be seen from the figure that the estimation of $\boldsymbol{f}^i$ is satisfactory. The fault estimation can follow the dynamics of the faults including the oscillatory faults. The results using KM1 are similar to those using KM2 in this case and are omitted here.

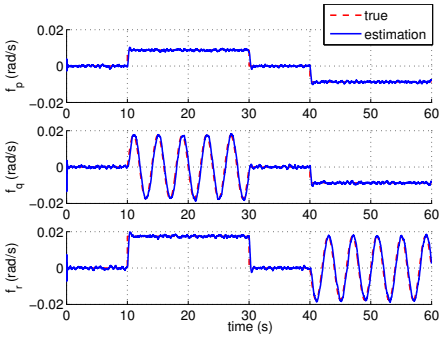
5.3.4. IMU SENSOR FI USING KM2 AND IOTSEKF IN THE PRESENCE OF TURBULENCE

In this subsection, the performance using KM2 is demonstrated in the presence of turbulence. A Dryden wind model is used to generate the turbulence whose speed is shown in Figure 5.2.

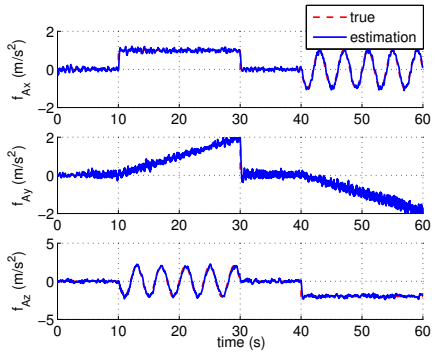
The results using KM1 and KM2 are shown in Figure 5.3. The estimation of f_{A_x}, f_{A_y}



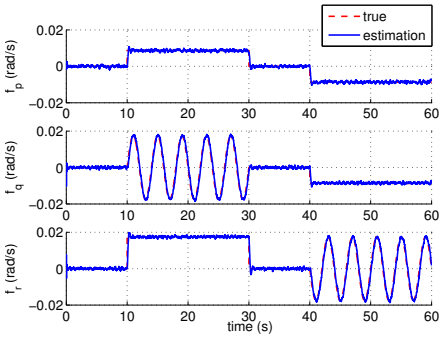
(a) True and estimated f_{A_x}, f_{A_y} and f_{A_z} using KM1



(b) True and estimated f_p, f_q and f_r using KM1



(c) True and estimated f_{A_x}, f_{A_y} and f_{A_z} using KM2



(d) True and estimated f_p, f_q and f_r using KM2

Figure 5.3: IMU sensor FI of simulated aircraft model using KM1 and KM2 in the presence of turbulence

and f_{A_z} using KM1, shown in Figure 5.3(a), is corrupted with turbulence whereas that of f_p, f_q and f_r using KM1 (shown in Figure 5.3(b)) is satisfactory. The reason is that in the presence of turbulence, Equation (5.27) is not equal to Equation (5.26) and is no longer correct. Consequently, the estimated f_{A_x}, f_{A_y} and f_{A_z} using Equation (5.27) are also biased.

The estimation of f_{A_x}, f_{A_y} and f_{A_z} using KM2 is shown in Figure 5.3(c). It can be seen that the performance is not influenced by the turbulence. The estimation of f_p, f_q and f_r using KM2 is shown in Figure 5.3(d), which is also comparable to the performance in the absence of turbulence shown in Fig 5.1(b). This demonstrates the superiority of using KM2 over KM1 for IMU sensor fault identification.

5.4. IMU SENSOR FI WITH APPLICATION TO REAL FLIGHT DATA

In the previous section, the performance of the IOTSEKF using KM1 and KM2 is tested and compared using simulated aircraft data. In this section, they are validated by making use of the recorded real flight data of the Cessna Citation II aircraft. It is a fixed-wing, business jet aircraft which is owned by Delft University of Technology and the Dutch Aerospace Laboratory.

The real flight data is taken from flight tests which were designed for the objective of aerodynamic model identification [91]. As stated in de Visser [91], sensor measurements contain biases. These biases can influence the FI performance of the IOTSEKF.

It should be noted that the IMU fault identification using KM1 requires the assumption of constant wind or no wind, which may not be satisfied in real flight, its performance may be degraded in the presence of changing wind. The IMU fault identification using KM2 does not assume this.

5.4.1. MEASUREMENTS OF THE REAL FLIGHT DATA

In reality, different sensors may have different update rates. Our GPS receiver for the ground speed measurements was manufactured before 2000 and its update rate is 1 Hz. It should be noted that GPS receivers with higher rates are available recently [133]. Also note that the GPS receiver is only required to measure the speed. The specific update rates of the on-board sensors are given in Table 5.3 [91].

Simultaneous faults are injected into the real flight data, which is the same as in Table 5.2.

5.4.2. REAL-LIFE MEASUREMENT MODEL

The sensors of the aircraft may contain biases and drifts. If KM1 is used, the measurements of α and β are required. In the flight test, α and β are measured by multiple vanes mounted on the nose boom of the aircraft. However, α and β measured by the vanes are not the true angle of attack and sideslip [90–92]. Since the vanes are not located in the center-of-gravity of the aircraft, the measurement model should incorporate the angular velocity induced flow at the vane location. Besides, the fuselage induced upwash and sidewash also have to be taken into account. The specific measurement model for α_m and β_m can be found in Lu et al [92].

However, if KM2 is used, the corresponding measurement model (Equations (5.16)-

Table 5.3: Specifications of the sensors using the real flight data

Variable	Unit	Std Deviation	Frequency	Sensor
V_m	[m/s]	n/a	100 Hz	Pitot-static probe
A_{xm}, A_{ym}, A_{zm}	[m/s^2]	$2{\cdot}10^{-2}$	100 Hz	Q-Flex 3100 accelerometer (IMU)
α_m, β_m	[rad]	$3.5{\cdot}10^{-3}$	100 Hz	vanes
ϕ_m, θ_m	[rad]	$8.7{\cdot}10^{-3}$	100 Hz	Sperry VG-14H vertical gyro
ψ_m	[rad]	n/a	10 Hz	gyrosyn compass
p_m, q_m, r_m	[rad/s]	$2{\cdot}10^{-3}$	100 Hz	LITEF μFORS rate gyro (IMU)
u_{GSm}, v_{GSm}, w_{GSm}	[m/s]	n/a	1 Hz	TANS III

(5.21)) remains since it does not use the air data information.

5

5.4.3. IMU SENSOR FI OF REAL FLIGHT DATA USING KM1

In this subsection, KM1 is used to identify $\boldsymbol{f}^i$ that are injected to the real flight data. The results are given in Figure 5.4. The estimation of the air speed components (u_a, v_a and w_a) and the Euler angles (ϕ, θ and ψ) are given in Figures 5.4(a) and 5.4(b), respectively. As can be seen, the state estimation is influenced by the faults. During 40 s $< t <$ 60 s, there are some oscillations in the estimation of u_a, v_a and ψ. However, the influence is small compared to the magnitude of the corresponding states. The estimation of f_{A_x}, f_{A_y} and f_{A_z} is shown in Figure 5.4(c). All the faults are estimated in the optimal minimum variance sense. The estimation can also follow the dynamics of the faults including the oscillatory faults. The estimation of f_p, f_q and f_r, shown in Figure 5.4(d), maintains satisfactory performance.

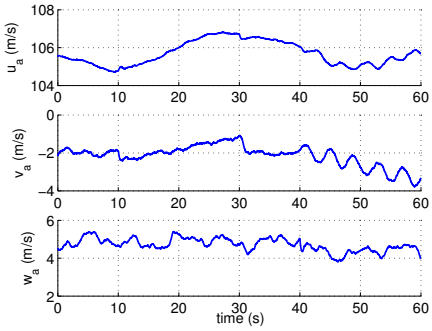
5.4.4. IMU SENSOR FI OF REAL FLIGHT DATA USING KM2

This subsection use KM2 to identify $\boldsymbol{f}^i$. The results are shown in Figure 5.5.

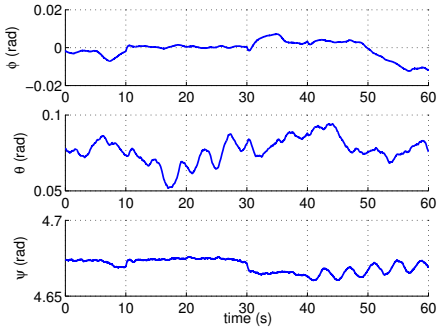
The estimation of the ground speed components (u^B_{GS}, v^B_{GS} and w^B_{GS}) and the Euler angles (ϕ, θ and ψ) are given in Figures 5.5(a) and 5.5(b), respectively. Some influence caused by the faults can be observed (e.g., there are some oscillations in the estimation of u^B_{GS} during 40 s $< t <$ 60 s), which is small compared to the magnitude of the faults and their corresponding states. The estimation of f_p, f_q and f_r, shown in Figure 5.5(d), is satisfactory, which is also comparable to the result using KM1. However, the estimation of f_{A_x}, f_{A_y} and f_{A_z}, shown in Figure 5.5(c), is degraded even though the estimation follows the dynamics of the faults. It is caused by the low sampling rate of the ground speed measurements (u_{GSm}, v_{GSm} and w_{GSm}).

5.4.5. COMPARISON AND DISCUSSION

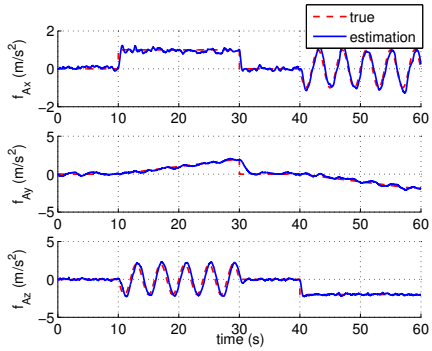
The fault estimation performance using KM1 and KM2 is compared using the RMSE of the fault estimation. The results are shown in Figure 5.6. The RMSE of the estimation of f_p, f_q and f_r using the two approaches, shown in Figure 5.6(b), are comparable. This



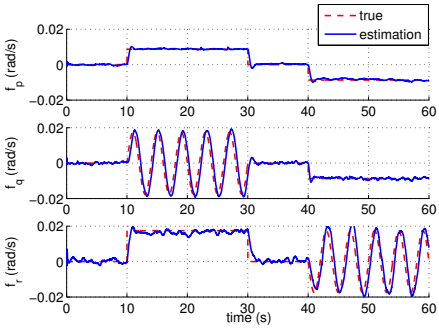
(a) Estimation of u_a, v_a and w_a



(b) Estimation of ϕ, θ and ψ

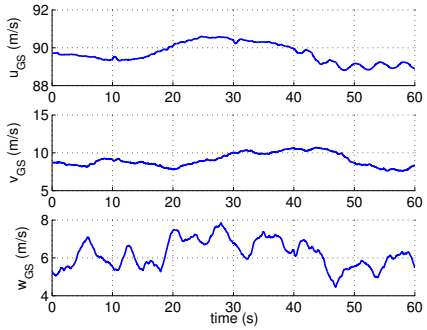


(c) True and estimated f_{A_x}, f_{A_y} and f_{A_z}

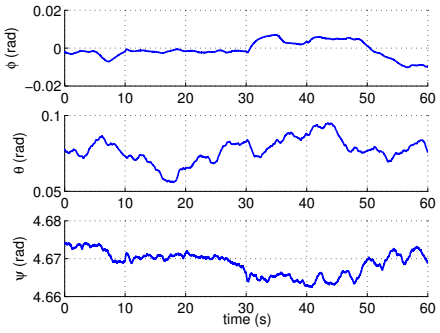


(d) True and estimated f_p, f_q and f_r

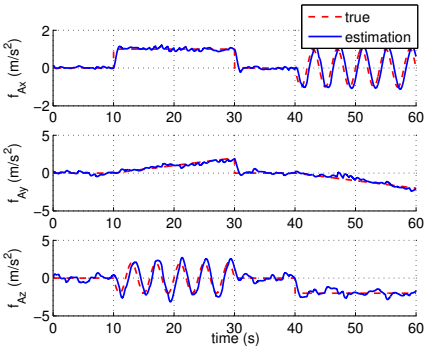
Figure 5.4: IMU sensor FI using the IOTSEKF and KM1 with application to real flight data



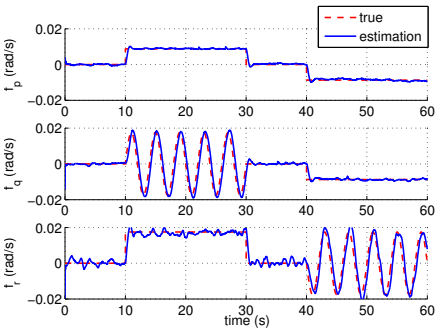
(a) Estimation of u^B_{GS}, v^B_{GS} and w^B_{GS}



(b) Estimation of ϕ, θ and ψ

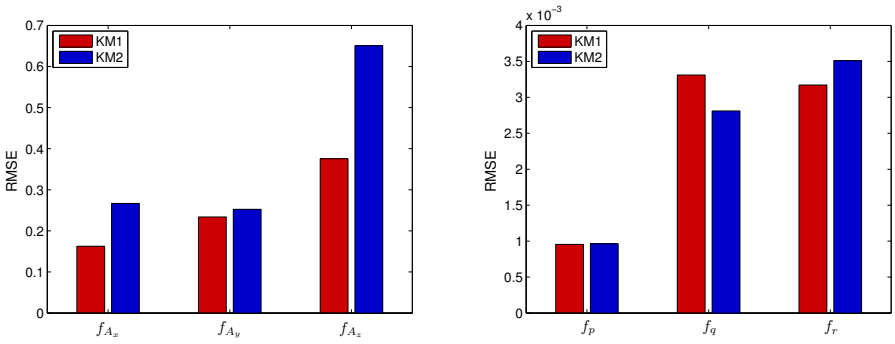


(c) True and estimated f_{A_x}, f_{A_y} and f_{A_z}



(d) True and estimated f_p, f_q and f_r

Figure 5.5: IMU sensor FI using the IOTSEKF and KM2 with application to real flight data



(a) RMSE of the estimation of f_{A_x}, f_{A_y} and f_{A_z}

(b) RMSE of the estimation of f_p, f_q and f_r

Figure 5.6: RMSE of the estimation of the IMU sensor faults using KM1 and KM2

is reasonable, since the measurements used for estimation of these faults using the two approaches are the same.

The RMSE of the estimation of f_{A_x}, f_{A_y} and f_{A_z} using KM2, shown in Figure 5.6(a), are slightly larger than those using KM1. However, recall that the sampling rate of the air data sensors used by KM1 is 100 Hz whereas that of the ground speed sensors used by KM2 is only 1 Hz. As can be seen from Figure 5.5(a), the vertical speed w_{GS}^B varies continually. Consequently, the estimation of w_{GS}^B will be influenced more by the low update rate. This is the reason why the RMSE of the estimation of f_{Az} using KM2 are approximately twice as much as those using KM1. On the contrary, v_{GS}^B does not change significantly, making the RMSE of the f_{Ay} using KM2 comparable to that using KM1.

The wind speed can be roughly estimated using the air speed and the ground speed in the body axis which are shown in Figures 5.4(a) and 5.5(a), respectively. The estimate of the wind speed in the body axis is shown in Figure 5.7.

As can be seen, the wind speed is time-varying. However, the variation of the wind speed is not significant. The wind speeds are approximately -16, 10, 1 m/s respectively. That explains why the performance of the IMU fault identification using KM1 is is still satisfactory. If the variation of the wind speeds is significant, e.g. when the aircraft is subjected to turbulence, the fault estimation performance using KM1 will be degraded.

Although the update rate of the air data sensors is 100 times that of the ground speed sensors, the performance of using KM1 and KM2 is comparable. Compared to the results from Section 5.3, it highlights that for real applications, GPS receivers with a higher update rate should be selected to guarantee the performance using KM2.

5.5. CONCLUSIONS

This chapter mainly addresses three problems. First, KM2 is proposed for the fault identification of the IMU sensor faults. It takes the wind into account, which makes it less sensitive to turbulence compared to KM1. Secondly, the fault identification using KM2 is tackled by proposing a novel IOTSEKF which improves the performance of the OTSEKF

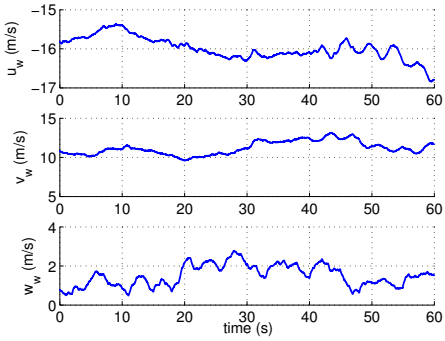


Figure 5.7: Estimation of the wind speed along the body axis

when dealing with nonlinear systems. Finally, the IMU sensor fault identification performance of the IOTSEKF using KM1 and KM2 is further validated by using the recorded real flight test data. Results demonstrate the effectiveness of the approaches. The proposed approach using KM1 and KM2 can be further extended to FTC systems, which can provide more accurate information to enhance the safety of the aircraft when there are malfunctions in the IMU sensors.

6

NONLINEAR AIRCRAFT SENSOR FAULT RECONSTRUCTION IN THE PRESENCE OF DISTURBANCES VALIDATED BY REAL FLIGHT DATA

In the previous chapter, an IMU FDD approach, which can deal with model uncertainties and external disturbances, was proposed. The filter used for state and fault estimation was the Iterated Optimal Two-Stage Extended Kalman Filter. The performance of this filter can be degraded due to a bad choice of the covariance matrices. Furthermore, although the approach was proposed to deal with turbulence. Its performance was not validated by real flight test data recorded during the presence of turbulence. To deal with those limitations, first an Adaptive Two-Stage Extended Kalman Filter, which can improve the performance of the Iterated Optimal Two-Stage Extended Kalman Filter, is proposed. Then, more flight tests were performed to collect aircraft response data in the presence of turbulence. The performance of the proposed approach is validated using the new recorded flight test data.

Parts of this chapter are based on:

P Lu, E van Kampen, C C de Visser, Q P Chu. Nonlinear Aircraft Sensor Fault Reconstruction In the Presence of Disturbances Validated by Real Flight data. *Control Engineering Practice*, (49), 112-128, 2016. [135].

This chapter proposes an approach for Inertial Measurement Unit sensor fault reconstruction by exploiting a ground speed-based kinematic model of the aircraft flying in a rotating earth reference system. Two strategies for the validation of sensor fault reconstruction are presented: closed-loop validation and open-loop validation. Both strategies use the same kinematic model and a newly-developed Adaptive Two-Stage Extended Kalman Filter to estimate the states and faults of the aircraft. Simulation results demonstrate the effectiveness of the proposed approach compared to an approach using an airspeed-based kinematic model. Furthermore, the major contribution is that the proposed approach is validated using real flight test data including the presence of external disturbances such as turbulence. Three flight scenarios are selected to test the performance of the proposed approach. It is shown that the proposed approach is robust to model uncertainties, unmodelled dynamics and disturbances such as time-varying wind and turbulence. Therefore, the proposed approach can be incorporated into aircraft Fault Detection and Isolation systems to enhance the performance of the aircraft.

6.1. INTRODUCTION

Sensor FDI is important to enhance the performance of the aircraft. For future aircraft, structural optimization is important [124]. Model-based sensor FDI can substantially decrease the weight of the aircraft, which in turn increases the performance of the aircraft and leads to fuel and noise reduction [124]. During the last few decades, many approaches are proposed to detect and estimate sensor faults by making use of the model information [12, 13, 66, 67]. Projects related to FDI in aerospace engineering, the REconfiguration of CONtrol in Flight for Integral Global Upset REcovery (RECONFIGURE) [136] and ADDSAFE project [124], also consider sensor faults. Recently, various model-based approaches have been proposed to deal with aircraft sensor faults. Marcos et al.[95] and Freeman et al. [25] use H_∞ to detect and isolate the sensor faults. Unknown input observers [16] are also applied to detect and isolate the sensor faults. Castaldi et al. [137, 138] propose a geometric approach for the FDI of aircraft sensors. SMOs [116, 139] are used to reconstruct the sensor faults. Control-based FDI method, which makes use of the control information, is also proposed [140]. Multiple-Model Adaptive Estimation approach is proposed in [83] to detect and estimate the aircraft sensor faults. For one thing, the performance of model-based approaches depends on the accuracy of the model used. Due to complexity, models of real-life systems are normally simplifications with limited accuracy. For aircraft dynamic models, exact aerodynamic forces and moments are difficult to obtain which leads to model uncertainties. For another, most systems are subject to external disturbances which will further affect the performance of model-based approaches. In civil aviation, wind and even time-varying wind is unavoidable during flight. One of the biggest challenges for aircraft FDI is the presence of disturbances such as time-varying wind and turbulence. Wind and turbulence can be present in various circumstances during the flight. As such, FDI approaches more robust to model uncertainties and disturbances are favored in aerospace engineering.

This chapter considers the fault reconstruction of the IMU sensors. The IMU contains accelerometers which measure the specific forces (A_x, A_y and A_z) and rate gyros which measure the angular rates (p, q and r) of the aircraft. The ADDSAFE project also considers faults in the yaw rate (r) sensor [114, 116]. Alwi and Edwards [116] use a Linear

Parameter Varying sliding mode scheme to reduce the influence of linearization errors. Van Eykeren and Chu [114] use an AS-KM of the aircraft which does not require linearization. Another advantage of using the AS-KM is that it does not require the calculation of the aerodynamic forces and moments. Since exact aerodynamic forces and moments are difficult to obtain, using the kinematic model reduces the effect of model uncertainties for FDI. Many approaches are proposed for the aircraft sensor FDI using this AS-KM, such as Double-Model Adaptive Estimation [92], Sliding Mode Differentiator [114] and Adaptive Three-Step Unscented Kalman Filter [65]. Van Eykeren and Chu [114] and Lu et al. [83] use the same KM for the fault estimation of the IMU sensors. The limitation of their approach is that the influence of disturbances such as turbulence is not considered. In Lu et al. [123], the turbulence influence is considered by a ground speed-based KM. The approach is validated by only simulation which demonstrates that the proposed FDI approach is less sensitive to turbulence.

Regarding the validation of aircraft sensor FDI approaches, the validation using real flight data is important. The performance of sensor FDI approaches is more reliable if they are validated by real flight tests or real flight test data. For real flights, sensors may not behave as in the simulations. More importantly, during real flight tests, various realistic wind and turbulence conditions can be experienced. Although in some circumstances the wind can be approximately constant, the wind is normally time varying. Moreover, turbulence can be encountered in various situations during the flight. These flight conditions can test the performance of the FDI approaches in the presence of external disturbances. Therefore, use of real flight test data containing different flight conditions, especially wind and turbulent conditions, is necessary to validate the performance of model-based FDI approaches.

Few researchers [97, 123, 131] validate the IMU sensor fault detectors using real flight data. Berdjag et al. [70, 97] detect oscillatory faults [39] of three inertial sensors. Their approach is validated on a real flight data set. Hu and Seiler [131] assess the false alarm probability of a model-based Unmanned Aerial Vehicle fault detection system. Lu et al. [123] also used real flight test data to validate the performance of model-based fault reconstruction for the IMU sensors. However, there are two limitations in [123]. Firstly, it assumes a flat and non-rotating earth which may introduce errors when long-range flights are considered. Furthermore, the real flight test in [123] contains no turbulence since the objective of the flight test was data acquisition for aerodynamic model identification [123, 141]. Therefore, the performance of the FDI approach in [123], in the presence of various disturbances such as time-varying wind and turbulence, is not validated.

In the present chapter, an IMU sensor fault reconstruction approach using a general GS-KM is proposed. This KM, which is a model of the aircraft flying over a spherical and rotating earth, is proposed as the model for the IMU sensor fault reconstruction. Two validation strategies (open-loop and closed-loop validations) are implemented and compared for the IMU sensor fault reconstruction. Another contribution of this chapter is that a new ATSEKF is proposed to reconstruct the states and faults. This ATSEKF is an improved version of the IOTSEKF [123] with covariance matrix adaption. The performance of the ATSEKF for the sensor fault reconstruction is validated by simulation. The fault scenarios include not only biases, but also drift and oscillatory faults. All the IMU

sensor faults are estimated in an unbiased sense. The performance using the GS-KM is compared with that using the AS-KM. The performance of the ATSEKF is also compared with existing approaches such as the SMO.

The major contribution of this chapter is that the performance of the proposed IMU sensor fault reconstruction approach is validated using real flight test data in the presence of disturbances such as time-varying wind and turbulence. Several flight tests are performed in order to gather data in different flight conditions. Zero-g flight is also performed in order to test the performance of the approach over a wide range of the flight envelope. Flight data in turbulent conditions is also collected which will test the performance in the presence of strong external disturbances. The validation is performed using three different flight conditions: level flight, zero-g and turbulence. The performance of using the GS-KM is compared to the approach using the AS-KM in order to demonstrate the superiority of the proposed approach. Furthermore, the results sufficiently demonstrate that the proposed approach is suitable for IMU sensor fault reconstruction in practice.

The structure of this chapter is as follows: Section 6.2 proposes an approach which uses a new KM for the IMU sensor fault reconstruction. Section 6.3 introduces the newly-developed ATSEKF for sensor fault reconstruction. Besides, two validation strategies are proposed which use the ATSEKF and the proposed KM. The performance of the proposed approach is demonstrated using simulated data. In Section 6.4, the three flight scenarios are introduced. Section 6.5 presents the real flight data validation results using the proposed approach in various flight and wind conditions. Section 6.6 concludes the chapter.

6.2. NOVEL APPROACH FOR IMU SENSOR FAULT RECONSTRUCTION

This section proposes an approach for IMU fault reconstruction based on a GS-KM. For comparison, the previously used KM, the AS-KM [82, 85], is also presented.

6.2.1. AIRCRAFT KMS INCLUDING IMU SENSOR FAULTS

The proposed GS-KM is a KM of the aircraft flying in a rotating earth reference frame [90]. It should be noted that this model has not been used for aircraft fault reconstruction.

The GS-KM including IMU sensor faults can be described as follows:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), \boldsymbol{f}^i(t), t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) \tag{6.1}$$

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t)) \tag{6.2}$$

$$\boldsymbol{y}_m(t) = \boldsymbol{y}(t) + \boldsymbol{v}(t), \quad t = t_i, \ i = 1, 2, \ldots \tag{6.3}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the system states, $\boldsymbol{u}_m \in \mathbb{R}^b$ the measured input, $\boldsymbol{y}_m \in \mathbb{R}^m$ the measured output. The subscript 'm' indicates that the corresponding variable is measured. $\mathbf{G} \in \mathbb{R}^{n \times b}$ is the noise distribution matrix. The function $\boldsymbol{f}^i \in \mathbb{R}^b$ represents the

input faults. The system equation variables are defined as follows:

$$\boldsymbol{x} = [U_N\ U_E\ U_D\ \phi\ \theta\ \psi]^T \tag{6.4}$$

$$\boldsymbol{u}_m = [A_{xm}\ A_{ym}\ A_{zm}\ p_m\ q_m\ r_m]^T = \boldsymbol{u}^0 + \boldsymbol{w} + \boldsymbol{f}^i \tag{6.5}$$

$$\boldsymbol{u}^0 = [A_x\ A_y\ A_z\ p\ q\ r\ ]^T \tag{6.6}$$

$$\boldsymbol{w} = [w_{Ax}\ w_{Ay}\ w_{Az}\ w_p\ w_q\ w_r\ ]^T \tag{6.7}$$

$$\boldsymbol{f}^i = [f_{Ax}\ f_{Ay}\ f_{Az}\ f_p\ f_q\ f_r]^T \tag{6.8}$$

$$\boldsymbol{y}_m = [U_{Nm}\ U_{Em}\ U_{Dm}\ \phi_m\ \theta_m\ \psi_m]^T \tag{6.9}$$

$$\boldsymbol{v} = [v_{U_N}\ v_{U_E}\ v_{U_D}\ v_\phi\ v_\theta\ v_\psi]^T \tag{6.10}$$

where the U_N, U_E and U_D are the ground speed velocity components in the local navigation frame. A_x, A_y and A_z are the specific forces, ϕ, θ and ψ are the Euler angles and p, q, r are the rotational rates. It should be noticed that $\boldsymbol{u}_m$ is the measurement of the IMU which contains faults $\boldsymbol{f}^i$. f_{Ax}, f_{Ay} and f_{Az} represent the faults in the accelerometers while f_p, f_q and f_r represent the faults in the rate gyros. $\boldsymbol{w}$ is the process noise sequence and $\boldsymbol{v}$ is the noise sequence in the output measurements denoted by $\boldsymbol{y}_m$.

For the GS-KM proposed in this chapter, $\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), \boldsymbol{f}^i(t), t)$ is given as follows:

$$\begin{cases}
(A_{xm} - f_{Ax})\mathrm{c}\theta\mathrm{c}\psi + (A_{ym} - f_{Ay})(\mathrm{s}\phi\mathrm{s}\theta\mathrm{c}\psi - \mathrm{c}\phi\mathrm{s}\psi) + (A_{zm} - f_{Az})(\mathrm{c}\phi\mathrm{s}\theta\mathrm{c}\psi + \mathrm{s}\phi\mathrm{s}\psi) \\
\quad + \dfrac{U_N U_D}{R_e + h} - \dfrac{U_E^2 \mathrm{t}\delta}{R_e + h} - 2\Omega U_E \mathrm{s}\delta \\
(A_{xm} - f_{Ax})\mathrm{c}\theta\mathrm{s}\psi + (A_{ym} - f_{Ay})(\mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi + \mathrm{c}\phi\mathrm{c}\psi) + (A_{zm} - f_{Az})(\mathrm{c}\phi\mathrm{s}\theta\mathrm{s}\psi - \mathrm{s}\phi\mathrm{c}\psi) \\
\quad + \dfrac{U_N U_E \mathrm{t}\delta}{R_e + h} + \dfrac{U_E U_D}{R_e + h} + 2\Omega(U_N \mathrm{s}\delta + U_D \mathrm{c}\delta) \\
-(A_{xm} - f_{Ax})\mathrm{s}\theta + (A_{ym} - f_{Ay})\mathrm{s}\phi\mathrm{c}\theta + (A_{zm} - f_{Az})\mathrm{c}\phi\mathrm{c}\theta \\
\quad - \dfrac{U_N^2}{R_e + h} - \dfrac{U_E^2}{R_e + h} - 2\Omega U_E \mathrm{c}\delta + g \\
(p_m - f_p) + (q_m - f_q)\mathrm{s}\phi\mathrm{t}\theta + (r_m - f_r)\mathrm{c}\phi\mathrm{t}\theta - (\dfrac{U_E}{R_e + h} + \Omega\mathrm{c}\delta)\dfrac{\mathrm{c}\psi}{\mathrm{c}\theta} + \dfrac{U_N \mathrm{s}\psi}{(R_e + h)\mathrm{c}\theta} \\
(q_m - f_q)\mathrm{c}\phi - (r_m - f_r)\mathrm{s}\phi + (\dfrac{U_E}{R_e + h} + \Omega\mathrm{c}\delta)\mathrm{s}\psi + \dfrac{U_N \mathrm{c}\psi}{R_e + h} \\
(q_m - f_q)\dfrac{\mathrm{s}\phi}{\mathrm{c}\theta} + (r_m - f_r)\dfrac{\mathrm{c}\phi}{\mathrm{c}\theta} + (\dfrac{U_E}{R_e + h} + \Omega\mathrm{c}\delta)\mathrm{t}\theta\mathrm{c}\psi + \dfrac{U_N \mathrm{t}\theta\mathrm{s}\psi}{R_e + h} + \dfrac{U_E \mathrm{t}\delta}{R_e + h} + \Omega\mathrm{s}\delta
\end{cases} \tag{6.11}$$

where s(•), c(•) and t(•) denote the trigonometric functions sin(•), cos(•) and tan(•) respectively. $R_e = 6367$ km, $\Omega = 7.2921 \times 10^{-5}$ rad/s, δ is the latitude and h is the altitude. The gravity acceleration g is calculated as follow [90]:

$$g = 9.780318(\frac{R_e}{R_e + h})^2(1 + 5.3024 \times 10^{-3}\mathrm{s}^2\delta - 5.9 \times 10^{-6}\mathrm{s}^2 2\delta) \tag{6.12}$$

In Equation (6.1), $\mathbf{G}(\boldsymbol{x}(t))$ is defined as follows:

$$\mathbf{G}(\boldsymbol{x}(t)) = \begin{bmatrix} -c\theta c\psi & -(s\phi s\theta c\psi - c\phi s\psi) & -(c\phi s\theta c\psi + s\phi s\psi) & 0 & 0 & 0 \\ -c\theta s\psi & -(s\phi s\theta s\psi + c\phi c\psi) & -(c\phi s\theta s\psi - s\phi c\psi) & 0 & 0 & 0 \\ s\theta & -s\phi c\theta & -c\phi c\theta & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -s\phi t\theta & -c\phi t\theta \\ 0 & 0 & 0 & 0 & -c\phi & s\phi \\ 0 & 0 & 0 & 0 & -s\phi/c\theta & -c\phi/c\theta \end{bmatrix} \tag{6.13}$$

For the GS-KM, $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{x}$. Therefore, the measurement model of the GS-KM reduces to

$$\boldsymbol{y}_m(t) = \boldsymbol{x}(t) + \boldsymbol{v}(t) \tag{6.14}$$

This concludes the definition of the GS-KM (given by Equations (6.11)-(6.14)). To demonstrate the performance of the proposed approach, the AS-KM, used in [82, 83, 85] and given by Equations (6.18) and (6.20), is also given for comparison. For the sake of explanation, the process model of the AS-KM is briefly shown below. The system variables of the AS-KM are defined as follows [123]:

$$\boldsymbol{x}' = [u_a \; v_a \; w_a \; \phi \; \theta \; \psi]^T \tag{6.15}$$

$$\boldsymbol{y}'_m = [V_m \; \alpha_m \; \beta_m \; \phi_m \; \theta_m \; \psi_m]^T \tag{6.16}$$

$$\boldsymbol{v}' = [v_V \; v_\alpha \; v_\beta \; v_\phi \; v_\theta \; v_\psi]^T \tag{6.17}$$

where u_a, v_a, w_a are the airspeed components expressed in the body-fixed reference frame. V_m, α_m and β_m are the measurements of the true airspeed, angle of attack and angle of sideslip. Other variables, such as $\boldsymbol{u}_m$ and $\boldsymbol{u}^0$, are the same as for the GS-KM.

The AS-KM is given as follows [85]:

$$\begin{cases} \dot{u}_a = (A_{xm} - f_{Ax} - w_{Ax}) + v_a(r_m - f_r - w_r) - w_a(q_m - f_q - w_q) - g\sin\theta \\ \dot{v}_a = (A_{ym} - f_{Ay} - w_{Ay}) - u_a(r_m - f_r - w_r) + w_a(p_m - f_p - w_p) + g\cos\theta\sin\phi \\ \dot{w}_a = (A_{zm} - f_{Az} - w_{Az}) + u_a(q_m - f_q - w_q) - v_a(p_m - f_p - w_p) + g\cos\theta\cos\phi \\ \dot{\phi} = (p_m - f_p - w_p) + (q_m - f_q - w_q)\sin\phi\tan\theta + (r_m - f_r - w_r)\cos\phi\tan\theta \\ \dot{\theta} = (q_m - f_q - w_q)\cos\phi - (r_m - f_r - w_r)\sin\phi \\ \dot{\psi} = (q_m - f_q - w_q)\dfrac{\sin\phi}{\cos\theta} + (r_m - f_r - w_r)\dfrac{\cos\phi}{\cos\theta} \end{cases} \tag{6.18}$$

By extracting the noise $\boldsymbol{w}$ from the above equations, one can readily obtain $\boldsymbol{f}(\boldsymbol{x}'(t), \boldsymbol{u}_m(t), \boldsymbol{f}^i(t), t)$ and $\mathbf{G}(\boldsymbol{x}'(t))$ and write the equations in the form of Equations(6.1)-

(6.3). The matrix $\mathbf{G}(\boldsymbol{x}'(t))$ is defined as follows:

$$\mathbf{G}(\boldsymbol{x}'(t)) = \begin{bmatrix} -1 & 0 & 0 & 0 & w_a & -v_a \\ 0 & -1 & 0 & -w_a & 0 & u_a \\ 0 & 0 & -1 & v_a & -u_a & 0 \\ 0 & 0 & 0 & -1 & -\sin\phi\tan\theta & -\cos\phi\tan\theta \\ 0 & 0 & 0 & 0 & -\cos\phi & \sin\phi \\ 0 & 0 & 0 & 0 & -\sin\phi/\cos\theta & -\cos\phi/\cos\theta \end{bmatrix} \tag{6.19}$$

The measurement model is:

$$\boldsymbol{y}'(t) = \boldsymbol{h}(\boldsymbol{x}'(t)) + \boldsymbol{v}'(t), \qquad t = t_i, i = 1,2,... \tag{6.20}$$

where $\boldsymbol{h}(\boldsymbol{x}'(t))$ is the nonlinear output function which can be found in [85].

Remarks: The model used in [123] is a simplified model of the GS-KM used in the present chapter. The model used in [123] assumes that the earth is flat and non-rotating, which could introduce errors for long-range flights [90]. In contrast, the GS-KM used in the present chapter does not have to assume that the earth is flat and non-rotating.

The advantage of the GS-KM compared to the AS-KM is that the GS-KM can deal with various flight situations and is more robust in the presence of disturbances such as time-varying wind and turbulence, which will be demonstrated in the following sections.

For the GS-KM, the ground speed measurements are assumed to be fault-free while for the AS-KM, the air data measurements are assumed to be fault-free.

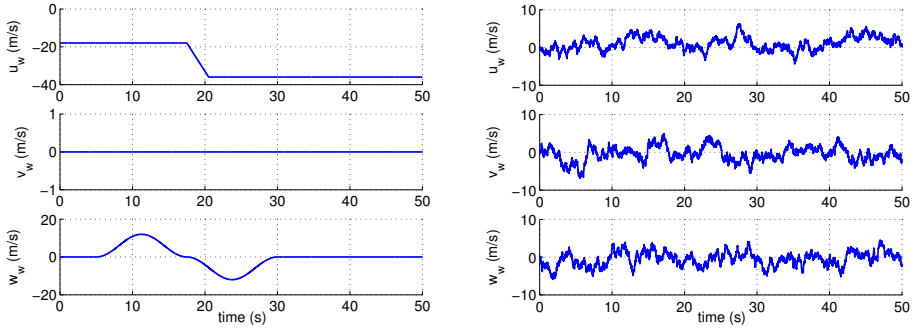
6.2.2. DISTURBANCES AND THEIR INFLUENCE ON FAULT RECONSTRUCTION

This chapter deals with IMU sensor fault reconstruction in the presence of external disturbances. Section 6.3 considers simulated disturbances while Section 6.5 deals with real-life disturbances. In this section, the simulated disturbances are presented first. Specifically, two main types of external disturbances are presented: wind shear and turbulence. The speed components of the disturbances in the earth axis and the body axis are denoted by u_w, v_w, w_w and u_w^B, v_w^B, w_w^B respectively. Then, the influence of disturbances on fault reconstruction is analyzed.

WIND SHEAR AND TURBULENCE

Wind shear is a rapid change in the wind vector which could be hazardous to aircraft especially when the aircraft is flying at a low altitude [64]. The wind shear model used in this chapter is derived from [64] and the wind speed components in the earth frame u_w, v_w and w_w are shown in Figure 6.1(a). v_w is assumed to be zero.

Turbulence is a random process which can occur in various flight situations such as in the clouds as well as near mountains [64]. In this chapter, the simulated turbulence is generated using the Dryden model. The power spectral density functions of the Dryden



(a) Wind speed components of the simulated wind shear

(b) Wind speed components of the simulated turbulence using the Dryden Model

Figure 6.1: Two main types of external disturbances: wind shear and turbulence

turbulence model are given as follows [64]:

$$\Phi_u(\omega_o) = \frac{2\sigma_u^2 L_u}{\pi V} \frac{1}{1+(L_u\omega_o/V)^2} \tag{6.21}$$

$$\Phi_v(\omega_o) = \frac{\sigma_v^2 L_v}{\pi V} \frac{1+3(L_v\omega_o/V)^2}{1+(L_v\omega_o/V)^2} \tag{6.22}$$

$$\Phi_w(\omega_o) = \frac{\sigma_w^2 L_w}{\pi V} \frac{1+3(L_w\omega_o/V)^2}{1+(L_w\omega_o/V)^2} \tag{6.23}$$

where V is the true airspeed, ω_o is the observed angular frequency. In this chapter, the scale lengths $L_u = L_v = L_w = 530$ m, the intensities $\sigma_u = \sigma_v = \sigma_w = 3.54$ m/s. The resulting speed components of the generated turbulence are shown in Figure 6.1(b).

INFLUENCE OF DISTURBANCES ON FAULT RECONSTRUCTION

The influence of the disturbances on fault reconstruction is explained below. It should be noted that the disturbances are included in the modeling of the aircraft response but they are considered unknown when designing fault reconstruction approaches.

Define the following:

$$\boldsymbol{V}_a^B = [u_a\ v_a\ w_a]^T,\ \boldsymbol{V}_w^B = [u_w^B\ v_w^B\ w_w^B]^T \tag{6.24}$$

$$\boldsymbol{A} = [A_x\ A_y\ A_z]^T,\ \boldsymbol{\omega} = [p\ q\ r]^T \tag{6.25}$$

$$\boldsymbol{f}_A = [f_{Ax}\ f_{Ay}\ f_{Az}]^T,\ \boldsymbol{f}_\omega = [f_p\ f_q\ f_r]^T \tag{6.26}$$

$$\boldsymbol{w}_A = [w_{Ax}\ w_{Ay}\ w_{Az}]^T,\ \boldsymbol{w}_\omega = [w_p\ w_q\ w_r]^T \tag{6.27}$$

According to [123], the dynamics of the airspeed can be expressed as

$$\dot{\boldsymbol{V}}_a^B = \boldsymbol{A} + \mathbf{T}_{be}\boldsymbol{g} - \boldsymbol{\omega} \times \boldsymbol{V}_a^B - (\dot{\boldsymbol{V}}_w^B + \boldsymbol{\omega} \times \boldsymbol{V}_w^B) \tag{6.28}$$

where $\mathbf{T}_{be}$ is the transformation matrix from earth axis to body axis, which can be found in Appendix A. $\boldsymbol{g} = [0\ 0\ g]^T$. Rewrite Equation (6.5) and substitute it into Equation (6.28), resulting in:

$$\dot{\boldsymbol{V}}_a^B = (\boldsymbol{A}_m - \boldsymbol{f}_A - \boldsymbol{w}_A) + \mathbf{T}_{be}\boldsymbol{g} - (\boldsymbol{\omega}_m - \boldsymbol{f}_\omega - \boldsymbol{w}_\omega) \times \boldsymbol{V}_a^B - (\dot{\boldsymbol{V}}_w^B + (\boldsymbol{\omega}_m - \boldsymbol{f}_\omega - \boldsymbol{w}_\omega) \times \boldsymbol{V}_w^B) \tag{6.29}$$

Define $\boldsymbol{d}_w = \dot{\boldsymbol{V}}_w^B + (\boldsymbol{\omega}_m - \boldsymbol{f}_\omega - \boldsymbol{w}_\omega) \times \boldsymbol{V}_w^B$. When the wind is constant, $\boldsymbol{d}_w = \mathbf{0}$, the above equation reduces to

$$\dot{\boldsymbol{V}}_a^B = (\boldsymbol{A}_m - \boldsymbol{f}_A - \boldsymbol{w}_\omega) + \mathbf{T}_{be}\boldsymbol{g} - (\boldsymbol{\omega}_m - \boldsymbol{f}_\omega - \boldsymbol{w}_\omega) \times \boldsymbol{V}_a^B \tag{6.30}$$

which is the vector form of the first three equations of Equation (6.18). When the wind is not constant, $\boldsymbol{d}_w \neq \mathbf{0}$. It is seen from Equation (6.29) that $\boldsymbol{f}_A$ and $\boldsymbol{d}_w$ have the same distribution matrix ($\mathbf{I}$) and it is difficult to distinguish between them. Consequently, when the AS-KM is used to reconstruct the faults in the presence of external disturbances, the faults and the disturbances are coupled, which could lead to incorrect fault reconstruction.

6.2.3. MEASUREMENTS AND SENSOR FAULTS

The aircraft model used in the simulation is the ADMIRE benchmark model [142]. To make the validation more realistic, noise is added to the measurements of the model. This simulation data is used in Section 6.3 to validate the performance of the proposed approaches while the real flight test data is used in Section 6.5. The standard deviations of the noises added to the sensors are given in Table 6.1. In the table, $\sigma_{(\bullet)}$ denotes the standard deviations of the noise in the corresponding sensors.

Table 6.1: Standard deviations of the noises added in the sensors

Standard deviations	Magnitudes	Units
σ_V	0.1	[m/s]
$\sigma_{U_N}, \sigma_{U_E}, \sigma_{U_D}$	0.01	[m/s]
$\sigma_\alpha, \sigma_\beta$	$0.1\pi/180$	[rad]
$\sigma_\phi, \sigma_\theta, \sigma_\psi$	$0.01\pi/180$	[rad]
$\sigma_{A_x}, \sigma_{A_y}, \sigma_{A_z}$	0.01	[m/s^2]
$\sigma_p, \sigma_q, \sigma_r$	$0.01\pi/180$	[rad/s]

In this chapter, all the IMU sensors fail simultaneously during 10 s $< t <$ 30 s. This means $\boldsymbol{f}^i = \mathbf{0}$ when $t \leq 10$ s or $t \geq 30$ s. During 10 s $< t <$ 30 s, the faults $\boldsymbol{f}^i$ are presented in Table 6.2. As can be seen from the table, different types of faults are considered, including bias faults and drift faults, as well as oscillatory faults. The fault scenario is arbitrarily chosen to test the performance of the proposed approach.

Table 6.2: Fault information

Time interval	Faults	Fault type	Fault magnitude	Fault unit
$10\text{ s} < t < 30\text{ s}$	f_{Ax}	bias	3	$[\text{m/s}^2]$
	f_{Ay}	drift	$0.1(t-10)$	$[\text{m/s}^2]$
	f_{Az}	oscillatory	$4\sin(0.5\pi t)$	$[\text{m/s}^2]$
	f_p	bias	$1.5\pi/180$	[rad/s]
	f_q	oscillatory	$\pi\sin\big(0.5\pi(t-10)\big)/180$	[rad/s]
	f_r	bias	$3\pi/180$	[rad/s]

6.3. IMU SENSOR FAULT RECONSTRUCTION

This section deals with the IMU sensor fault reconstruction problem. First, the newly-developed ATSEKF is introduced in Section 6.3.1. Then, in Section 6.3.2, two different strategies are proposed to validate the proposed approach for IMU sensor fault reconstruction. The validation results of the proposed approach in the absence of disturbances are shown in Section 6.3.3. The validation results in the presence of wind shear and turbulence are given in Sections 6.3.4 and 6.3.5 respectively.

6.3.1. ADAPTIVE TWO-STAGE EXTENDED KALMAN FILTER

In this chapter, the ATSEKF is proposed to estimate the states and the faults. The ATSEKF is an improved version of the IOTSEKF [123]. First, the IOTSEKF is introduced. Then, the adaption scheme for the covariance matrix is introduced.

The IOTSEKF is composed of a modified bias-free filter and a bias filter [33, 123]. The bias filter, which estimates the faults, models the faults as: $\boldsymbol{f}_k^i = \boldsymbol{f}_{k-1}^i + \boldsymbol{w}_{k-1}^{\boldsymbol{f}^i}$ with covariances: $E\{\boldsymbol{w}_k^{\boldsymbol{f}^i}(\boldsymbol{w}_l^{\boldsymbol{f}^i})^T\} = \mathbf{Q}_k^{\boldsymbol{f}^i}\delta_{kl}$ and $E\{\boldsymbol{w}_k(\boldsymbol{w}_l^{\boldsymbol{f}^i})^T\} = \mathbf{Q}_k^{\boldsymbol{x}\boldsymbol{f}^i}\delta_{kl}$. $\boldsymbol{w}_k$ is the defined in Equation (6.1) and δ_{kl} the Kronecker function. The IOTSEKF, slightly modified from [123] for clarity, is presented below.

1. The bias-free filter of the IOTSEKF is as follows:

   (a) Predicted state and error covariance matrix:

$$\boldsymbol{\eta}_1 = \bar{\boldsymbol{x}}_{k|k-1} = \bar{\boldsymbol{x}}_{k-1|k-1} + \int_{t_{k-1}}^{t_k} \boldsymbol{f}(\bar{\boldsymbol{x}}(t), \boldsymbol{u}_m(t), \mathbf{0}, t)\,dt + \bar{\boldsymbol{u}}_{k-1} \tag{6.31}$$

$$\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}} = \boldsymbol{\Phi}_{k-1}\mathbf{P}_{k-1|k-1}^{\bar{\boldsymbol{x}}}\boldsymbol{\Phi}_{k-1}^T + \bar{\mathbf{Q}}_{k-1} \tag{6.32}$$

   where $\boldsymbol{f}(\bar{\boldsymbol{x}}(t), \boldsymbol{u}_m(t), \mathbf{0}, t)$ means the fault vector is set to $\mathbf{0}$. This is because the IOTSEKF decouples the state and fault estimation. $\bar{\boldsymbol{u}}_{k-1}$ and $\bar{\mathbf{Q}}_{k-1}$ are defined by Equations (6.45) and (6.46) respectively. Discrete-time matrices $\boldsymbol{\Phi}_{k-1}$ and

$\mathbf{\Gamma}_{k-1}$ are calculated by:

$$\mathbf{\Phi}_{k-1} = e^{\mathbf{F}_{k-1}\Delta t} = \sum_{n}^{\infty} \frac{\mathbf{F}_{k-1}^{n}(\Delta t)^{n}}{n!}, \ \Delta t = t_k - t_{k-1}, \tag{6.33}$$

$$\mathbf{F}_{k-1} = \frac{\partial \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{f}^{i}(t))}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_{k-1|k-1}}, \ \mathbf{\Gamma}_{k-1} = \int_{t_{k-1}}^{t_k} \mathbf{\Phi}_{k-1}\mathbf{G}(\bar{\boldsymbol{x}}_{k-1|k-1})dt \tag{6.34}$$

(b) Kalman gain calculation:

$$\mathbf{H}_k = \frac{\partial \boldsymbol{h}(\boldsymbol{x}(t))}{\partial \boldsymbol{x}}|_{\boldsymbol{x}=\boldsymbol{\eta}_1} \tag{6.35}$$

$$\mathbf{K}_k^{\bar{\boldsymbol{x}}} = \mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}\mathbf{H}_k^{T}(\mathbf{H}_k\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}\mathbf{H}_k^{T} + \mathbf{R}_k)^{-1} \tag{6.36}$$

where $\mathbf{R}_k$ is the covariance matrix defined by $E\{\boldsymbol{v}_k\boldsymbol{v}_l^{T}\} = \mathbf{R}_k\delta_{kl}$.

(c) Measurement update:

$$\boldsymbol{\eta}_2 = \bar{\boldsymbol{x}}_{k|k-1} + \mathbf{K}_k^{\bar{\boldsymbol{x}}}\big(\boldsymbol{y}_k - \boldsymbol{h}(\boldsymbol{\eta}_1) - \mathbf{H}_k(\bar{\boldsymbol{x}}_{k|k-1} - \boldsymbol{\eta}_1)\big) \tag{6.37}$$

Define $\epsilon := \frac{\|\boldsymbol{\eta}_2 - \boldsymbol{\eta}_1\|}{\|\boldsymbol{\eta}_2\|}$ and ϵ_0 the desired parameter to stop the iteration. If $\epsilon > \epsilon_0$, repeat step (b) and step (c). In this chapter, ϵ_0 is chosen to be 10^{-8} and this iteration is only performed in the first ten time steps. After each iteration, $\boldsymbol{\eta}_1 := \boldsymbol{\eta}_2$.

(d) Update the state estimation error covariance matrix, if $\epsilon \leq \epsilon_0$:

$$\bar{\boldsymbol{x}}_{k|k} = \boldsymbol{\eta}_2 \tag{6.38}$$

$$\mathbf{P}_{k|k}^{\bar{\boldsymbol{x}}} = (\mathbf{I} - \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{H}_k)\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}(\mathbf{I} - \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{H}_k)^{T} + \mathbf{K}_k^{\bar{\boldsymbol{x}}}\mathbf{R}_k(\mathbf{K}_k^{\bar{\boldsymbol{x}}})^{T} \tag{6.39}$$

2. The bias filter of the IOTSEKF, which is used to estimate the faults, is as follows:

$$\hat{\boldsymbol{f}}^{i}{}_{k|k-1} = \hat{\boldsymbol{f}}^{i}{}_{k-1|k-1} \tag{6.40}$$

$$\mathbf{P}_{k|k-1}^{\boldsymbol{f}^i} = \mathbf{P}_{k-1|k-1}^{\boldsymbol{f}^i} + \mathbf{Q}_{k-1}^{\boldsymbol{f}^i} \tag{6.41}$$

$$\mathbf{K}_k^{\boldsymbol{f}^i} = \mathbf{P}_{k|k-1}^{\boldsymbol{f}^i}\mathbf{S}_k^{T}(\mathbf{H}_k\mathbf{P}_{k|k-1}^{\bar{\boldsymbol{x}}}\mathbf{H}_k^{T} + \mathbf{R}_k + \mathbf{S}_k\mathbf{P}_{k|k-1}^{\boldsymbol{f}^i}\mathbf{S}_k^{T})^{-1} \tag{6.42}$$

$$\hat{\boldsymbol{f}}^{i}{}_{k|k} = \hat{\boldsymbol{f}}^{i}{}_{k|k-1} + \mathbf{K}_k^{\boldsymbol{f}^i}(\boldsymbol{y}_k - \boldsymbol{h}(\bar{\boldsymbol{x}}_{k|k-1}) - \mathbf{S}_k\hat{\boldsymbol{f}}^{i}{}_{k|k-1}) \tag{6.43}$$

$$\mathbf{P}_{k|k}^{\boldsymbol{f}^i} = (\mathbf{I} - \mathbf{K}_k^{\boldsymbol{f}^i}\mathbf{S}_k)\mathbf{P}_{k|k-1}^{\boldsymbol{f}^i} \tag{6.44}$$

3. The coupling equations of the IOTSEKF are:

$$\bar{\boldsymbol{u}}_{k-1} = (\bar{\mathbf{U}}_k - \mathbf{U}_k)\hat{\boldsymbol{f}}^{i}{}_{k-1|k-1} \tag{6.45}$$

$$\bar{\mathbf{Q}}_{k-1} = \mathbf{\Gamma}_{k-1}\mathbf{Q}_{k-1}\mathbf{\Gamma}_{k-1}^{T} - \mathbf{Q}_{k-1}^{\boldsymbol{x}\boldsymbol{f}^i}\bar{\mathbf{U}}_k^{T} - \mathbf{U}_k(\mathbf{Q}_{k-1}^{\boldsymbol{x}\boldsymbol{f}^i} - \bar{\mathbf{U}}_k\mathbf{Q}_{k-1}^{\boldsymbol{f}^i})^{T} \tag{6.46}$$

$$\bar{\mathbf{U}}_k = \mathbf{\Phi}_{k-1}\mathbf{V}_{k-1} + \mathbf{\Gamma}_{k-1} \tag{6.47}$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{U}_k \tag{6.48}$$

where $\mathbf{Q}_{k-1}$ the covariance matrix defined by $E\{\boldsymbol{w}_k \boldsymbol{w}_l^T\} = \mathbf{Q}_k \delta_{kl}$. $\mathbf{U}_k$ and $\mathbf{V}_k$ are the two-stage blending matrices given by

$$\mathbf{U}_k = \bar{\mathbf{U}}_k + (\mathbf{Q}_{k-1}^{\boldsymbol{x}\boldsymbol{f}^i} - \bar{\mathbf{U}}_k \mathbf{Q}_{k-1}^{\boldsymbol{f}^i})(\mathbf{P}_{k|k-1}^{\boldsymbol{f}^i})^{-1} \tag{6.49}$$

$$\mathbf{V}_k = \mathbf{U}_k - \mathbf{K}_k^{\bar{\boldsymbol{x}}} \mathbf{S}_k \tag{6.50}$$

4. The final state estimate and its covariance matrix using the IOTSEKF are given as follows:

$$\hat{\boldsymbol{x}}_{k|k} = \bar{\boldsymbol{x}}_{k|k} + \mathbf{V}_k \hat{\boldsymbol{f}}^i{}_{k|k} \tag{6.51}$$

$$\mathbf{P}_{k|k}^{\boldsymbol{x}} = \mathbf{P}_{k|k}^{\bar{\boldsymbol{x}}} + \mathbf{V}_k \mathbf{P}_{k|k}^{\boldsymbol{f}^i} \mathbf{V}_k^T \tag{6.52}$$

The final fault estimation is given in Equation (6.43). This concludes the introduction of the IOTSEKF. For the benefit of using the IOTSEKF, please refer to [123] and the reference therein.

Although this IOTSEKF can estimate both the states and faults, its performance can be degraded by bad choices of $\mathbf{Q}_k^{\boldsymbol{f}^i}$ and $\mathbf{Q}_k^{\boldsymbol{x}\boldsymbol{f}^i}$. According to [33], $\mathbf{Q}_k^{\boldsymbol{x}\boldsymbol{f}^i}$ can be chosen to be zero. The present chapter proposes an adaptive update scheme for $\mathbf{Q}_k^{\boldsymbol{f}^i}$.

Define $\hat{\mathbf{Q}}_k$ as

$$\hat{\mathbf{Q}}_k := \mathbf{C}_{r,k} - \mathbf{H}_k \boldsymbol{\Gamma}_{k-1} \mathbf{Q}_{k-1} \boldsymbol{\Gamma}_{k-1}^T \mathbf{H}_k^T / \Delta t - \mathbf{R}_k \tag{6.53}$$

where $\mathbf{C}_{r,k} = \frac{1}{N} \sum_{j=k-N+1}^{k} \boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^T$. N is the size of a moving window. In this chapter, $N = 10$. $\boldsymbol{\gamma}_j$ is the innovation at time step j which can be calculated by

$$\boldsymbol{\gamma}_j = \boldsymbol{y}_j - \boldsymbol{h}(\boldsymbol{\Phi}_{j-1} \hat{\boldsymbol{x}}_{j-1|j-1} + \boldsymbol{\Gamma}_{j-1} \hat{\boldsymbol{f}}_{j-1|j-1}^i) \tag{6.54}$$

In Equation (6.53), $\mathbf{Q}_k$ and $\mathbf{R}_k$ are the process noise and measurement noise covariance matrices, which are defined in Equations (6.46) and (6.36) respectively. Take the GS-KM as an example, $\mathbf{R}_k$ can be readily determined by the following

$$\mathbf{R}_k = \text{diag}(\sigma_{U_N}^2, \sigma_{U_E}^2, \sigma_{U_D}^2, \sigma_\phi^2, \sigma_\theta^2, \sigma_\psi^2) \tag{6.55}$$

where $\sigma_{(\bullet)}$ is given in Table 6.1. Regarding the process noise covariance matrix $\mathbf{Q}_k$, it should be noted that the input of the KM used by the IOTSEKF is $\boldsymbol{u}_m$. As can be seen from Equation (6.1), the only process noise is $\boldsymbol{w}$ which is given in Equation (6.7). This is the noise in the IMU sensor measurements. Therefore, $\mathbf{Q}_k$ can be determined by

$$\mathbf{Q}_k = \text{diag}(\sigma_{Ax}^2, \sigma_{Ay}^2, \sigma_{Az}^2, \sigma_p^2, \sigma_q^2, \sigma_r^2) \tag{6.56}$$

This is also an advantage when using the GS-KM as the model for fault reconstruction since both the process noise and output noise covariance matrices are defined explicitly.

According to covariance matching techniques [88], $\mathbf{Q}_k^{\boldsymbol{f}^i}$ can be updated based on the main diagonal of the following matrix

$$\boldsymbol{\Gamma}_{k-1}^{-1} \mathbf{H}_k^{-1} \tilde{\mathbf{Q}}_k (\mathbf{H}_k^T)^{-1} (\boldsymbol{\Gamma}_{k-1}^T)^{-1} (\Delta t)^2 \tag{6.57}$$

where $\tilde{\mathbf{Q}}_k$ is determined

$$\tilde{\mathbf{Q}}_k = \text{diag}(\max\{0, \hat{Q}_{11}\}, ..., \max\{0, \hat{Q}_{mm}\}) \tag{6.58}$$

with $\hat{Q}_{jj}$, $j = 1, 2, ..., m$ the jth diagonal element of $\hat{\mathbf{Q}}_k$ which is defined in Equation (6.53).

The IOTSEKF with this adaption of $\mathbf{Q}_k^{f^i}$ is called ATSEKF. In the following of this chapter, this ATSEKF is used as the filter for both state and fault reconstruction.

6.3.2. CLOSED-LOOP VALIDATION VS OPEN-LOOP VALIDATION

In this section, two different validation strategies are presented. The aircraft model used in this section is the ADMIRE benchmark model [142]. The controller is the baseline controller of the benchmark model. It contains a longitudinal controller which controls the airspeed, pitch rate and load factor, and a lateral controller which controls the roll rate and angle of sideslip [142]. The controller design is based on the PID control law. More detailed information can be found in [142].

The first validation strategy is the closed-loop validation which is shown in Figure 6.2(a). This is the strategy to achieve FTC in the presence of sensor faults. As can be seen from the figure, after the occurrence of the sensor faults, a fault reconstruction block is implemented to compensate for the sensor faults.

Let $\boldsymbol{u}_m^0$ denote the IMU sensor measurements without faults, then

$$\boldsymbol{u}_m^0 = \boldsymbol{u}^0 + \boldsymbol{w} = \boldsymbol{u}_m - \boldsymbol{f}^i \tag{6.59}$$

where $\boldsymbol{u}_m$, $\boldsymbol{u}^0$, $\boldsymbol{w}$ and $\boldsymbol{f}^i$ are defined in Equations (6.5), (6.6), (6.7) and (6.8) respectively. Let subscript 'CL' denote the corresponding variables in the closed-loop validation. The fault reconstruction block receives the measurements $\boldsymbol{y}_{m,CL}$ and $\boldsymbol{u}_{m,CL}$, and output unbiased estimates of the state $\boldsymbol{x}_{CL}$ and the faults $\boldsymbol{f}_{CL}^i$ online by making use of the ATSEKF. Without this fault reconstruction block, the estimate of $\boldsymbol{x}_{CL}$ will be biased due to the faults in the sensors.

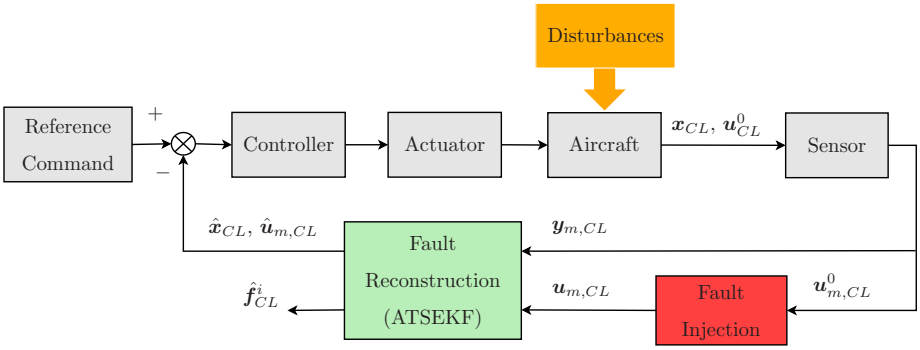
Since $\boldsymbol{f}_{CL}^i$ are also estimated in an unbiased sense by using the ATSEKF, a more reliable IMU measurement can be obtained by

$$\hat{\boldsymbol{u}}_{m,CL} = \boldsymbol{u}_{m,CL} - \hat{\boldsymbol{f}}_{CL}^i \tag{6.60}$$

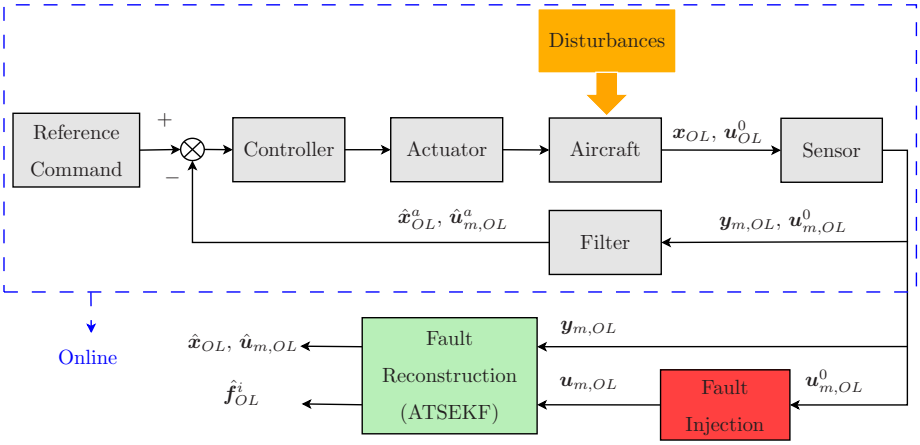
where $\hat{\boldsymbol{u}}_{m,CL}$ denotes the estimated IMU measurements. Due to this fault reconstruction, FTC can be achieved in the presence of faults by making use of $\hat{\boldsymbol{x}}_{CL}$ and $\hat{\boldsymbol{u}}_{m,CL}$. The estimated faults $\hat{\boldsymbol{f}}_{CL}^i$ can also be used for fault detection.

Although the sensor FTC (closed-loop validation) can be validated in simulation, it is difficult to be validated in real-life especially for fixed-wing manned aircraft. For these aircraft, validation with fault injection during the flight is risky. Consequently, this prevents researchers from validating fault reconstruction approaches under real-life uncertainties and disturbances.

In this research, an alternative validation strategy (open-loop validation strategy) is implemented and compared to the closed-loop validation. The open-loop validation strategy is shown in Figure 6.2(b) where the subscript 'OL' denotes the corresponding variables in the open-loop validation. It is seen from the figure that only the blocks



(a) Closed-loop validation strategy (Sensor FTC strategy). The states which are irrelevant to the fault reconstruction are not shown in the figure. All the blocks are run online including the fault injection and fault reconstruction. The subscript '*CL*' denotes closed-loop validation.



(b) Open-loop validation strategy. The states which are irrelevant to the fault reconstruction are not shown in the figure. The blocks within the blue dashed line are run online while the fault injection and reconstruction are run off-line and are not in the feedback loop. The subscript '*OL*' denotes open-loop validation.

Figure 6.2: Block diagram of two different validation strategies

within the blue dashed line are run online or on-board, and they are free from the influence of the sensor faults. The fault injection and reconstruction are not in the feedback loop. Specifically, the faults are injected into the recorded measurements and are estimated by the fault reconstruction block off-line. In the figure, $\hat{\boldsymbol{x}}^a_{OL}$ and $\hat{\boldsymbol{u}}^a_{m,OL}$, which are the estimated $\boldsymbol{x}_{OL}$ and $\boldsymbol{u}_{m,OL}$ by the Filter block on-board the aircraft, are assumed to be unbiased.

Comparing these two validation strategies, it can be inferred that if the estimates $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{f}}^i$ in the closed-loop and the open-loop are unbiased, then the differences between these two strategies are limited. Note that unbiased $\hat{\boldsymbol{f}}^i$ indicates unbiased $\hat{\boldsymbol{u}}_m$. In the following section, the comparison between these two strategies will be shown.

6.3.3. SIMULATION VALIDATION IN THE ABSENCE OF SIMULATED DISTURBANCES

In this section, the results of the above-mentioned two validation strategies are shown. The fault scenario is shown in Table 6.2. No disturbance is present. The model used is the GS-KM.

The differences of the true states ($\boldsymbol{x} = [U_N, U_E, U_D, \phi, \theta, \psi]^T$) between the open-loop ($\boldsymbol{x}_{OL}$) and closed-loop ($\boldsymbol{x}_{CL}$) validations are shown in Figures 6.3(a) and Figure 6.3(b). It should be noted that for the closed-loop validation, the faults are generated online in the feedback loop. As can be seen from the figure, although the faults are in the feedback loop of closed-loop validation, the differences between $\boldsymbol{x}_{OL}$ and $\boldsymbol{x}_{CL}$ are limited even in the presence of faults. The maximum difference of U_N, U_E and U_D is smaller than 0.2 m/s and that of ϕ, θ and ψ is smaller than 5×10^{-3} rad.

The fault reconstruction results of the two validations are shown in Figures 6.3(c) and 6.3(d). As can be seen, the fault estimates $\hat{\boldsymbol{f}}^i$ ($\hat{f}_{Ax}$, $\hat{f}_{Ay}$, $\hat{f}_{Az}$ and $\hat{f}_p$, $\hat{f}_q$, $\hat{f}_r$) of the two validation strategies, denoted by $\hat{\boldsymbol{f}}^i_{OL}$ and $\hat{\boldsymbol{f}}^i_{CL}$ in the open-loop and closed-loop validations respectively, closely match each other. This indicates that reconstructed faults are similar no matter whether it is the open-loop or closed-loop validation. The state estimates of the open-loop and closed-loop validations, denoted by $\hat{\boldsymbol{x}}_{OL}$ and $\hat{\boldsymbol{x}}_{CL}$ respectively, are both unbiased. The differences between $\hat{\boldsymbol{x}}_{OL}$ and $\hat{\boldsymbol{x}}_{CL}$ are similar to those shown in Figure 6.3(a) and Figure 6.3(b) and are not shown.

Let us further check the differences between $\hat{\boldsymbol{f}}^i_{OL}$ and $\hat{\boldsymbol{f}}^i_{CL}$ which are shown in Figures 6.3(e) and 6.3(f). It can be seen that the differences mainly occur in the beginning or the end of the fault period (except for $\hat{f}_{Az}$ and $\hat{f}_q$ which are oscillatory faults). During 10 s$< t <$ 30 s, although the true states of the two validations ($\boldsymbol{x}_{OL}$ and $\boldsymbol{x}_{CL}$) are different (as shown in Figure 6.3(a) and Figure 6.3(b)), the differences between $\hat{\boldsymbol{f}}^i_{OL}$ and $\hat{\boldsymbol{f}}^i_{CL}$ are limited. This becomes more obvious when the faults are removed after $t >$ 30 s. $\boldsymbol{x}_{OL}$ and $\boldsymbol{x}_{CL}$ are different (see Figure 6.3(a) and Figure 6.3(b)), but $\hat{\boldsymbol{f}}^i_{OL}$ and $\hat{\boldsymbol{f}}^i_{CL}$ are all zero-mean.

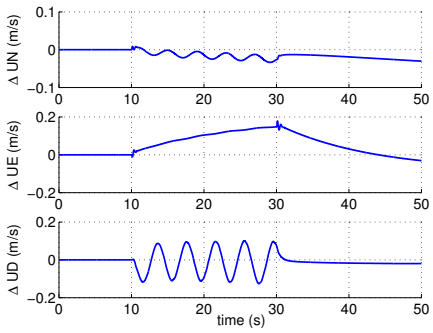
The above results demonstrate that if the fault reconstruction block can provide unbiased estimates of $\boldsymbol{x}$ and $\boldsymbol{f}^i$, the differences between the two validation strategies are limited. Under this condition, the open-loop validation, which injects and estimates the faults offline, is also valid. Therefore, open-loop validation can be used to test the performance of fault reconstruction approaches under real-life disturbances and uncertainties, as will be shown in Section 6.5.

In the following sections, the subscript 'OL' and 'CL' will be dropped for readability. In the remainder of this section, all the results are obtained using the closed-loop validation while in Section 6.5, all the results are obtained using the open-loop validation.

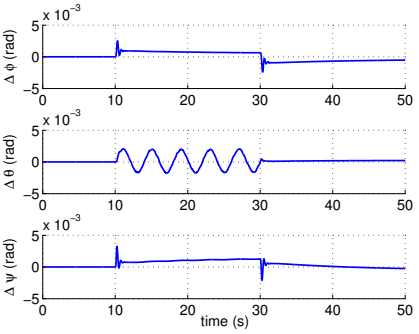
6.3.4. SIMULATION VALIDATION IN THE PRESENCE OF SIMULATED WIND SHEAR

In this section, the simulated wind shear, as shown in Figure 6.1(a), is included in the simulation model but is assumed unknown for the design of the fault reconstruction approaches. The GS-KM together with the ATSEKF are used for the fault reconstruction. The AS-KM together with the ATSEKF are also used for comparison.

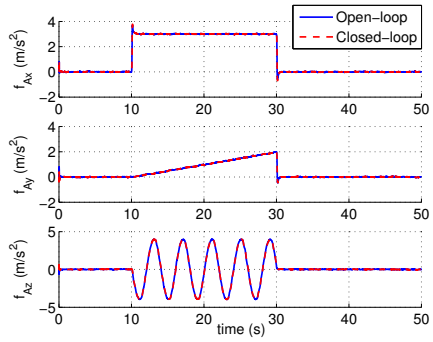
The fault reconstructions of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and the GS-KM are compared in Figure 6.4(a). It is seen that the fault estimates using the GS-KM can follow



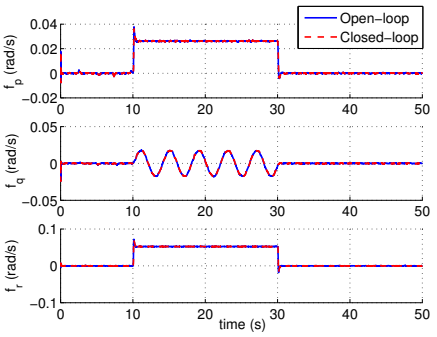
(a) Difference of the true U_N, U_E and U_D between the closed-loop and open-loop validations



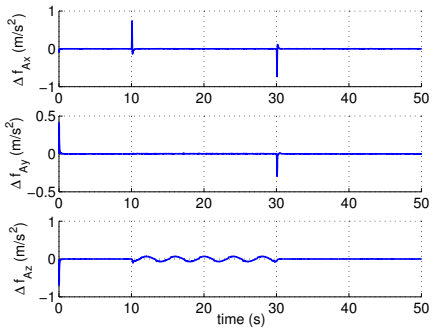
(b) Difference of the true ϕ, θ and ψ between the closed-loop and open loop validations



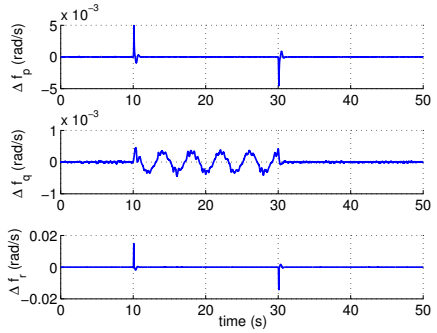
(c) Estimated f_{Ax}, f_{Ay} and f_{Az} of the closed-loop and open-loop validations



(d) Estimated f_p, f_q and f_r of the closed-loop and open-loop validations

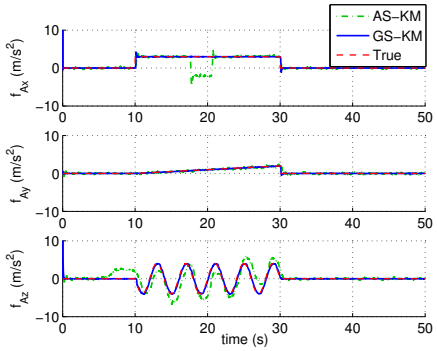


(e) Difference of the estimated f_{Ax}, f_{Ay} and f_{Az} between the closed-loop and open-loop validations

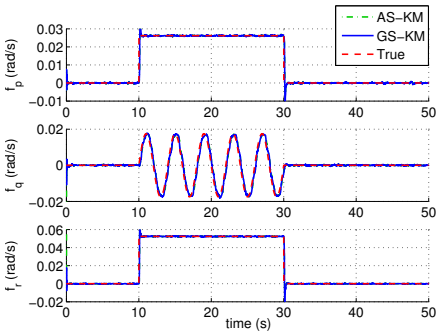


(f) Difference of the estimated f_p, f_q and f_r between the closed-loop and open-loop validations

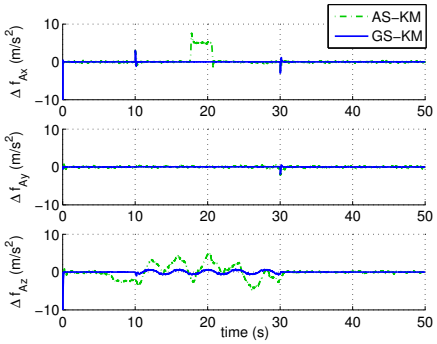
Figure 6.3: Comparisons between the open-loop and closed-loop validations in the absence of disturbances



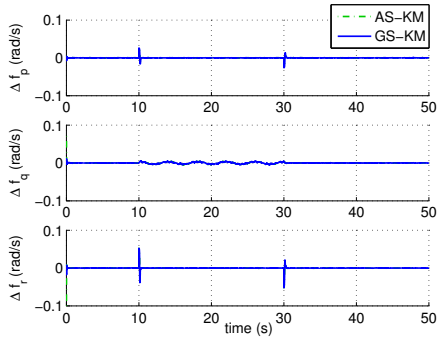
(a) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



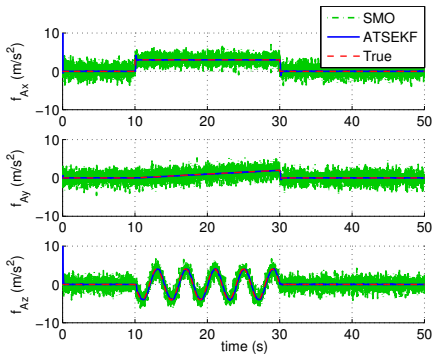
(b) True and estimated f_p, f_q and f_r using the AS-KM and GS-KM



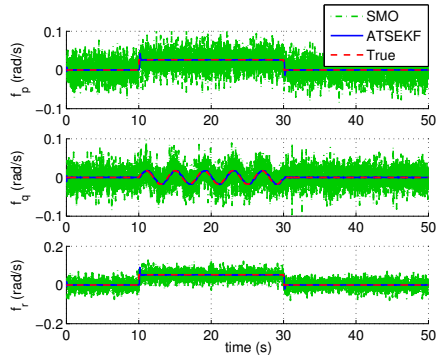
(c) Estimation errors of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(d) Estimation errors of f_p, f_q and f_r using the AS-KM and GS-KM



(e) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the SMO and the ATSEKF together with the GS-KM



(f) True and estimated f_p, f_q and f_r using the SMO and the ATSEKF together with the GS-KM

Figure 6.4: State estimation and fault estimation results in the presence of simulated wind shear

the true faults well. The fault estimation using the AS-KM is not satisfactory. Estimation of f_{Az} significantly deviates from the true faults around $t = 5$ s. This is caused by the fact the $w_w > 0$ after $t = 5$ s. The vertical wind is interpreted into a fault in the sensor which measures a_z. The reason can be explained using Equation (6.29). When w_w is not constant, f_{Az} is coupled with the influence of the wind which leads to the wrong fault reconstruction. Similarly, the estimation of f_{Ax} using the AS-KM between 18 s $< t <$ 20 s is wrong due to u_w which significantly decreases during 18 s $< t <$ 20 s, as shown in Figure 6.1(a). Since v_w is assumed to be zero, estimation of f_{Ay} is not influenced. The estimates of f_p, f_q and f_r using the AS-KM and GS-KM are similar and are shown in Figure 6.4(b).

The corresponding estimation errors of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and the GS-KM are shown in Figure 6.4(c). As can be seen, the errors using the GS-KM maintain zero-mean while those using the AS-KM deviate from zero. The estimation errors of f_p, f_q and f_r using the two KMs are similar and zero-mean, as shown in Figure 6.4(d).

To demonstrate the performance of the ATSEKF, a SMO [19, 143] is implemented using the GS-KM. The estimation of f_{Ax}, f_{Ay}, f_{Az} and f_p, f_q, f_r using the SMO are shown in Figures 6.4(e) and 6.4(f) respectively. As can be seen from the figure, although the faults are reconstructed, the standard deviations of the estimation are significantly larger than those of the ATSEKF. This is caused by the noise in the measurements. The influence of the noise on the fault reconstruction can be reduced by using a low-pass filter, which will be discussed in the following section.

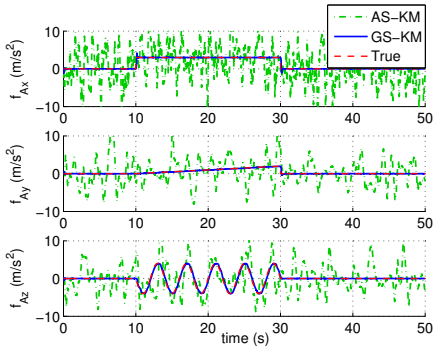
6.3.5. SIMULATION VALIDATION IN THE PRESENCE OF SIMULATED TURBULENCE

In this section, the simulated turbulence, as shown in Figure 6.1(b), is included in the simulation but is considered unknown for the design of the fault reconstruction approaches. The GS-KM is again compared with the AS-KM.

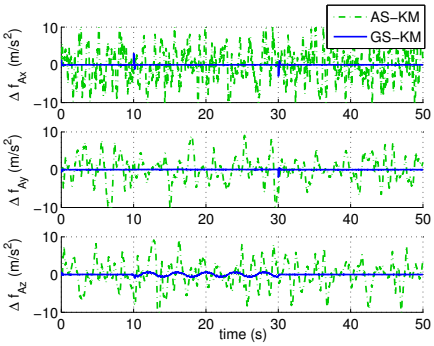
If the AS-KM is used when the turbulence is present, as can be seen from Equation (6.29), f_{Ax}, f_{Ay} and f_{Az} are coupled with the influence of the turbulence, denoted by $\boldsymbol{d}_w$. This explains why the estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM are corrupted by the turbulence, as shown in Figure 6.5(a). Using the GS-KM, satisfactory fault estimation results are achieved, as shown in Figure 6.5(a).

The estimates of f_p, f_q and f_r using the two KMs are satisfactory and are the same as in Figure 6.4(b). The estimation errors of f_p, f_q and f_r using the two KMs are the same as in Figure 6.4(d) and are not shown.

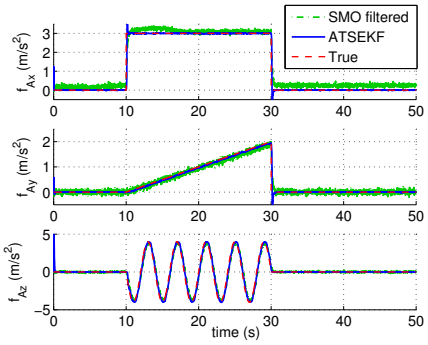
To show the performance of the ATSEKF, the SMO is implemented again to estimate the faults using the GS-KM. Since the standard deviations of the fault estimation are large, a low-pass filter is implemented to filter the estimated faults and the results are shown in Figures 6.5(c) and 6.5(d) respectively. Although the estimates of the SMO are filtered, the standard deviations are still larger than those of the ATSEKF. Therefore, in the following sections, only the results using the ATSEKF are presented.



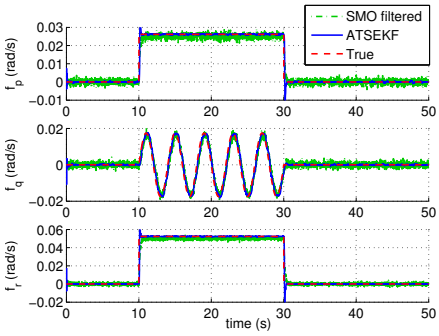
(a) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(b) Errors of estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(c) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the SMO and the ATSEKF together with the GS-KM



(d) True and estimated f_p, f_q and f_r using the SMO (filtered) and the ATSEKF together with the GS-KM

Figure 6.5: State estimation and fault estimation results in the presence of simulated turbulence

Figure 6.6: Cessna Citation II aircraft, owned by Delft University of Technology and the Dutch Aerospace Laboratory

6.4. REAL FLIGHT TEST SCENARIOS

The above results are obtained using a simulated aircraft model. However, to fully demonstrate the performance of the proposed approach, it should be validated using real flight data where realistic uncertainties and disturbances are included. The robustness of the approach should be validated using different flight conditions and wind conditions. In Lu et al. [123], real flight data is also used to validate the performance. However, no turbulence is present in the flight data. Consequently, the robustness of the proposed approach in the presence of disturbances such as time-varying wind and turbulence is not validated.

In order to validate the performance of the proposed approach in the presence of disturbances, more flight tests were performed in 2015. The aircraft used is the Cessna Citation II aircraft which is shown in Figure 6.6. Aircraft response data are recorded in various flight conditions and wind conditions. Parabolic flights or so called zero-g flights, during which the aircraft follows a parabolic trajectory to achieve a temporary sensation of weightlessness, are performed in order to test the performance of the approach over a wide range of the flight envelope with varying uncertainties. The measurement data were recorded for post-flight analysis.

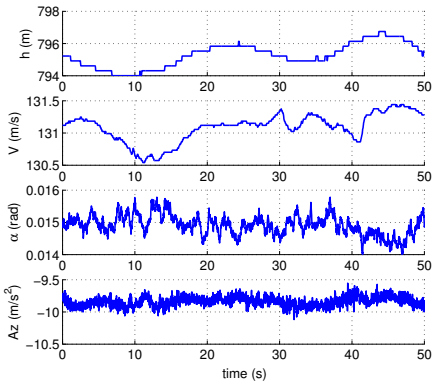
The air data are measured with a frequency of 100 Hz and the angle of attack is measured by a mechanical vane on the fuselage used for stall warning. This is different from the measurements used in the flight tests in [123] where the angle of attack is measured by the vanes mounted on an air data boom. The ground speed components are measured by a Global Positioning System receiver at 1 Hz and are linearly interpolated to obtain data with a frequency of 100 Hz. Aircraft response data in the presence of turbulence are also recorded and the crew recorded the time when the turbulence occurred.

In this chapter, three representative flight scenarios are chosen to validate the performance of the proposed approach and to show the advantage of the proposed approach over other methods. The three flight scenarios are shown in Figure 6.7 and the descriptions are as follows:

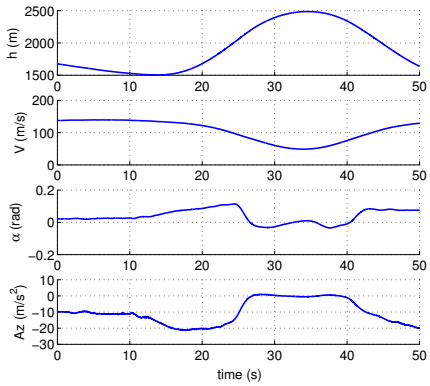
1. Scenario 1 is a level flight which is shown in Figure 6.7(a). As can be seen, the altitude of the aircraft h is nearly constant and so are other parameters such as the true airspeed V, angle of attack α and specific force A_z. In this flight scenario, no manoeuvre is performed which means that there is no controlled excitation.

2. Scenario 2 is a parabolic flight which is shown in Figure 6.7(b). As can be seen from the figure, A_z of the aircraft decreases to almost -20 m/s^2 at around $t = 16$ s and increases to almost 0 m/s^2 at around $t = 27$ s. This clearly shows the motion of the aircraft. During this period, the altitude h, true airspeed V and angle of attack α of the aircraft change significantly. At around $t = 28$ s, α even decreases below 0 rad. This scenario is to test the performance over a wide range of the flight envelope.

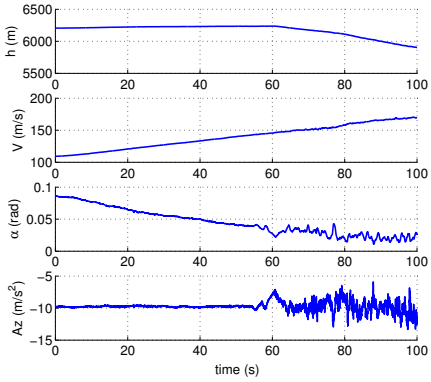
3. Scenario 3 is a flight in the presence of turbulence which is shown in Figure 6.7(c). In the first 60 s, A_z is smooth while after $t = 60$ s, it oscillates significantly. The oscillations are also observable in α. This scenario is to test the performance of the fault reconstruction approaches in the presence of turbulence. In the validation, only the data between 50 s $< t <$ 100 s are used.



(a) Aircraft response data during the level flight



(b) Aircraft response data during the parabolic flight



(c) Aircraft response data in the presence of turbulence, only the data between 50 s $< t <$ 100 s are used in the sensor fault reconstruction validation

Figure 6.7: Real flight test data in different flight scenarios

In the following section, the performance of the GS-KM will be compared to the AS-KM using these three flight scenarios.

6.5. VALIDATION USING REAL FLIGHT DATA UNDER VARIOUS FLIGHT SCENARIOS

In this section, the performance of the proposed approach is validated using the flight scenarios in the previous section. Since injecting faults during flight is unfeasible due to safety concerns, this section uses the open-loop validation strategy to validate the fault reconstruction approaches. As shown in Section 6.3, this is also an effective validation strategy if the states and faults are estimated in an unbiased sense. All the faults are injected into the recorded data and the fault scenario is the same as in Table 6.2. The filter used for the estimation of the states and faults is the ATSEKF. The performance of the proposed approach as well as the comparison with the approach using the AS-KM are presented in the following.

6.5.1. VALIDATION OF REAL FLIGHT TEST SCENARIO 1

In this section, the performance of the fault reconstruction of flight scenario 1 is presented. The state estimation performance of the GS-KM is shown in Figure 6.8(a) and 6.8(b). In the figure, the measured states are plotted since real states are not available. As can be seen, the estimated states closely match the measured states. The differences between the measured and estimated states are shown in Figures 6.8(c) and 6.8(d). It can be seen that even during the period when there are faults, the differences between the measured and estimated states are small. This demonstrates that the ATSEKF can provide unbiased state estimates even in the presence of faults.

The fault estimates using the AS-KM and GS-KM are compared in Figures 6.9(a) and 6.9(b). The true injected faults are also plotted to show the performance of the fault reconstruction. It can be seen from the figure that the performances of using the GS-KM and the AS-KM are comparable. Both of the estimates follow the true faults well.

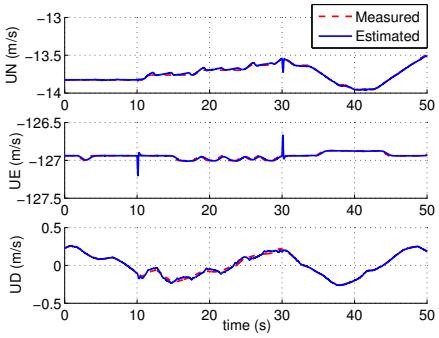
The estimation errors of f_{Ax}, f_{Ay} and f_{Az} using the two KMs are shown in Figure 6.9(c) while those of f_p, f_q and f_r are given in Figure 6.9(d). It is seen that the estimation errors using the two KMs are both zero-mean. The estimation error of f_{Az} using the AS-KM is slightly larger than that using the GS-KM in the presence of faults, as can be seen from Figure 6.9(c).

The RMSEs of the fault estimates using the two KMs are compared in Figures 6.9(e) and 6.9(f). The RMSEs of the estimates using the two approaches are comparable except for f_{Az}. For the RMSE of the estimate of f_{Az}, the result using the AS-KM is almost twice the result using the GS-KM.

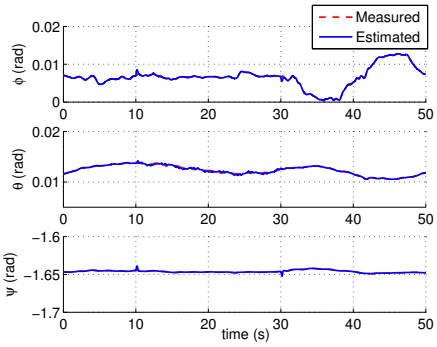
In this flight scenario, the performance using the two KMs is similar. According to Equation (6.28), it can be inferred that the wind during this level flight is close to constant.

6.5.2. VALIDATION OF REAL FLIGHT TEST SCENARIO 2

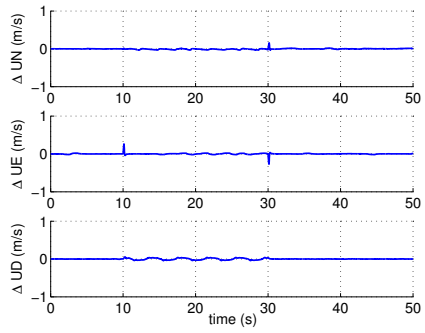
In this flight scenario, the aircraft is performing a parabolic flight. As can be seen from the measured states shown in Figure 6.7(b). All the states change significantly especially



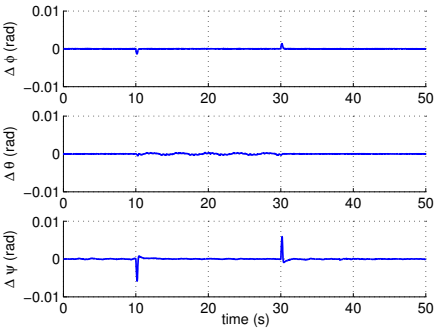
(a) Measured and estimated U_N, U_E and U_D using the GS-KM



(b) Measured and estimated ϕ, θ and ψ using the GS-KM

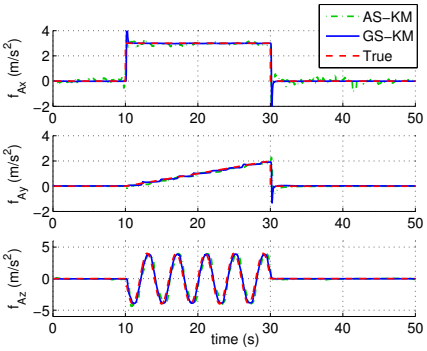


(c) Difference between measured and estimated U_N, U_E and U_D using the GS-KM

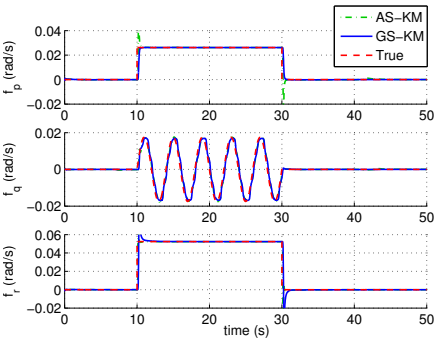


(d) Difference between measured and estimated ϕ, θ and ψ using the GS-KM

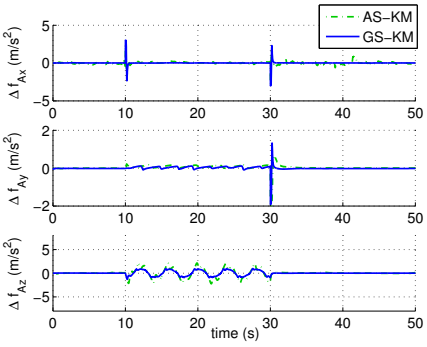
Figure 6.8: State estimation results of real flight test scenario 1



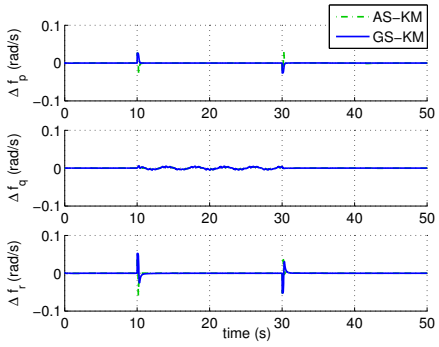
(a) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



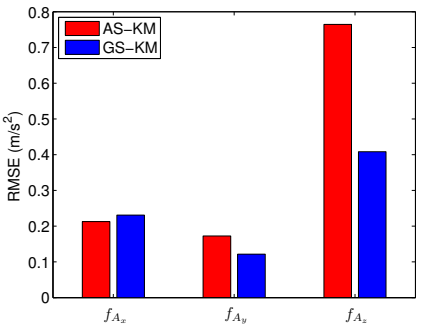
(b) True and estimated f_p, f_q and f_r using the AS-KM and GS-KM



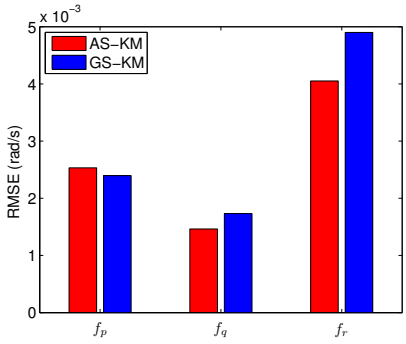
(c) Errors of estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(d) Errors of estimates of f_p, f_q and f_r using the AS-KM and GS-KM

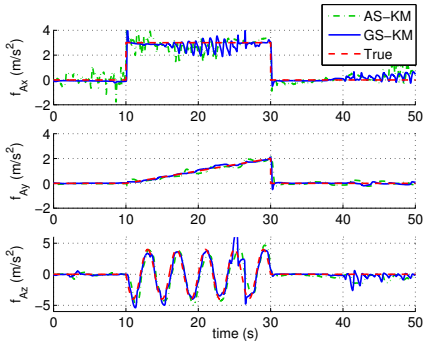


(e) RMSEs of the estimates of f_{Ax}, f_{Ay} and f_{Az}

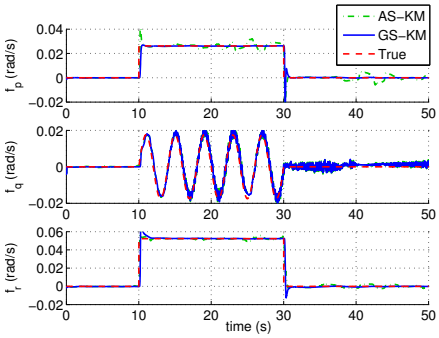


(f) RMSEs of the estimates of f_p, f_q and f_r

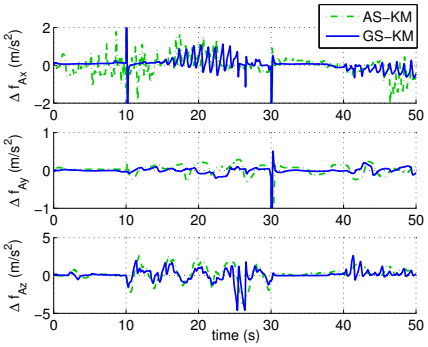
Figure 6.9: Fault estimation results of real flight test scenario 1 (level flight)



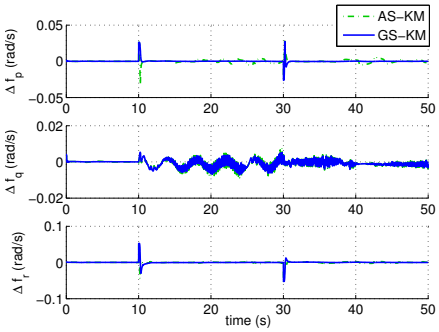
(a) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



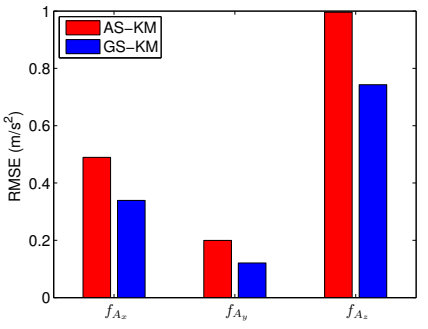
(b) True and estimated f_p, f_q and f_r using the AS-KM and GS-KM



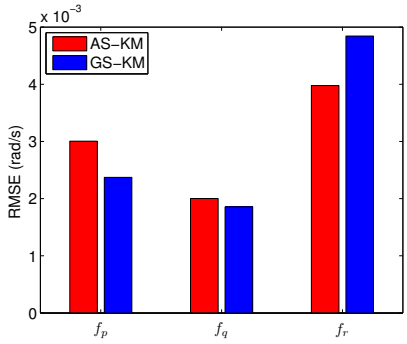
(c) Errors of estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(d) Errors of estimates of f_p, f_q and f_r using the AS-KM and GS-KM



(e) RMSEs of the estimates of f_{Ax}, f_{Ay} and f_{Az}



(f) RMSEs of the estimates of f_p, f_q and f_r

Figure 6.10: Fault estimation results of real flight test scenario 2 (parabolic flight)

during 16 s < t < 26 s and 40 s < t < 50 s. Therefore, it can test the performance over a wide range of the flight envelope.

The estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM are compared in Figure 6.10(a). The results using the AS-KM are worse than those using the GS-KM. It is noticed that during 16 s < t < 26 s and 40 s < t < 50 s, the performances using the two KMs both degrade. This is caused by the fact that the system states are changing significantly. However, the performance using the GS-KM is better than that using the AS-KM. The estimates of f_{Ax} using the AS-KM deviate from the true fault condition even at the beginning of the simulation. In contrast, the estimates using the GS-KM maintain zero-mean before the injection of the faults. The estimation errors of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM are compared in Figure 6.10(c)

The estimates of f_p, f_q and f_r using the two KMs are similar, as shown in Figure 6.10(b). The corresponding errors of the estimation are shown in Figure 6.10(d). It is seen that the estimation performance is satisfactory. The maximum error is less than 0.05 rad/s even when the faults are injected.

The RMSEs of the fault estimates using the two KMs are shown in Figures 6.10(e) and 6.10(f), which confirms the better performance of using the GS-KM over the AS-KM.

6.5.3. VALIDATION OF REAL FLIGHT TEST SCENARIO 3

In this flight scenario, the aircraft is flying in turbulence. The estimates of f_{Ax}, f_{Ay} and f_{Az} using the two KMs are compared in Figure 6.11(a). The influence of turbulence on fault reconstruction when using the AS-KM is noticeably seen in the figure after t = 10 s. The estimates of the ATSEKF using the AS-KM are corrupted by the turbulence, which is similar to those shown in Figure 6.5(a). In contrast, the ATSEKF using the GS-KM still maintains satisfactory performance and is significantly better than that using the AS-KM. The fault estimates using the GS-KM can follow the true faults well without being influenced by the turbulence. This clearly demonstrates the superiority of the GS-KM.

The estimates of f_p, f_q and f_r using the two approaches are still comparable, as shown in Figure 6.11(b). It shows that although the turbulence can influence the estimation of the accelerometer faults (f_{Ax}, f_{Ay} and f_{Az}), its influence on the estimation of rate gyro faults (f_p, f_q and f_r) is limited.

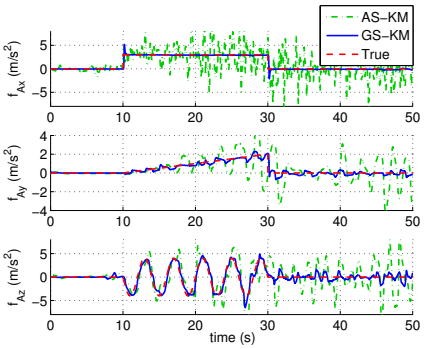
The fault estimation errors using the two KMs are compared in Figure 6.11(c) and 6.11(d). It is seen that the estimation errors using the GS-KM remain small despite the presence of the turbulence.

The RMSEs of the estimation of f_{Ax}, f_{Ay} and f_{Az} using the two approaches are shown in Figure 6.11(e). The RMSEs using the GS-KM are prominently smaller than those using the AS-KM. However, the RMSEs of the estimates of f_p, f_q and f_r using the two approaches are comparable, as shown in Figure 6.11(f).

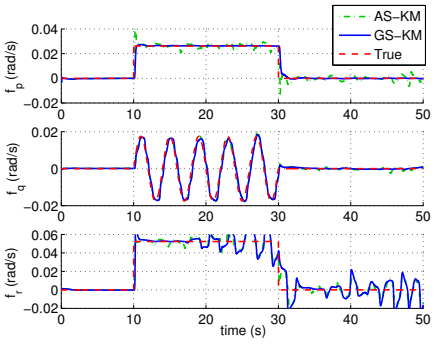
6.5.4. DISCUSSION ON ROBUSTNESS AGAINST TURBULENCE

In this section, the robustness of the proposed approach (GS-KM) and the approach using the AS-KM is compared and summarized. Without loss of generality, the RMSE of the estimation of f_{Ax} is taken as an example.

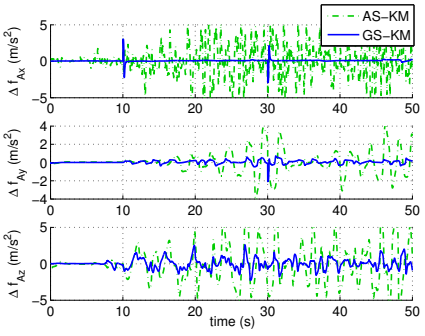
The RMSEs of the estimation of f_{Ax} using the AS-KM and the GS-KM in the three flight scenarios are plotted in Figure 6.12. It is evident that the approach using the AS-



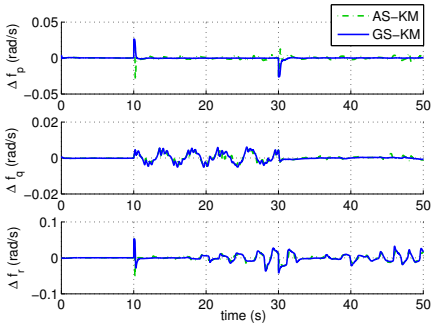
(a) True and estimated f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



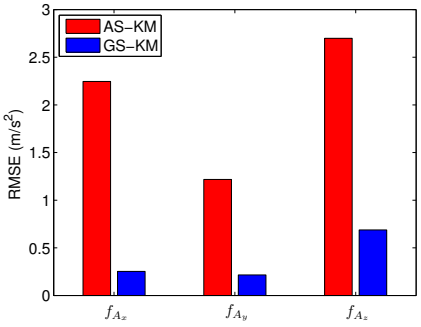
(b) True and estimated f_p, f_q and f_r using the AS-KM and GS-KM



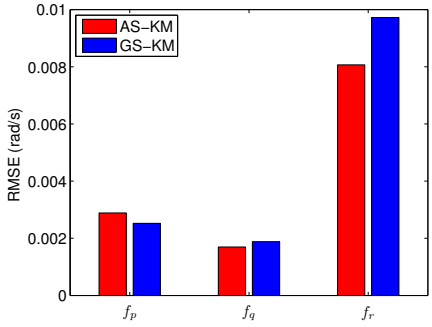
(c) Errors of estimates of f_{Ax}, f_{Ay} and f_{Az} using the AS-KM and GS-KM



(d) Errors of estimates of f_p, f_q and f_r using the AS-KM and GS-KM



(e) RMSEs of the estimates of f_{Ax}, f_{Ay} and f_{Az}



(f) RMSEs of the estimates of f_p, f_q and f_r

Figure 6.11: Fault estimation results of real flight test scenario 3 (during turbulence)

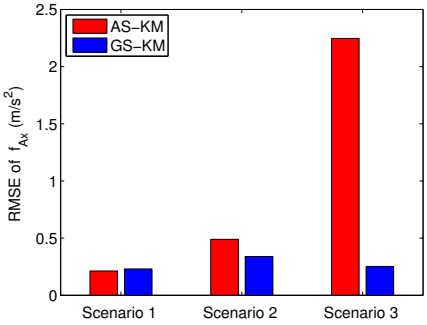


Figure 6.12: RMSEs of the estimates of f_{Ax} using the AS-KM and GS-KM in the three different flight scenarios

KM is not robust. The performance changes when the flight scenarios varies. The worst performance occurs when the turbulence is present, which demonstrates that it is sensitive to disturbances. This implies the necessity of considering external disturbances such as turbulence when dealing with aircraft sensor reconstruction.

In contrast, the performance of the proposed approach which uses the GS-KM is more robust to disturbances. The performance does not change significantly even in the presence of turbulence. The performance is notably better than that of using the AS-KM.

Finally, the ground speed measurements are obtained at 1 Hz and are linearly interpolated to obtain data at 100 Hz. To obtain a better performance, ground speed measurements with a high update rate should be applied.

6.6. CONCLUSIONS

This chapter proposes a new approach for Inertial Measurement Unit sensor fault reconstruction by making use of the ground-speed-based kinematic model of the aircraft flying in a rotating earth reference frame. Two strategies are proposed for the validation of sensor fault reconstruction approaches: closed-loop and open-loop strategy. Both validation strategies use a newly-developed Adaptive Two-Stage Extended Kalman Filter to estimate the states and faults. The Adaptive Two-Stage Extended Kalman Filter can adaptively update the covariance matrix for fault estimation. The fault reconstruction performance using the ground-speed-based kinematic model is compared to the approach which uses the airspeed-based kinematic model, which shows the performance of the proposed approach.

More importantly, the proposed approach is validated using real flight test data. The performance is tested in various flight conditions and wind conditions including turbulence. The proposed approach is found to be robust in various flight conditions and disturbances such as turbulence, which demonstrates the advantage of the proposed approach.

It is suggested that the proposed approach be incorporated in the aircraft Fault Detection and Isolation and Fault-Tolerant Control systems to enhance the performance of the aircraft.

7

NOVEL FRAMEWORK FOR STATE AND UNKNOWN INPUT ESTIMATION: APPLICATION TO AIR DATA SENSOR FAULT DIAGNOSIS IN THE PRESENCE OF TURBULENCE

In the previous chapter, the IMU FDD in the presence of model uncertainties and external disturbances is resolved. In this chapter, the ADS FDD in the presence of external disturbances is considered. First, the DMAE, which was proposed in Chapter 5, is further extended such that it can deal with the case when the effects of the faults and external disturbances are coupled. To demonstrate the performance of the extended DMAE, two simple examples are given. The extended DMAE is also compared to existing approaches to demonstrate its superior performance. Then, the extended DMAE is applied to the ADS FDD in the presence of turbulence. Finally, the performance of the extended DMAE for ADS FDD in the presence of turbulence using real flight data is presented.

The software of the work in this chapter is available at: https://www.researchgate.net/profile/Peng_Lu15/publications?pubType=dataset.

Parts of this chapter are based on:

P Lu, E van Kampen, C C de Visser, Q P Chu. Novel Framework to State and Unknown Input Estimation of Linear Time-varying Systems. *Automatica*, (2016), http://dx.doi.org/10.1016/j.automatica.2016.07.009. [144]

The design of unknown-input decoupled observers and filters requires the assumption of an existence condition in the literature. This chapter addresses an unknown input filtering problem where the existence condition is not satisfied. Instead of designing a traditional unknown input decoupled filter, a Double-Model Adaptive Estimation approach is extended to solve the unknown input filtering problem. It is proved that the state and the unknown inputs can be estimated and decoupled using the extended Double-Model Adaptive Estimation approach without satisfying the existence condition. Finally, the proposed approach is applied to air data sensor fault diagnosis in the presence of turbulence.

7.1. INTRODUCTION

Faults and model uncertainties such as disturbances can be represented as unknown inputs. The problem of filtering in the presence of unknown inputs has received intensive attention in the past three decades.

It is common to treat the unknown inputs as part of the system state and then estimate the unknown inputs as well as the system state [145]. This is an augmented Kalman filter, whose computational load may become excessive when the number of the unknown inputs is comparable to the states of the original system [32]. Friedland [32] derived a two-stage Kalman filter which decomposes the augmented filter into two reduced-order filters. However, Friedland's approach is only optimal in the presence of a constant bias [145]. Hsieh and Chen derived an optimal two-stage Kalman filter which performance is also optimal for the case of a random bias [145].

On the other hand, unknown input filtering can be achieved by making use of unbiased minimum-variance estimation [146–151]. Kitanidis [147] first developed an unbiased recursive filter based on the assumption that no prior information about the unknown input is available [84]. Hou and Patton [149] used an unknown-input decoupling technique and the innovation filtering technique to derive a general form of unknown-input decoupled filters [149, 150]. Darouach, Zasadzinski and Boutayeb [152] extended Kitanidis' method using a parameterizing technique to derive an optimal estimator filter. The problem of joint input and state estimation, when the unknown inputs only appear in the system equation, was addressed by Hsieh [150] and Gillijns and De Moor [153]. Gillijns and De Moor [84] further proposed a recursive three-step filter for the case when the unknown inputs also appear in the measurement equation. However, their approach requires the assumption that the distribution matrix of the unknown inputs in the measurement equation is of full rank. Cheng et al. [154] proposed a global optimal filter which removed this assumption, but this filter is limited to state estimation [155]. Later, Hsieh [156] presented a unified approach to design a specific globally optimal state estimator which is based on the desired form of the distribution matrix of the unknown input in the measurement equation [156].

However, all the above-mentioned filters require the assumption that an existence condition is satisfied. This necessary condition is given by Hou and Patton [149] and Darouach, Zasadzinski and Boutayeb [152], in the form of rank condition (7.5). Hsieh [156] presents different decoupling approaches for different special cases. However, these approaches also have to satisfy the existence condition (7.5). In some applications, such as that presented in the current chapter, the existence condition is not satisfied. Therefore, a traditional unknown input decoupled filter can not be designed.

Recently, particle filters are also applied to unknown input estimation [28, 157, 158]. These filters can cope with systems with non-Gaussian noise and have a number of applications such as for robot fault detection [159–161]. In this chapter, the performance of particle filters will be compared with that of our approach.

This chapter proposes an extended DMAE approach, which can cope with the unknown input filtering problem when a traditional unknown input filter can not be designed. The original DMAE approach, which was proposed by Lu et al. [92] for the estimation of unknown inputs in the measurement equation, is extended to allow estimation of the unknown inputs which appear both in the system equation and the measurement equation. The unknown inputs are augmented as system states and are modeled as random walk processes. The unknown inputs in the system equation are assumed to be Gaussian random processes of which covariances are estimated on-line. It is proved that the state and unknown inputs can be estimated and decoupled while not requiring the existence condition. Two illustrative examples are given to demonstrate the effectiveness of the proposed approach with comparison to other methods from literature such as the RTS-KF [84] and Optimal Two-Stage Kalman Filter (OTSKF) [145].

The problem of ADS FDD in the presence of turbulence is challenging because the existence condition is not satisfied. In this chapter, first the model used for ADS FDD in the presence of turbulence, which shows that the effects of the turbulence and faults are coupled, is introduced. Then the proposed approach is validated using both simulated and real turbulence data.

The structure of the chapter is as follows: the preliminaries of the chapter are given in Section 7.2, formulating the filtering problem when the existence condition is not satisfied and generalizing the DMAE approach. In Section 7.3, the extension of the DMAE approach to the filtering problem when the unknown inputs appear both in the system equation and the measurement equation is presented. It is proved that the state and the unknown inputs can still be estimated and decoupled in Section 7.4. In Section 7.5, two illustrative examples are given to show the performance of the proposed approach with comparison to some existing unknown-input decoupled filters. The proposed approach is applied to ADS FDD in the presence of turbulence in Section 7.6. Finally, Section 7.7 concludes the chapter.

7.2. THE DMAE APPROACH

This section presents the problem formulation and the DMAE approach.

7.2.1. PROBLEM FORMULATION

Consider the following linear time-varying system:

$$\boldsymbol{x}_{k+1} = \mathbf{A}_k \boldsymbol{x}_k + \mathbf{B}_k u_k + \mathbf{E}_k \boldsymbol{d}_k + \boldsymbol{w}_k \tag{7.1}$$

$$\boldsymbol{y}_k = \mathbf{H}_k \boldsymbol{x}_k + \mathbf{F}_k \boldsymbol{f}_k + \boldsymbol{v}_k \tag{7.2}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ represents the system states, $\boldsymbol{y}_k \in \mathbb{R}^m$ the measurements, $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ are the unknown inputs. Specifically, $\boldsymbol{d}_k \in \mathbb{R}^{n_d}$ the disturbances, $\boldsymbol{f}_k \in \mathbb{R}^{n_f}$ are the output faults. $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are assumed to be uncorrelated zero-mean white noise sequences with covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$ respectively. u_k, the known inputs, is omitted in the following

discussion because it does not affect the filter design [149]. Without loss of generality, we consider the case: $n = m = n_d = n_f$ and rank $\mathbf{H}_k$ = rank $\mathbf{E}_k$ = rank $\mathbf{F}_k = m$, which implies all the states are influenced by $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$. It should be noted that the approach proposed in this chapter can be readily extended to the case when $n \neq m$ or rank $\mathbf{H}_k \neq$ rank $\mathbf{E}_k$.

The unknown inputs are denoted as $\boldsymbol{d}'_k$, i.e., $\boldsymbol{d}'_k = \begin{bmatrix} \boldsymbol{d}_k \\ \boldsymbol{f}_k \end{bmatrix} \in \mathbb{R}^{n_{d'}}$. Then, model (7.1) and (7.2) can be reformulated into the general form as given in Hou and Patton [149] and Darouach, Zasadzinski and Boutayeb [152]:

$$\boldsymbol{x}_{k+1} = \mathbf{A}_k \boldsymbol{x}_k + \mathbf{E}'_k \boldsymbol{d}'_k + \boldsymbol{w}_k \tag{7.3}$$

$$\boldsymbol{y}_k = \mathbf{H}_k \boldsymbol{x}_k + \mathbf{F}'_k \boldsymbol{d}'_k + \boldsymbol{v}_k \tag{7.4}$$

In this chapter, $\mathbf{E}'_k = [\mathbf{E}_k \ \mathbf{0}]$, $\mathbf{F}'_k = [\mathbf{0} \ \mathbf{F}_k]$. The existence of an unknown-input decoupled filter must satisfy the following existence condition [149, 152]:

$$\text{rank} \begin{bmatrix} \mathbf{F}'_k & \mathbf{H}_k \mathbf{E}'_k \\ \mathbf{0} & \mathbf{F}'_k \end{bmatrix} = \text{rank} \, [\mathbf{F}'_k] + \text{rank} \begin{bmatrix} \mathbf{E}'_k \\ \mathbf{F}'_k \end{bmatrix} \tag{7.5}$$

In our case, since rank $\mathbf{H}_k = m$, the left-hand side of condition (7.5) is $2m$ while the right-hand side is $3m$. Therefore, the above existence condition does not hold, which means that all the unknown-input filters mentioned in the introduction can not be directly implemented.

In this chapter, we consider the consecutive bias fault estimation of a system subjected to disturbances, as described in Equations (7.1) and (7.2). Although the existence condition of designing a traditional unknown input decoupled filter is not satisfied, it will be shown that the unknown inputs can still be decoupled using an extended DMAE approach.

Remark 1. The model described by Equations (7.1) and (7.2) is useful for applications where the disturbances appear in the system equation and the faults appear in the measurement equation, such as bias fault estimation in aircraft air data sensors [92].

7.2.2. THE DMAE APPROACH

The DMAE1 approach proposed in Lu et al. [92] considers the model (7.1) and (7.2) for $\boldsymbol{d}_k = \mathbf{0}$. It is referred to as the DMAE approach in this chapter, which is generalized in the following.

The DMAE [92], which is a modified approach of multiple-model-based approach [34, 37], is composed of two KFs operating in parallel: a no-fault (or fault-free) filter and an augmented fault filter. These two filters are based on two modes of the system: fault-free ($\boldsymbol{f}_k = \mathbf{0}$) and faulty ($\boldsymbol{f}_k \neq \mathbf{0}$). The two filters use the same vector of measurements $\boldsymbol{Y}$ and vector of input u, and are based on the same equations of motion, while each hypothesizes a different fault scenario. The state vector of the no-fault filter $\boldsymbol{x}_{nf}$ and that of the augmented fault filter $\boldsymbol{x}_{af}$ are as follows:

$$\boldsymbol{x}_{nf,k} = \boldsymbol{x}_k, \ \boldsymbol{x}_{af,k} = \begin{bmatrix} \boldsymbol{x}_{nf,k} \\ \boldsymbol{f}_k \end{bmatrix} \tag{7.6}$$

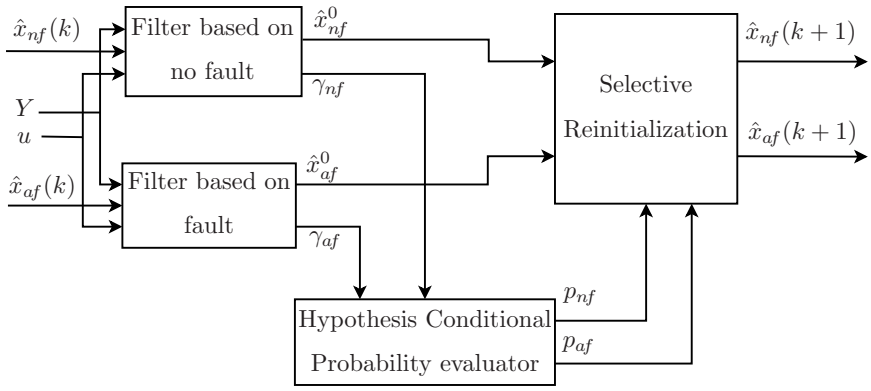


Figure 7.1: Block diagram for the DMAE approach

where "nf" means no fault and "af" means augmented fault. It can be noted that the state vector of the augmented fault filter is the state vector of the no-fault filter with augmentation of the fault vector $\boldsymbol{f}_k$.

At time step k, each of the filters produces a state estimate $\hat{\boldsymbol{x}}^0_i(k)$ and a vector of innovations $\boldsymbol{\gamma}_i(k)$. The principle is that the KF which produces the most well-behaved innovations, contains the model which matches the true faulty model best [34, 37]. The block diagram of the DMAE is given in Figure 7.1.

A hypothesis test uses the innovation $\boldsymbol{\gamma}_i(k)$ and the innovation covariance matrix $\mathbf{C}_i(k)$ of the filters in order to assign a conditional probability to each of the filters. Let a denote the fault scenarios of the system. If we define the hypothesis conditional probability $p_i(k)$ as the probability that a is assigned a_i for $i = 1,2$ ($a_1 = nf$, $a_2 = af$), conditioned on the measurement history up to time step k:

$$p_i(k) = \Pr[a = a_i | \boldsymbol{Y}(k) = \boldsymbol{Y}_k], \quad i = 1,2 \tag{7.7}$$

then the conditional probability of the two filters can be updated recursively using the following equation:

$$p_i(k) = \frac{f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i,\boldsymbol{Y}_{k-1})p_i(k-1)}{\sum_{j=1}^{2} f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_j,\boldsymbol{Y}_{k-1})p_j(k-1)}, \quad i = 1,2 \tag{7.8}$$

where $\boldsymbol{Y}_{k-1}$ is the measurement history vector which is defined as $\boldsymbol{Y}_{k-1} = \{\boldsymbol{y}(1), \boldsymbol{y}(2), .., \boldsymbol{y}(k-1)\}$.
$f_{\boldsymbol{y}_k|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}_k|a_i,\boldsymbol{Y}_{k-1})$ is the probability density function which is given by the following Gaussian form [37]:

$$\begin{aligned} &f_{\boldsymbol{y}(k)|a,\boldsymbol{Y}_{k-1}}(\boldsymbol{y}(k)|a_i,\boldsymbol{Y}_{k-1}) \\ &\quad = \beta_i(k)\exp\{-\boldsymbol{\gamma}_i^T(k)\mathbf{C}_i^{-1}(k)\boldsymbol{\gamma}_i(k)/2\} \end{aligned} \tag{7.9}$$

where

$$\beta_i(k) = \frac{1}{(2\pi)^{m/2}|\mathbf{C}_i(k)|^{1/2}} \tag{7.10}$$

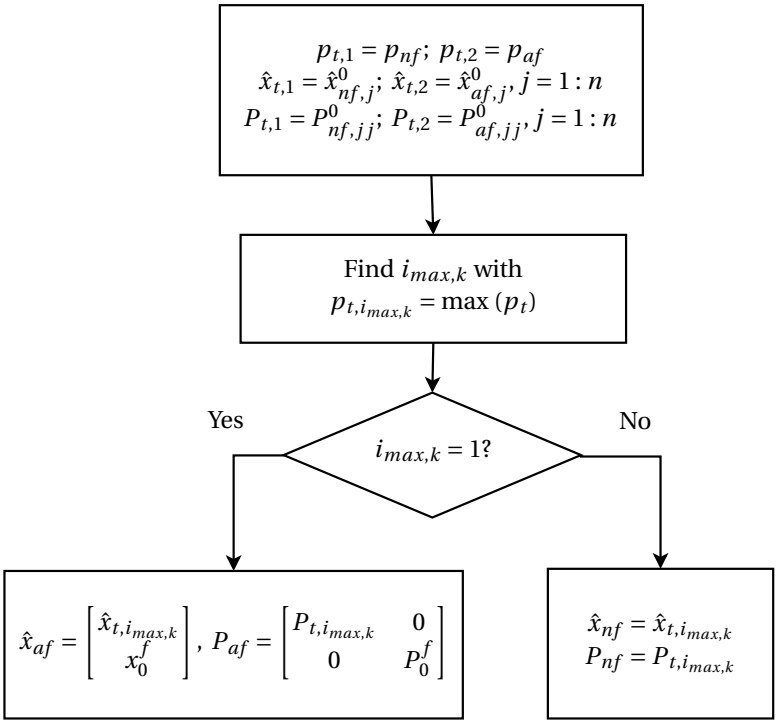


Figure 7.2: Flow chart of the Selective Reinitialization algorithm. Note n refers to the dimension of $\hat{x}_{nf}$.

7

In Equation (7.10), $|\bullet|$ denotes the determinant of the covariance matrix $\mathbf{C}_i(k)$ which is computed by the KF at time step k. The filter which matches the fault scenario produces the smallest innovation which is the difference between the estimated measurement and the true measurement. Therefore, the conditional probability of the filter which matches the true fault scenario is the highest between the two filters. After the computation of the conditional probability, the state estimate of the nonlinear system $\hat{\boldsymbol{x}}(k)$ can be generated by the weighted state estimate $\hat{\boldsymbol{x}}_i(k)$ of the two filters:

$$\begin{aligned}\hat{\boldsymbol{x}}(k) &= \sum_{i=1}^{2} \hat{\boldsymbol{x}}_i(k)\, p_i(k) \\ &= \hat{\boldsymbol{x}}_{nf}(k)\, p_{nf}(k) + \hat{\boldsymbol{x}}_{af}(k)\, p_{af}(k).\end{aligned} \tag{7.11}$$

The fault is only estimated by the augmented fault filter and the estimate is denoted as $\hat{\boldsymbol{f}}(k)$. The probability-weighted fault estimate of the DMAE approach $\bar{\boldsymbol{f}}(k)$ is calculated as follows:

$$\bar{\boldsymbol{f}}(k) = \hat{\boldsymbol{f}}(k)\, p_{af}(k) \tag{7.12}$$

The core of the DMAE approach is selective reinitialization. The flow chart of the selective reinitialization algorithm is presented in Figure 7.2.

In the algorithm, $\hat{\boldsymbol{x}}^0_{nf}$ ($\hat{\boldsymbol{x}}^0_{af}$) and $\hat{\boldsymbol{x}}_{nf}$ ($\hat{\boldsymbol{x}}_{af}$) denote the state estimate of the no-fault

(augmented fault) filter before and after the reinitialization, respectively. $\mathbf{P}^0_{nf}$ ($\mathbf{P}^0_{af}$) and $\mathbf{P}_{nf}$ ($\mathbf{P}_{af}$) denote the covariance of state estimate error of the no-fault (augmented fault) filter before and after the reinitialization, respectively. $\hat{\boldsymbol{x}}_t$, $\boldsymbol{p}_t$ and $\mathbf{P}_t$ are the vectors which contain the state estimate, model probability and the covariance matrix of state estimation error of the no-fault filter and the fault filter respectively. $i_{max,k}$ is the index of the model with the maximum model probability at time step k. $\boldsymbol{x}^f_0$ and $\mathbf{P}^f_0$ are the parameters which are used for the initialization of the fault filter.

7.3. EXTENSION OF THE DMAE APPROACH

The DMAE approach can achieve an unbiased estimation of $\boldsymbol{x}_k$ and $\boldsymbol{f}_k$ when $\boldsymbol{d}_k = 0$ [92]. However, when $\boldsymbol{d}_k \neq 0$, the unknown-input filtering problem becomes more challenging. Since the existence condition (7.5) is no longer satisfied, traditional unknown-input decoupled filters can not be designed.

In this section, the DMAE is extended to the case when $\boldsymbol{d}_k \neq 0$. In order to achieve this, the state vectors of the no-fault filter and augmented fault filter are changed to:

$$\bar{\mathbf{x}}_{nf,k} = \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{d}_k \end{bmatrix}, \bar{\mathbf{x}}_{af,k} = \begin{bmatrix} \bar{\mathbf{x}}_{nf,k} \\ \boldsymbol{f}_k \end{bmatrix} \tag{7.13}$$

where $\bar{\boldsymbol{x}}_{nf,k} \in \mathbb{R}^{n+n_d}$ and $\bar{\boldsymbol{x}}_{af,k} \in \mathbb{R}^{n+n_d+n_f}$. The state vector of the augmented fault filter is that of the no-fault filter augmented with the fault vector. Therefore, the state vector of the no-fault filter can be inferred from that of the augmented fault filter and vice versa.

The random walk process provides a useful and general tool for the modeling of unknown time-varying processes [32, 150, 162]. $\boldsymbol{d}_k$ can be modeled by a random walk process [150, 162] as:

$$\boldsymbol{d}_{k+1} = \boldsymbol{d}_k + \boldsymbol{w}_{d,k}, \tag{7.14}$$

where $\boldsymbol{w}_{d,k}$ is a white noise sequence with covariance: $E\{\boldsymbol{w}_{d,k}(\boldsymbol{w}_{d,l})^T\} = \mathbf{Q}^d_k \delta_{kl}$. $\boldsymbol{f}_k$ is also modeled as a random walk process as:

$$\boldsymbol{f}_{k+1} = \boldsymbol{f}_k + \boldsymbol{w}_{f,k}, \tag{7.15}$$

where $\boldsymbol{w}_{f,k}$ is a white noise sequence with covariance: $E\{\boldsymbol{w}_{f,k}(\boldsymbol{w}_{f,l})^T\} = \mathbf{Q}^f_k \delta_{kl}$. Then, the system model and measurement model of the no-fault filter can be described as follows:

$$\bar{\boldsymbol{x}}_{nf,k+1} = \bar{\mathbf{A}}_{nf,k}\bar{\boldsymbol{x}}_{nf,k} + \bar{\boldsymbol{w}}_{nf,k} \tag{7.16}$$

$$\boldsymbol{y}_k = \bar{\mathbf{H}}_{nf,k}\bar{\boldsymbol{x}}_{nf,k} + \boldsymbol{v}_k \tag{7.17}$$

where

$$\bar{\mathbf{A}}_{nf,k} = \begin{bmatrix} \mathbf{A}_k & \mathbf{E}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \bar{\mathbf{H}}_{nf,k} = [\mathbf{H}_k \; 0], \bar{\boldsymbol{w}}_{nf,k} = \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{w}_{d,k} \end{bmatrix} \tag{7.18}$$

The model of the augmented fault filter is as follows:

$$\bar{\boldsymbol{x}}_{af,k+1} = \bar{\mathbf{A}}_{af,k}\bar{\boldsymbol{x}}_{af,k} + \bar{\boldsymbol{w}}_{af,k} \tag{7.19}$$

$$\boldsymbol{y}_k = \bar{\mathbf{H}}_{af,k}\bar{\boldsymbol{x}}_{af,k} + \boldsymbol{v}_k \tag{7.20}$$

where

$$\bar{\mathbf{A}}_{af,k} = \begin{bmatrix} \bar{\mathbf{A}}_{nf,k} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \bar{\mathbf{H}}_{af,k} = [\bar{\mathbf{H}}_{nf,k}\ \mathbf{F}_k], \bar{\boldsymbol{w}}_{af,k} = \begin{bmatrix} \bar{\boldsymbol{w}}_{nf,k} \\ \boldsymbol{w}_{f,k} \end{bmatrix} \tag{7.21}$$

Since the difference from the DMAE in Lu et al. [92] is the augmentation of $\boldsymbol{d}_k$, only the covariance related to $\boldsymbol{w}_{d,k}$, i.e., $\mathbf{Q}_k^d$ is discussed below. It should be noted that $\mathbf{Q}_k^d$ is usually unknown, the optimality of the filter can be compromised by a poor choice of $\mathbf{Q}_k^d$ [147, 150]. If $\mathbf{Q}_k^d$ is not properly chosen, it can influence the estimation of $\boldsymbol{d}_k$ as well as $\boldsymbol{x}_k$.

This chapter proposes a method to adapt $\mathbf{Q}_k^d$ by making use of the augmented fault filter of the DMAE approach. To compensate the influence of a bad choice of $\mathbf{Q}_k^d$ on the estimation of $\boldsymbol{x}_k$, the system noise vector $\bar{\boldsymbol{w}}_{nf,k}$ in Equations(7.16), (7.18) and (7.21) is modified to:

$$\bar{\boldsymbol{w}}_{nf,k} = \begin{bmatrix} \boldsymbol{w}_k + \mathbf{E}_k\boldsymbol{w}_{d,k} \\ \boldsymbol{w}_{d,k} \end{bmatrix} \tag{7.22}$$

It should be noted that this is only an approximation. However, the simulation results in Section 7.5 demonstrate that this choice is satisfactory.

Let $\hat{\bar{\boldsymbol{x}}}_{af,k-1|k-1}$ denote the unbiased estimate of $\bar{\boldsymbol{x}}_{af,k-1}$ given measurements up to time $k-1$. $\hat{\boldsymbol{x}}_{k-1|k-1}$, $\hat{\boldsymbol{d}}_{k-1|k-1}$ and $\hat{\boldsymbol{f}}_{k-1|k-1}$ denote the estimates of $\boldsymbol{x}_{k-1}$, $\boldsymbol{d}_{k-1}$ and $\boldsymbol{f}_{k-1}$, respectively. The innovation of the augmented fault filter is:

$$\begin{aligned} \boldsymbol{\gamma}_{af,k} &= \boldsymbol{y}_k - \bar{\mathbf{H}}_{af,k}\hat{\bar{\boldsymbol{x}}}_{af,k|k-1} \\ &= \mathbf{H}_k\mathbf{A}_{k-1}\tilde{\boldsymbol{x}}_{k-1|k-1} + \mathbf{H}_k\mathbf{E}_{k-1}\tilde{\boldsymbol{d}}_{k-1|k-1} + \mathbf{F}_k\tilde{\boldsymbol{f}}_{k-1|k-1} \\ &+ \mathbf{H}_k w_{k-1} + \mathbf{H}_k\mathbf{E}_{k-1}w_{d,k-1} + \mathbf{F}_k\boldsymbol{w}_{f,k-1} + \boldsymbol{v}_k \end{aligned} \tag{7.23}$$

with

$$\tilde{\boldsymbol{x}}_{k-1|k-1} := \mathbf{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1|k-1} \tag{7.24}$$

$$\tilde{\boldsymbol{d}}_{k-1|k-1} := \mathbf{d}_{k-1} - \hat{\boldsymbol{d}}_{k-1|k-1} \tag{7.25}$$

$$\tilde{\boldsymbol{f}}_{k-1|k-1} := \mathbf{f}_{k-1} - \hat{\boldsymbol{f}}_{k-1|k-1} \tag{7.26}$$

Therefore, the innovation covariance of the augmented fault filter is:

$$\begin{aligned}\mathbf{C}_{af,k} &= E\{\boldsymbol{\gamma}_{af,k}\boldsymbol{\gamma}_{af,k}^T\}\\ &= \mathbf{H}_k\mathbf{A}_{k-1}\mathbf{P}^x_{k-1|k-1}\mathbf{A}^T_{k-1}\mathbf{H}^T_k\\ &+ \mathbf{H}_k\mathbf{E}_{k-1}\mathbf{P}^d_{k-1|k-1}E^T_{k-1}\mathbf{H}^T_k + \mathbf{F}_k\mathbf{P}^f_{k-1|k-1}\mathbf{F}^T_k\\ &+ \mathbf{H}_k\mathbf{A}_{k-1}\mathbf{P}^{xd}_{k-1|k-1}\mathbf{E}^T_{k-1}\mathbf{H}^T_k + \mathbf{H}_k\mathbf{A}_{k-1}\mathbf{P}^{xf}_{k-1|k-1}\mathbf{F}^T_k\\ &+ \mathbf{H}_k\mathbf{E}_{k-1}\mathbf{P}^{dx}_{k-1|k-1}\mathbf{A}^T_{k-1}\mathbf{H}^T_k + \mathbf{H}_k\mathbf{E}_{k-1}\mathbf{P}^{df}_{k-1|k-1}\mathbf{F}^T_{k-1}\\ &+ \mathbf{F}_{k-1}\mathbf{P}^{fx}_{k-1|k-1}\mathbf{A}^T_{k-1}\mathbf{H}^T_k + \mathbf{F}_k\mathbf{P}^{fd}_{k-1|k-1}\mathbf{E}^T_{k-1}\mathbf{H}^T_k + \mathbf{R}_k\\ &+ \mathbf{H}_k\mathbf{Q}_{k-1}\mathbf{H}^T_k + \mathbf{H}_k\mathbf{E}_{k-1}\mathbf{Q}^d_{k-1}\mathbf{E}^T_{k-1}\mathbf{H}^T_k + \mathbf{F}_kQ^f_k\mathbf{F}^T_k\end{aligned} \tag{7.27}$$

where the covariance matrices are defined as follows:

$$\begin{aligned}&\mathbf{P}^x_{k|k} := E[\tilde{\boldsymbol{x}}_{k|k}\tilde{\boldsymbol{x}}^T_{k|k}],\ \mathbf{P}^d_{k|k} := E[\tilde{\boldsymbol{d}}_{k|k}\tilde{\boldsymbol{d}}^T_{k|k}]\\ &\mathbf{P}^f_{k|k} := E[\tilde{\boldsymbol{f}}_{k|k}\tilde{\boldsymbol{f}}^T_{k|k}],\ \mathbf{P}^{xd}_{k|k} := E[\tilde{\boldsymbol{x}}_{k|k}\tilde{\boldsymbol{d}}^T_{k|k}]\\ &\mathbf{P}^{dx}_{k|k} := E[\tilde{\boldsymbol{d}}_{k|k}\tilde{\boldsymbol{x}}^T_{k|k}],\ \mathbf{P}^{xf}_{k|k} := E[\tilde{\boldsymbol{x}}_{k|k}\tilde{\boldsymbol{f}}^T_{k|k}]\\ &\mathbf{P}^{fx}_{k|k} := E[\tilde{\boldsymbol{f}}_{k|k}\tilde{\boldsymbol{x}}^T_{k|k}],\ \mathbf{P}^{df}_{k|k} := E[\tilde{\boldsymbol{d}}_{k|k}\tilde{\boldsymbol{f}}^T_{k|k}]\\ &\mathbf{P}^{fd}_{k|k} := E[\tilde{\boldsymbol{f}}_{k|k}\tilde{\boldsymbol{d}}^T_{k|k}].\end{aligned}$$

The actual $\mathbf{C}_{af,k}$ is approximated as follows [30, 31]:

$$\hat{\mathbf{C}}_{af,k} = \frac{1}{N}\sum_{j=k-N+1}^{k}\boldsymbol{\gamma}_{af,j}\boldsymbol{\gamma}^T_{af,j} \tag{7.28}$$

$\mathbf{Q}^d_k$ can be approximated by the main diagonal of

$$\mathbf{E}_k^{-1}\mathbf{H}_k^{-1}\tilde{\mathbf{Q}}_k(\mathbf{H}_k^T)^{-1}(\mathbf{E}_k^T)^{-1} \tag{7.29}$$

with $\tilde{\mathbf{Q}}_k$ is a diagonal matrix defined as:

$$\tilde{\mathbf{Q}}_k := \text{diag}(\max\{0,\hat{Q}_{k,11}\},...,\max\{0,\hat{Q}_{k,mm}\}) \tag{7.30}$$

where $\hat{Q}_{k,jj}$, $j = 1,2,...,m$ is the jth diagonal element of $\hat{\mathbf{Q}}_k$ which is denoted as:

$$\hat{\mathbf{Q}}_k = (\hat{\mathbf{C}}_{af,k} - \mathbf{H}_k\mathbf{Q}_{k-1}\mathbf{H}^T_k - \mathbf{F}_k\mathbf{Q}^f_k\mathbf{F}^T_k - \mathbf{R}_k) \tag{7.31}$$

The restriction $\tilde{Q}_{k,jj} \geq 0$, $j = 1,2,...,m$ in Equation (7.30) is to preserve the properties of a variance [163].

7.4. UNKNOWN INPUT DECOUPLED FILTERING

This section proves that the unknown input decoupled filtering can be achieved using the extended DMAE approach which does not need to satisfy the existence condition

(7.5). Let l ($l \geq 1$) denote the time step when the first fault occurs and l_e denote the time step when the first fault is removed, which means $\boldsymbol{f}_k = 0$ when $k < l$ and $\boldsymbol{f}_k \neq 0$ when $l \leq k \leq l_e$. Without loss of generality, it will be proven that $\boldsymbol{f}_k$ can be estimated when $k \leq l_e$.

7.4.1. UNKNOWN INPUT ESTIMATION DURING $k < l$

Theorem 2. *During $k < l$, an unbiased estimation of $\boldsymbol{d}_k$ can be achieved using the fault-free filter of the extended DMAE approach.*

Proof. When $k < l$, $\boldsymbol{f}_k = 0$. The fault-free model matches the true fault scenario while the augmented fault filter does not. Therefore, according to the DMAE approach, $i_{max,k} = 1$ during this time period.

The system model during this period is as follows:

$$\boldsymbol{x}_{k+1} = \mathbf{A}_k \boldsymbol{x}_k + \mathbf{E}_k \boldsymbol{d}_k + \boldsymbol{w}_k \tag{7.32}$$

$$\boldsymbol{y}_k = \mathbf{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k \tag{7.33}$$

Under this situation, $\boldsymbol{d}_k$ can be estimated using the fault-free filter whose convergence condition will be discussed later. □ □

The estimation of $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ when $l \leq k \leq l_e$ will be discussed in the following.

7.4.2. UNKNOWN INPUT ESTIMATION AT $k = l$

For the sake of readability, the subscript *af* will be discarded for the remainder of the section. All the variables with a bar on top in the remainder of this section refer to the augmented fault filter.

Using the DMAE approach, the Kalman gain $\bar{\mathbf{K}}_l$ can be partitioned as follows:

$$\bar{\mathbf{K}}_l = \begin{bmatrix} \mathbf{K}_l^x \\ \mathbf{K}_l^d \\ \mathbf{K}_l^f \end{bmatrix} \tag{7.34}$$

where $\mathbf{K}_l^x$, $\mathbf{K}_l^d$ and $\mathbf{K}_l^f$ are the Kalman gains associated with $\boldsymbol{x}_k$, $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$, respectively.

Lemma 1. *Let $\hat{\boldsymbol{x}}_{l-1|l-1}$ and $\hat{\boldsymbol{d}}_{l-1|l-1}$ be unbiased, if $\boldsymbol{x}_0^f$ is chosen to be $\mathbf{0}$ or sufficiently small, then $\boldsymbol{f}_l$ can be estimated using the augmented fault filter if and only if $\mathbf{K}_l^f$ satisfies*

$$\mathbf{K}_l^f \mathbf{F}_l = \mathbf{I}. \tag{7.35}$$

Proof. The innovation of the augmented filter is

$$\bar{\boldsymbol{\gamma}}_l = \boldsymbol{e}_l + \mathbf{F}_l \boldsymbol{f}_l \tag{7.36}$$

where $\boldsymbol{e}_l$ is defined as

$$\begin{aligned} \boldsymbol{e}_l := {} & \mathbf{H}_l \mathbf{A}_{l-1} \tilde{\boldsymbol{x}}_{l-1|l-1} + \mathbf{H}_l \mathbf{E}_{l-1} \tilde{\boldsymbol{d}}_{l-1|l-1} \\ & + \mathbf{H}_l \boldsymbol{w}_{l-1} + \mathbf{H}_l \mathbf{E}_{l-1} \boldsymbol{w}_{d,l-1} + \boldsymbol{v}_l \end{aligned} \tag{7.37}$$

Since $\hat{\boldsymbol{x}}_{l-1|l-1}$ and $\hat{\boldsymbol{d}}_{l-1|l-1}$ are unbiased (this can be achieved by the DMAE1 in Lu et. al [92] since $\boldsymbol{f}_k = 0$ when $k < l$), $E[\boldsymbol{e}_l] = 0$.

Consequently, the expectation of $\bar{\boldsymbol{\gamma}}_l$ is:

$$E[\bar{\boldsymbol{\gamma}}_l] = \mathbf{F}_l \boldsymbol{f}_l. \tag{7.38}$$

The estimation of the fault can be given by

$$\begin{aligned} \hat{\boldsymbol{f}}_{l|l} &= \hat{\boldsymbol{f}}_{l|l-1} + \mathbf{K}_l^f \bar{\boldsymbol{\gamma}}_l \\ &= \hat{\boldsymbol{f}}_{l-1|l-1} + \mathbf{K}_l^f \bar{\boldsymbol{\gamma}}_l \end{aligned} \tag{7.39}$$

Since $i_{max,k} = 1$ when $k < l$, according to the flow chart of the selective reinitialization algorithm given in Figure 7.2, Equation (7.39) can be further written into

$$\hat{\boldsymbol{f}}_{l|l} = \boldsymbol{x}_0^f + \mathbf{K}_l^f \bar{\boldsymbol{\gamma}}_l \tag{7.40}$$

Substituting (7.36) into (7.40), yields

$$\hat{\boldsymbol{f}}_{l|l} = \mathbf{K}_l^f \mathbf{F}_l \boldsymbol{f}_l + \mathbf{K}_l^f \boldsymbol{e}_l \tag{7.41}$$

Consequently, the expectation of $\hat{\boldsymbol{f}}_{l|l}$

$$E[\hat{\boldsymbol{f}}_{l|l}] = E[\mathbf{K}_l^f \mathbf{F}_l \boldsymbol{f}_l]. \tag{7.42}$$

Therefore, it can concluded that $\boldsymbol{f}_l$ can be estimated if and only if $\mathbf{K}_l^f$ satisfies

$$\mathbf{K}_l^f \mathbf{F}_l = \mathbf{I}. \qquad \square \tag{7.43}$$

$\square$

Theorem 3. *Let $\hat{\boldsymbol{x}}_{l-1|l-1}$ and $\hat{\boldsymbol{d}}_{l-1|l-1}$ be unbiased, then $\boldsymbol{f}_l$ can be estimated using the augmented fault filter of the DMAE approach by choosing a sufficiently large $\mathbf{P}_0^f$ and a sufficiently small $\boldsymbol{x}_0^f$.*

Proof. Define the following covariance matrix:

$$\bar{\mathbf{P}}_{l-1|l-1} := E[\tilde{\bar{\boldsymbol{x}}}_{l-1|l-1}\, \tilde{\bar{\boldsymbol{x}}}_{l-1|l-1}^T]$$

where $\tilde{\bar{\boldsymbol{x}}}_{l-1|l-1} = \bar{\boldsymbol{x}}_{l-1} - \hat{\bar{\boldsymbol{x}}}_{l-1|l-1}$.

Due to the selective reinitialization algorithm given in Figure 7.2, $\mathbf{P}^f_{l-1|l-1} = \mathbf{P}_0^f$. Therefore, the covariance of the state prediction error $\bar{\mathbf{P}}_{l|l-1}$ can be computed and partitioned

as follows:

$$\begin{aligned}\bar{\mathbf{P}}_{l|l-1} &= \bar{\mathbf{A}}_{l-1}\begin{bmatrix}\mathbf{P}^{x}_{l-1|l-1} & \mathbf{P}^{xd}_{l-1|l-1} & \mathbf{0}\\ \mathbf{P}^{dx}_{l-1|l-1} & \mathbf{P}^{d}_{l-1|l-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{P}^{f}_{0}\end{bmatrix}\bar{\mathbf{A}}^{T}_{l-1}\\ &\quad+\begin{bmatrix}\mathbf{Q}_{l-1}+\mathbf{E}_{l-1}\mathbf{Q}^{d}_{l-1}\mathbf{E}^{T}_{l-1} & \mathbf{E}_{l-1}Q^{d}_{l-1} & \mathbf{0}\\ Q^{d}_{l-1}\mathbf{E}^{T}_{l-1} & \mathbf{Q}^{d}_{l-1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{Q}^{f}_{l-1}\end{bmatrix} \qquad (7.44)\\ &= \begin{bmatrix}\mathbf{P}^{x}_{l|l-1} & \mathbf{P}^{xd}_{l|l-1} & 0\\ \mathbf{P}^{dx}_{l|l-1} & \mathbf{P}^{d}_{l|l-1} & 0\\ 0 & 0 & \mathbf{P}^{f}_{l|l-1}\end{bmatrix} \qquad (7.45)\end{aligned}$$

where

$$\begin{aligned}\mathbf{P}^{x}_{l|l-1} &:= \mathbf{A}_{l-1}\mathbf{P}^{x}_{l-1|l-1}\mathbf{A}^{T}_{l-1}+\mathbf{E}_{l-1}\mathbf{P}^{d}_{l-1|l-1}\mathbf{E}^{T}_{l-1}\\ &\quad+\mathbf{A}_{l-1}\mathbf{P}^{xd}_{l-1|l-1}\mathbf{E}^{T}_{l-1}+\mathbf{E}_{l-1}\mathbf{P}^{dx}_{l-1|l-1}\mathbf{A}^{T}_{l-1}\\ &\quad+\mathbf{Q}_{l-1}+\mathbf{E}_{l-1}\mathbf{Q}^{d}_{l-1}\mathbf{E}^{T}_{l-1}\\ \mathbf{P}^{d}_{l|l-1} &:= \mathbf{P}^{d}_{l-1|l-1}+\mathbf{Q}^{d}_{l-1}\\ \mathbf{P}^{xd}_{l|l-1} &:= \mathbf{A}_{l-1}\mathbf{P}^{xd}_{l-1|l-1}+\mathbf{E}_{l-1}\mathbf{P}^{d}_{l-1|l-1}+\mathbf{E}_{l-1}Q^{d}_{l-1}\\ \mathbf{P}^{dx}_{l|l-1} &:= \mathbf{P}^{dx}_{l-1|l-1}\mathbf{A}^{T}_{l-1}+\mathbf{P}^{d}_{l-1|l-1}\mathbf{E}^{T}_{l-1}+Q^{d}_{l-1}\mathbf{E}^{T}_{l-1}\\ \mathbf{P}^{f}_{l|l-1} &:= \mathbf{P}^{f}_{0}+\mathbf{Q}^{f}_{l-1}\end{aligned}$$

Define

$$\bar{\mathbf{C}}_{l} := \bar{\mathbf{H}}_{l}\bar{\mathbf{P}}_{l|l-1}\bar{\mathbf{H}}^{T}_{l}+\mathbf{R}_{l}. \qquad (7.46)$$

Substituting Equation (7.45) into the above equation, it follows that

$$\bar{\mathbf{C}}_{l} = \mathbf{H}_{l}\mathbf{P}^{x}_{l|l-1}\mathbf{H}^{T}_{l}+\mathbf{F}_{l}\mathbf{P}^{f}_{l|l-1}\mathbf{F}^{T}_{l}+\mathbf{R}_{l} \qquad (7.47)$$

Consequently, the Kalman gain of the augmented filter can be calculated and partitioned as follows:

$$\begin{aligned}\bar{\mathbf{K}}_{l} &= \bar{\mathbf{P}}_{l|l-1}\bar{\mathbf{H}}^{T}_{l}\bar{\mathbf{C}}^{-1}_{l}\\ &= \begin{bmatrix}\mathbf{P}^{x}_{l|l-1}\mathbf{H}^{T}_{l}\\ \mathbf{P}^{dx}_{l|l-1}\mathbf{H}^{T}_{l}\\ \mathbf{P}^{f}_{l|l-1}\mathbf{F}^{T}_{l}\end{bmatrix}\bar{\mathbf{C}}^{-1}_{l} \qquad (7.48)\end{aligned}$$

If $\mathbf{P}_0^f$ is chosen sufficiently large, then $\bar{\mathbf{C}}_l \approx \mathbf{F}_l \mathbf{P}_0^f \mathbf{F}_l^T$. It follows that

$$\bar{\mathbf{K}}_l = \begin{bmatrix} \mathbf{P}_{l|l-1}^x \mathbf{H}_l^T \bar{\mathbf{C}}_l^{-1} \\ \mathbf{P}_{l|l-1}^{dx} \mathbf{H}_l^T \bar{\mathbf{C}}_l^{-1} \\ \mathbf{F}_l^{-1} \end{bmatrix} \tag{7.49}$$

Therefore, $\mathbf{K}_l^f = \mathbf{F}_l^{-1}$. It follows from Lemma 1 that $\boldsymbol{f}_l$ can be estimated. □ □

7.4.3. UNKNOWN INPUT ESTIMATION DURING $l < k \leq l_e$

Theorem 4. *Provided that $\boldsymbol{f}_k$ has been estimated at $k = l$, $\boldsymbol{d}_k$ can be estimated using the augmented fault filter of the extended DMAE approach.*

Proof. During this period, the augmented fault model matches the true fault scenario. Therefore, $i_{max,k} = 2$, which means that the fault-free filter is reinitialized by the fault filter during this period. Since this chapter considers bias fault, $\boldsymbol{f}_k$ is constant for $l < k \leq l_e$. Therefore, during this period, we can set:

$$\begin{aligned} \hat{\bar{\boldsymbol{x}}}_{k|k-1} &= \begin{bmatrix} \hat{\boldsymbol{x}}_{k|k-1}^* \\ \hat{\boldsymbol{f}}_{l|l} \end{bmatrix}, \bar{\mathbf{P}}_{k|k-1} = \begin{bmatrix} \mathbf{P}_{k|k-1}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{l|l}^f \end{bmatrix} \\ \hat{\bar{\boldsymbol{x}}}_{k|k} &= \begin{bmatrix} \hat{\boldsymbol{x}}_{k|k}^* \\ \hat{\boldsymbol{f}}_{l|l} \end{bmatrix}, \bar{\mathbf{P}}_{k|k} = \begin{bmatrix} \mathbf{P}_{k|k}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{l|l}^f \end{bmatrix}, \bar{\mathbf{K}}_k = \begin{bmatrix} \mathbf{K}_k^* \\ \mathbf{0} \end{bmatrix}, \end{aligned} \tag{7.50}$$

where

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k-1}^* &:= \begin{bmatrix} \hat{\boldsymbol{x}}_{k|k-1} \\ \hat{\boldsymbol{d}}_{k|k-1} \end{bmatrix}, \mathbf{P}_{k|k-1}^* := \begin{bmatrix} \mathbf{P}_{k|k-1}^x & \mathbf{P}_{k|k-1}^{xd} \\ \mathbf{P}_{k|k-1}^{dx} & \mathbf{P}_{k|k-1}^d \end{bmatrix}, \\ \hat{\boldsymbol{x}}_{k|k}^* &:= \begin{bmatrix} \hat{\boldsymbol{x}}_{k|k} \\ \hat{\boldsymbol{d}}_{k|k} \end{bmatrix}, \mathbf{P}_{k|k}^* := \begin{bmatrix} \mathbf{P}_{k|k}^x & \mathbf{P}_{k|k}^{xd} \\ \mathbf{P}_{k|k}^{dx} & \mathbf{P}_{k|k}^d \end{bmatrix}, \mathbf{K}_k^* := \begin{bmatrix} \mathbf{K}_k^{\boldsymbol{x}} \\ \mathbf{K}_k^{\boldsymbol{d}} \end{bmatrix} \end{aligned} \tag{7.51}$$

are updated by the normal Kalman filtering procedure. It can be seen that during this period, the estimation of the fault and the covariance are:

$$\hat{\boldsymbol{f}}_{k|k} = \hat{\boldsymbol{f}}_{l|l},\ \mathbf{P}_{k|k}^f = \mathbf{P}_{l|l}^f,\ l < k \leq l_e \tag{7.52}$$

It can be inferred that the model of the fault filter is equivalent to:

$$\boldsymbol{x}_{k+1} = \mathbf{A}_k \boldsymbol{x}_k + \mathbf{E}_k \boldsymbol{d}_k + \boldsymbol{w}_k \tag{7.53}$$

$$\boldsymbol{y}_k = \mathbf{H}_k \boldsymbol{x}_k + \mathbf{F}_k \hat{\boldsymbol{f}}_{l|l} + \boldsymbol{v}_k \tag{7.54}$$

As can be seen, the only unknown input is $\boldsymbol{d}_k$ since the fault filter treats $\boldsymbol{f}_k$ as a known input during this period. Since a known input does not affect the design of a filter [149], the convergence condition of this fault filter is the same as that of the fault-free filter based on Equations (7.32) and (7.33).

Therefore, $\boldsymbol{d}_k$ can be estimated using the augmented fault filter under the same condition as for the model described by Equations (7.32) and (7.33). □ □

7.4.4. ERROR ANALYSIS

In the previous sections, it is assumed that $\hat{\boldsymbol{x}}_{l-1|l-1}$ and $\hat{\boldsymbol{d}}_{l-1|l-1}$ are unbiased. We analyze the estimation error of $\boldsymbol{f}_l$ when $\hat{\boldsymbol{x}}_{l-1|l-1}$ and $\hat{\boldsymbol{d}}_{l-1|l-1}$ are biased.

Through Equation (7.43), Equation (7.41) can be further rewritten into

$$\hat{\boldsymbol{f}}_{l|l} = \boldsymbol{f}_l + \mathbf{F}_l^{-1}\boldsymbol{e}_l \tag{7.55}$$

Substitute Equation (7.37) into Equation (7.55), it follows

$$\begin{aligned}\hat{\boldsymbol{f}}_{l|l} = \boldsymbol{f}_l + \mathbf{F}_l^{-1}(&\mathbf{H}_l\mathbf{A}_{l-1}\tilde{\boldsymbol{x}}_{l-1|l-1} + \mathbf{H}_l\mathbf{E}_{l-1}\tilde{\boldsymbol{d}}_{l-1|l-1} \\ &+ \mathbf{H}_l w_{l-1} + \mathbf{H}_l\mathbf{E}_{l-1}\boldsymbol{w}_{d,l-1} + \boldsymbol{v}_l)\end{aligned} \tag{7.56}$$

The estimation error of $\boldsymbol{f}_l$ as a function of $\tilde{\boldsymbol{x}}_{l-1|l-1}$ and $\tilde{\boldsymbol{d}}_{l-1|l-1}$ can be obtained as follows:

$$\tilde{\boldsymbol{f}}_{l|l} = \boldsymbol{f}_l - \hat{\boldsymbol{f}}_{l|l} \tag{7.57}$$

$$\begin{aligned}&= \mathbf{F}_l^{-1}(\mathbf{H}_l\mathbf{A}_{l-1}\tilde{\boldsymbol{x}}_{l-1|l-1} + \mathbf{H}_l\mathbf{E}_{l-1}\tilde{\boldsymbol{d}}_{l-1|l-1} \\ &+ \mathbf{H}_l\boldsymbol{w}_{l-1} + \mathbf{H}_l\mathbf{E}_{l-1}\boldsymbol{w}_{d,l-1} + \boldsymbol{v}_l)\end{aligned} \tag{7.58}$$

If $\tilde{\boldsymbol{x}}_{l-1|l-1}$ and $\tilde{\boldsymbol{d}}_{l-1|l-1}$ are unbiased, the expectation of $\tilde{\boldsymbol{f}}_{l|l}$ is zero, which means the fault estimate is unbiased. If $\tilde{\boldsymbol{x}}_{l-1|l-1}$ and $\tilde{\boldsymbol{d}}_{l-1|l-1}$ are biased, assume

$$\underline{a}\mathbf{I} \le \mathbf{A}_{l-1} \le \bar{a}\mathbf{I}, \quad \underline{f}\mathbf{I} \le \mathbf{F}_{l-1} \le \bar{f}\mathbf{I} \tag{7.59}$$

$$\underline{h}\mathbf{I} \le \mathbf{H}_{l-1} \le \bar{h}\mathbf{I}, \quad \underline{e}\mathbf{I} \le \mathbf{E}_{l-1} \le \bar{e}\mathbf{I} \tag{7.60}$$

$$\underline{e}_x\mathbf{I} \le \tilde{\boldsymbol{x}}_{l-1|l-1} \le \bar{e}_x\mathbf{I}, \quad \underline{e}_d\mathbf{I} \le \tilde{\boldsymbol{d}}_{l-1|l-1} \le \bar{e}_d\mathbf{I} \tag{7.61}$$

$$\underline{w}\mathbf{I} \le \boldsymbol{w}_{l-1} \le \bar{w}\mathbf{I}, \quad \underline{w}_d\mathbf{I} \le \boldsymbol{w}_{d,l-1} \le \bar{w}_d I, \tag{7.62}$$

$$\underline{v}\mathbf{I} \le \boldsymbol{v}_{l-1} \le \bar{v}\mathbf{I}. \tag{7.63}$$

Then it follows that the fault estimation error is bounded by the following:

$$\left[\frac{\underline{h}(\underline{a}\underline{e}_x + \underline{e}\underline{e}_d + \underline{w} + \underline{e}\underline{w}_d) + \underline{v}}{\bar{f}}, \frac{\bar{h}(\bar{a}\bar{e}_x + \bar{e}\bar{e}_d + \bar{w} + \bar{e}\bar{w}_d) + \bar{v}}{\underline{f}}\right] \tag{7.64}$$

7.4.5. DISCUSSION

The approach proposed in this section uses a double-model framework, instead of the traditional unknown input filtering framework, to decouple the unknown inputs. Therefore, the existence condition given in Equation (7.5) is not required.

For the model given in Equations (7.32) and (7.33), the convergence condition for time-invariant case has been given by Darouach et al. [164], which is given as follows:

$$\operatorname{rank}\begin{bmatrix} z\mathbf{I} - \mathbf{A} & -\mathbf{E} \\ \mathbf{H} & \mathbf{0} \end{bmatrix} = n + n_d, \forall z \in \mathscr{C}, |z| \ge 1 \tag{7.65}$$

This convergence condition is also required by traditional unknown input decoupled filters such as those in Darouach, Zasadzinski and Boutayeb [152] and Cheng et al. [154].

The system considered up until now in this chapter is linear and the noise is assumed to be Gaussian. If the system is nonlinear, the DMAE can be extended using Unscented Kalman Filters [27, 92]. If the system noise is non-Gaussian, then it should be extended by making use of particle filters [28, 157, 158]. However, this is out of the scope of the present chapter.

7.5. ILLUSTRATIVE EXAMPLES WITH COMPARISON TO EXISTING METHODS

In this section, two examples similar to that in [162], [152] and [146] are provided to demonstrate the performance of the extended DMAE approach.

The system is described by model (7.1) and (7.2) where

$$\mathbf{A} = \begin{bmatrix} -0.0005 & -0.0084 \\ 0.0517 & 0.8069 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0.1815 \\ 1.7902 \end{bmatrix}, \tag{7.66}$$

$$\mathbf{E} = \begin{bmatrix} 0.629 & 0 \\ 0 & -0.52504 \end{bmatrix}, \mathbf{H} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathbf{F} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{7.67}$$

$$\mathbf{Q} = \begin{bmatrix} 0.002^2 & 0 \\ 0 & 0.002^2 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0.01^2 & 0 \\ 0 & 0.01^2 \end{bmatrix} \tag{7.68}$$

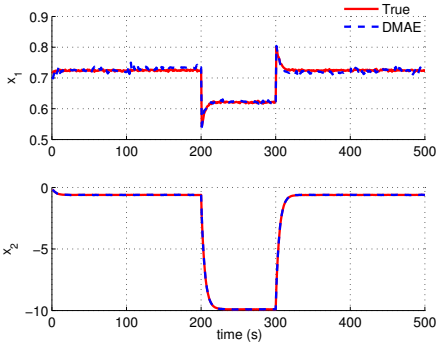
The input u_k is: $u_k = -0.5$ when $200 < k \leq 300$, otherwise $u_k = 0.5$. $\boldsymbol{f}_k$ is given by the red solid lines in Figure 7.3(c). It can be noted that the number of unknown inputs in [162], [152] and [146] is n_d ($n_d = 2$) while this chapter deals with $2n_d$ unknown inputs.

In both examples, since $\mathbf{E}'_k = [\mathbf{E}_k \ \mathbf{0}]$, $\mathbf{F}'_k = [\mathbf{0} \ \mathbf{F}_k]$, condition (7.5) is not satisfied. In addition, rank $\boldsymbol{y}_k <$ rank $\boldsymbol{d}'_k$. Consequently, all the unknown input decoupled filters in the introduction are not applicable to solve the problem, except for special cases when $\boldsymbol{d}_k = \mathbf{0}$ or $\boldsymbol{f}_k = \mathbf{0}$. N in Equation (7.28) is set to be 10. In both examples, $\mathbf{Q}_k^f = \mathbf{0}$, $\mathbf{Q}_k^d$ is updated by the main diagonal of the matrix given in (7.29), $\boldsymbol{x}_0^f = [10^{-3}, 10^{-3}]^T, \mathbf{P}_0^f = 10^2\mathbf{I}$.

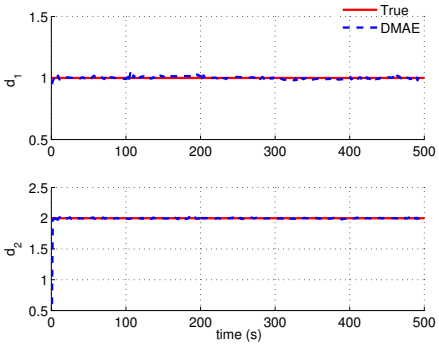
Example 1. In this example, $\boldsymbol{d}_k$ is a constant bias vector, which is shown by the red solid lines in Figure 7.3(b). The condition (7.5) is not satisfied. Therefore, traditional unknown input filters, which require the satisfaction of condition (7.5), can not be implemented.

The extended DMAE approach is implemented. The true and estimated p_{nf} and p_{af} using the extended DMAE approach are well matched. The probability-weighted estimates of $\boldsymbol{x}_k$, $\boldsymbol{d}_k$, which are calculated using Equation (7.11), are shown in Figures 7.3(a) and 7.3(b), respectively. The probability-weighted estimate of $\boldsymbol{f}_k$ (calculated using Equation (7.12)) is shown in Figure 7.3(c). As can be seen, $\boldsymbol{x}_k$, $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ can all be estimated.

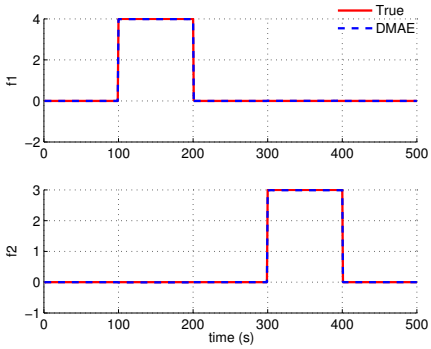
Example 2. In this example, the disturbances, which are taken from [64], are stochas-



(a) True and estimated states, example 1



(b) True and estimated disturbances, example 1



(c) True and estimated faults, example 1

Figure 7.3: Results of the DMAE approach, example 1

Table 7.1: RMSEs of the fault and disturbance estimation for Example 2

	Methods	d_1	d_2	f_1	f_2
Case 1	RTSKF [84]	-	-	0.0103	0.0102
	PF [28, 157]	-	-	0.1549	0.1496
	DMAE	-	-	0.0060	0.0047
Case 2	OTSKF [145]	0.0697	0.1442	-	-
	PF [28, 157]	0.1088	0.2035	-	-
	DMAE	0.0709	0.1459	-	-
Case 3	[28, 84, 145, 150, 153, 157]	N/A	N/A	N/A	N/A
	DMAE	0.0845	0.1655	0.0230	0.0283

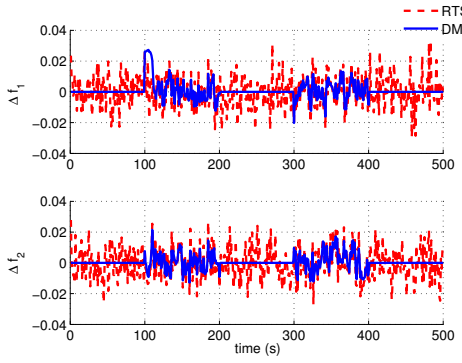


Figure 7.4: Errors of estimation of f_1 and f_2 using the RTS-KF and the DMAE approach, case 1, example 2

tic. $\boldsymbol{d}_k = \begin{bmatrix} d_{1,k} \\ d_{2,k} \end{bmatrix}$ is generated using the following model [64]:

$$\begin{bmatrix} d_{i,k} \\ d'_{i,k} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{V^2}{L_{gi}^2} & -2\frac{V}{L_{gi}} \end{bmatrix} \begin{bmatrix} d_{i,k-1} \\ d'_{i,k-1} \end{bmatrix} + \begin{bmatrix} \sigma_i \sqrt{\frac{3V}{L_{gi}}} \\ (1-2\sqrt{3})\sigma_i \sqrt{(\frac{V}{L_{gi}})^3} \end{bmatrix} \boldsymbol{w}'_{d,k},\ i = 1,2 \tag{7.69}$$

where $V = 35$, $\sigma_1 = 0.5$, $\sigma_2 = 0.8$, $L_{g1} = 2500$, $L_{g2} = 1500$ and $\boldsymbol{w}'_{d,k} \sim N(0,1)$. The generated $\boldsymbol{d}_k$ is shown by the red solid lines in Figure 7.5(b). It should be noted that the DMAE approach still models $\boldsymbol{d}_k$ as a random walk process since $\boldsymbol{d}_k$ is treated as an unknown input.

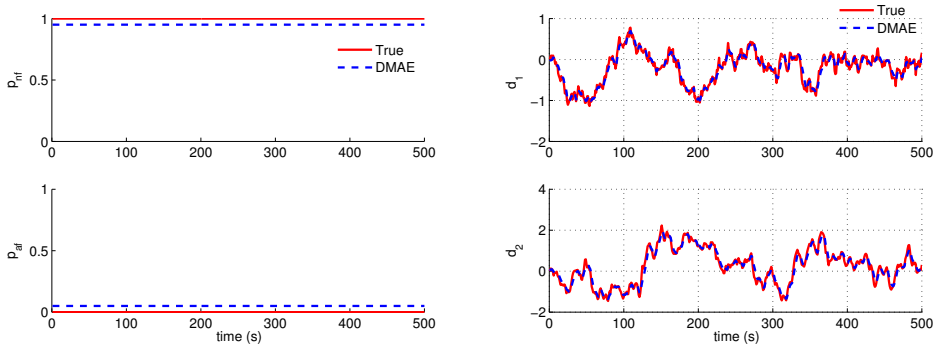
Three cases are considered for this example. The first two cases are special cases. In these two cases, the existence condition (7.5) is satisfied. Therefore, some of the approaches mentioned in the introduction can still be used.

Case 1. $\boldsymbol{d}_k = \boldsymbol{0}$, $\boldsymbol{f}_k \neq \boldsymbol{0}$

In this case, $\mathbf{E}_k$ is a zero matrix. Therefore, condition (7.5) is satisfied. The probability-weighted estimate of $\boldsymbol{f}_k$ using the extended DMAE is the same as in Figure 7.3(c). The RTS-KF in Gillijns and De Moor [84] is also applied and the errors of estimation of $\boldsymbol{f}_k$ compared to the DMAE are shown in Figure 7.4. In addition, particle filters [28, 157] are also applied. The model used for estimation of f_k is also the random walk. 100 particles are used. The RMSEs of estimation of f_1 and f_2 using the RTS-KF, the particle filter [28, 157] and the extended DMAE are shown in Table 7.1.

Case 2. $\boldsymbol{d}_k \neq \boldsymbol{0}$, $\boldsymbol{f}_k = \boldsymbol{0}$

In this case, $\mathbf{F}_k$ is a zero matrix. Therefore, condition (7.5) is also satisfied. The true and estimated p_{nf} and p_{af} using the extended DMAE approach are shown in Fig-



(a) True and estimated model probabilities, case 2, example 2

(b) True and estimated disturbances, case 2, example 2

Figure 7.5: Results of the DMAE approach, case 2, example 2

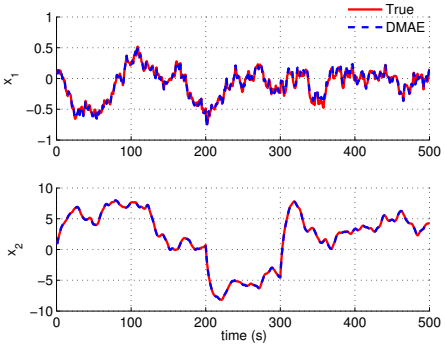


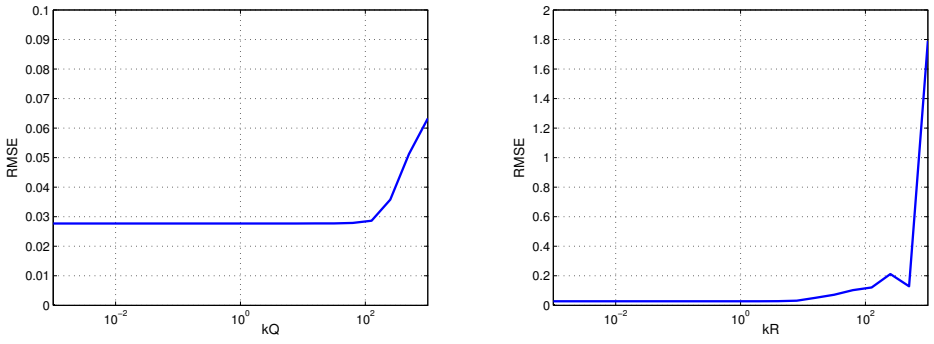
Figure 7.6: True and estimated states, case 3, example 2

ure 7.5(a). The probability-weighted estimate of $\boldsymbol{d}_k$ is presented in Figure 7.5(b). The results using the methods in Heish [150], Heish and Chen [145], and Gillijns and De Moor [153], are similar to that of the DMAE. Particle filter is also applied. The model used for estimation of d_k is the random walk. The RMSEs of estimation of d_1 and d_2 using the OTSKF in Heish [145], the particle filter [28, 157] and the extended DMAE are shown in Table 7.1.

Case 3. $\boldsymbol{d}_k \neq \mathbf{0}$, $\boldsymbol{f}_k \neq \mathbf{0}$

In this case, condition (7.5) is not satisfied. Thus, all the conventional filters mentioned in the introduction are not applicable.

The true and estimated p_{nf} and p_{af} using the extended DMAE approach are also well matches. The probability-weighted estimates of $\boldsymbol{x}_k$, is shown in Figure 7.6. The probability-weighted estimates of $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ are the same as in Figures 7.5(b) and 7.3(c) respectively. It can be seen that despite the fact that the existence condition for tradi-



(a) Sensitivity with respect to $\mathbf{Q}_k$ errors, case 3, example 2

(b) Sensitivity with respect to $\mathbf{R}_k$ errors, case 3, example 2

Figure 7.7: Sensitivity of the fault estimation using the DMAE approach with respect to $\mathbf{Q}_k$ and $\mathbf{R}_k$ errors, case 3, example 2

tional unknown-input decoupled filters is not satisfied, $\boldsymbol{x}_k$, $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ can all be estimated using the extended DMAE approach. The RMSEs of the estimation of $\boldsymbol{d}_k$ and $\boldsymbol{f}_k$ using the extended DMAE approach are shown in Table 7.1.

Finally, the sensitivity of the DMAE with respect to errors in $\mathbf{Q}_k$ and $\mathbf{R}_k$ is discussed. To demonstrate the sensitivity with respect to errors in $\mathbf{Q}_k$, $\mathbf{R}_k$ is fixed and $\mathbf{Q}_k$ is multiplied with a coefficient k_Q. The sensitivity result of the RMSE of fault estimation with k_Q ranging from 10^{-3} to 10^3 is shown in Figure 7.7(a). To show the sensitivity with respect to $\mathbf{R}_k$ errors, $\mathbf{Q}_k$ is fixed and $\mathbf{R}_k$ is multiplied with a coefficient k_R. The sensitivity result of the RMSE of fault estimation with k_R ranging from 10^{-3} to 10^3 is shown in Figure 7.7(b).

It can be seen from Figures 7.7(a) and 7.7(b) that the minimum RMSEs are obtained when $k_Q = 1$ or $k_R = 1$. However, it is also noted that the extended DMAE approach is more sensitive to $\mathbf{R}_k$ errors. The RMSE of the fault estimation increases to 0.063 when $\mathbf{Q}_k$ is multiplied with 10^3 and increases to 1.79 when $\mathbf{R}_k$ is multiplied with 10^3. This is expected since in section 7.3, the process noise $\bar{\boldsymbol{w}}_{nf,k}$ is adapted while the output noise v_k is not adapted. Therefore, selection of $\mathbf{R}_k$ should be performed with more caution.

7.6. APPLICATION TO THE ADS FDD USING SIMULATED ATMOSPHERIC TURBULENCE

In this section, the extended DMAE is applied to the ADS bias FDD in the presence of atmospheric turbulence. The process model for the ADS FDD, which can be derived

using Appendix B, is as follows:

$$\begin{aligned}\dot{V} &= (A_x - g\sin\theta)\cos\alpha\cos\beta + (A_y + g\cos\theta\sin\phi)\sin\beta \\ &+ (A_z + g\cos\theta\cos\phi)\sin\alpha\cos\beta + (r v_w - q w_w - \dot{u}_w)\cos\alpha\cos\beta \\ &+ (p w_w - r u_w - \dot{v}_w)\sin\beta + (q u_w - p v_w - \dot{w}_w)\sin\alpha\cos\beta\end{aligned} \tag{7.70}$$

$$\begin{aligned}\dot{\alpha} &= \frac{1}{V\cos\beta}(-A_x\sin\alpha + A_z\cos\alpha + g\cos\phi\cos\theta\cos\alpha + g\sin\theta\sin\alpha) \\ &+ q - (p\cos\alpha + r\sin\alpha)\tan\beta \\ &+ \frac{1}{V\cos\beta}[(q u_w - p v_w - \dot{w}_w)\cos\alpha - (r v_w - q w_w - \dot{u}_w)\sin\alpha]\end{aligned} \tag{7.71}$$

$$\begin{aligned}\dot{\beta} &= \frac{1}{V}[-(A_x - g\sin\theta)\cos\alpha\sin\beta + (A_y + g\sin\phi\cos\theta)\cos\beta \\ &- (A_z + g\cos\phi\cos\theta)\sin\alpha\sin\beta] + p\sin\alpha - r\cos\alpha \\ &+ \frac{1}{V}[-(r v_w - q w_w - \dot{u}_w)\cos\alpha\sin\beta + (p w_w - r u_w - \dot{v}_w)\cos\beta \\ &- (q u_w - p v_w - \dot{w}_w)\sin\alpha\sin\beta]\end{aligned} \tag{7.72}$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{7.73}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{7.74}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{7.75}$$

The measurement model including the ADS faults is:

$$V_m = V + f_V + v_V \tag{7.76}$$

$$\alpha_m = \alpha + f_\alpha + v_\alpha \tag{7.77}$$

$$\beta_m = \beta + f_\beta + v_\beta \tag{7.78}$$

$$\phi_m = \phi + v_\phi \tag{7.79}$$

$$\theta_m = \theta + v_\theta \tag{7.80}$$

$$\psi_m = \psi + v_\psi \tag{7.81}$$

The simulated turbulence is generated using the Dryden model. For readability, the power spectral density functions of the Dryden turbulence model are repeated as follows [64]:

$$\Phi_u(\omega_o) = \frac{2\sigma_u^2 L_u}{\pi V}\frac{1}{1 + (L_u\omega_o/V)^2} \tag{7.82}$$

$$\Phi_v(\omega_o) = \frac{\sigma_v^2 L_v}{\pi V}\frac{1 + 3(L_v\omega_o/V)^2}{1 + (L_v\omega_o/V)^2} \tag{7.83}$$

$$\Phi_w(\omega_o) = \frac{\sigma_w^2 L_w}{\pi V}\frac{1 + 3(L_w\omega_o/V)^2}{1 + (L_w\omega_o/V)^2} \tag{7.84}$$

where V is the true airspeed, ω_o is the observed angular frequency. L_u, L_v, L_w are the scale lengths and σ_u, σ_v, σ_w are the intensities of the turbulence. To demonstrate the

Table 7.2: Scenarios with different

Variables	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Units
L_u, L_v, L_w	150	530	530	530	[m]
$\sigma_u, \sigma_v, \sigma_w$	0.25	0.25	2	6.4	[m/s]

Table 7.3: Fault scenario of simultaneous faults

Time interval	Fault	Fault magnitude	Fault unit
$10\text{ s} < t < 20\text{ s}$	f_V	3	[m/s]
	f_α	$2\pi/180$	[rad]
	f_β	$2\pi/180$	[rad]
$30\text{ s} < t < 40\text{ s}$	f_V	-3	[m/s]
	f_α	$-2\pi/180$	[rad]
	f_β	$-2\pi/180$	[rad]

performance of the proposed approach, the approach is validated using difference scenarios with different L_u, L_v, L_w and σ_u, σ_v, σ_w. The values are listed in Table 7.2. The difference between scenarios 1 and 2 is that the scale lengths of scenario 2 is larger. The difference between scenarios 2, 3 and 4 is that the intensity increases. In scenario 4, the scale lengths and intensities are selected based on the thunderstorm given in [64].

The corresponding speed of the turbulence of different scenarios are shown in Figure 7.8. The influence of scale lengths can be observed from Figures 7.8(a) and 7.8(b). The influence of intensities can be observed from Figures 7.8(b), 7.8(c) and 7.8(d). The maximum magnitude of u_w in scenario 4 is more than 20 times that in scenario 2.

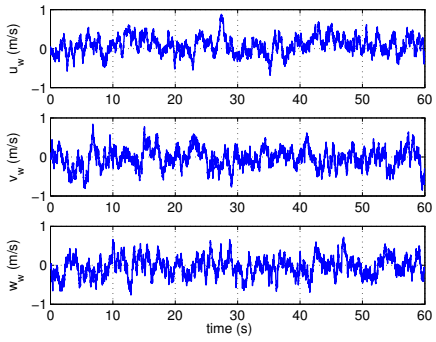
The considered fault scenario is simultaneous fault given in Table 7.3.

7.6.1. RESULTS USING THE DMAE WITHOUT EXTENSION

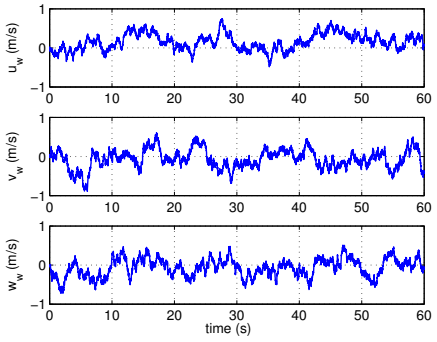
For comparison, in this section, the results of the DMAE without extension are shown first.

True and estimated model probabilities using the DMAE without extension in the first two scenarios are shown in Figures 7.9(a) and 7.9(b). It is seen that there are some false alarms at around $t = 27$ s. The model probabilities using the DMAE without extension for scenario 3 are shown in Figures 7.9(c). It is seen that the probabilities are correct when there are faults ($10\text{ s} < t < 20\text{ s}$ and $30\text{ s} < t < 40\text{ s}$). However, when there are no faults, many false alarms are observed. In scenario 4, when the turbulence intensities are increased, the model probabilities are completely wrong, as seen from Figure 7.9(d).

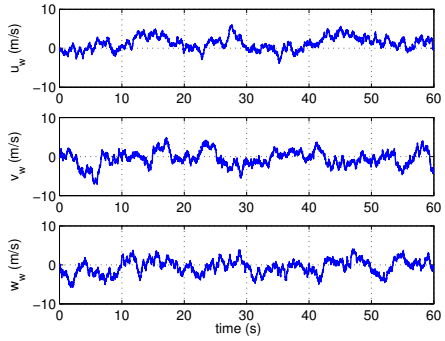
True and estimated f_V, f_α and f_β using the DMAE *without* extension for scenarios 1 and 2 are shown in Figures 7.10(a) and 7.10(b). It is seen that the fault estimation performance is acceptable. However, when the intensities are increased, the results are much



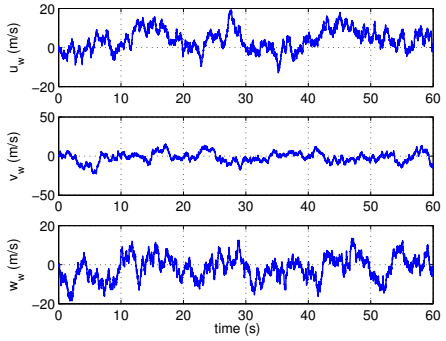
(a) Speed components u_w, v_w and w_w of the turbulence in scenario 1.



(b) Speed components u_w, v_w and w_w of the turbulence in scenario 2.

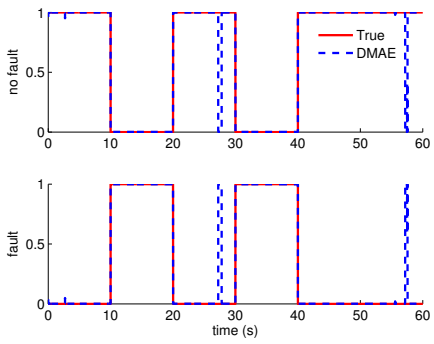


(c) Speed components u_w, v_w and w_w of the turbulence in scenario 3.

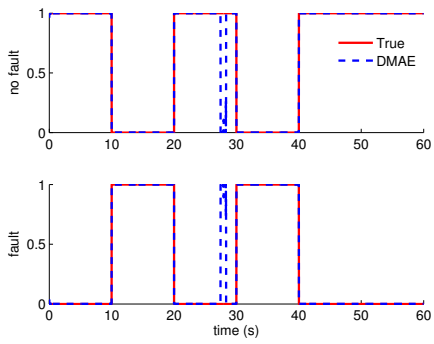


(d) Speed components u_w, v_w and w_w of the turbulence in scenario 4.

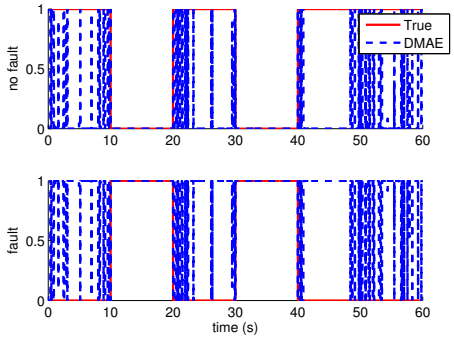
Figure 7.8: Speed components u_w, v_w and w_w of the turbulence in different scenarios.



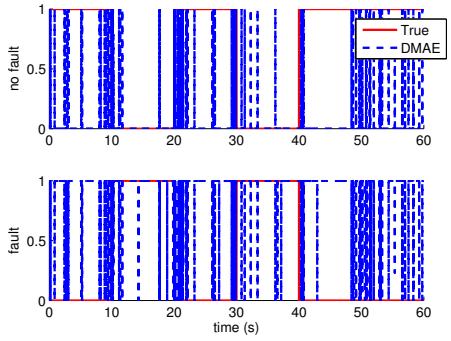
(a) True and estimated model probabilities in scenario 1.



(b) True and estimated model probabilities in scenario 2.

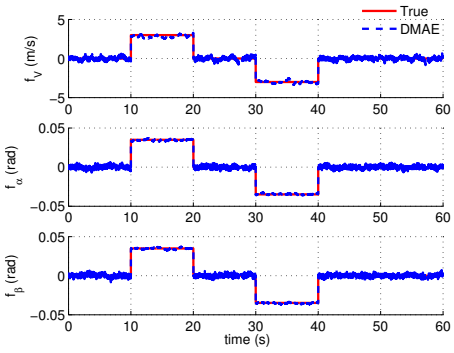


(c) True and estimated model probabilities in scenario 3.

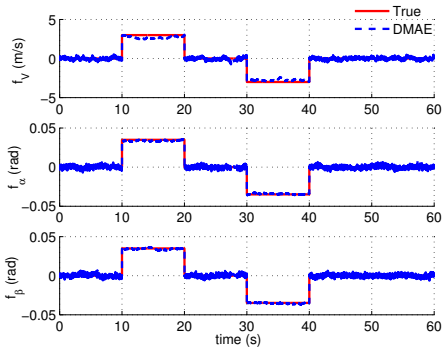


(d) True and estimated model probabilities in scenario 4.

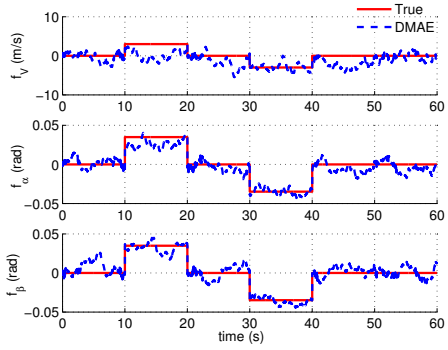
Figure 7.9: True and estimated model probabilities using the DMAE *without* extension.



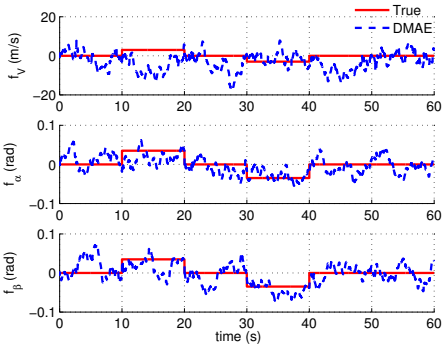
(a) True and estimated f_V, f_α and f_β in scenario 1.



(b) True and estimated f_V, f_α and f_β in scenario 2.

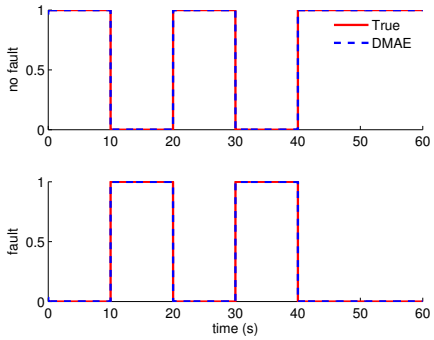


(c) True and estimated f_V, f_α and f_β in scenario 3.

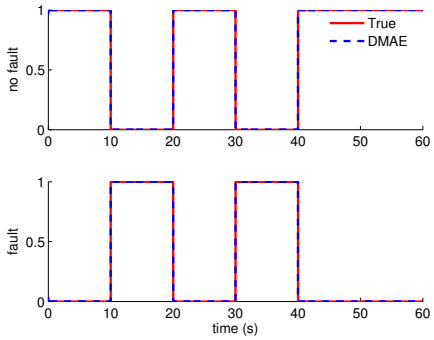


(d) True and estimated f_V, f_α and f_β in scenario 4.

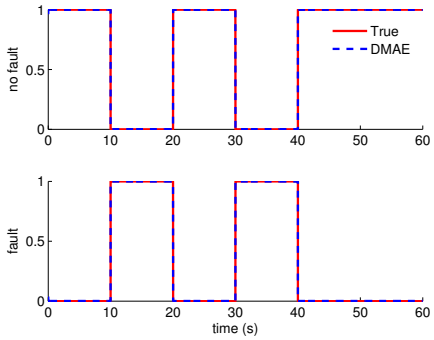
Figure 7.10: True and estimated f_V, f_α and f_β using the DMAE *without* extension



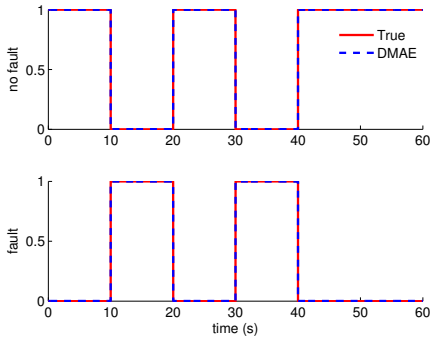
(a) True and estimated model probabilities in scenario 1.



(b) True and estimated model probabilities in scenario 2.



(c) True and estimated model probabilities in scenario 3.



(d) True and estimated model probabilities in scenario 4.

Figure 7.11: True and estimated model probabilities using the DMAE *with* extension.

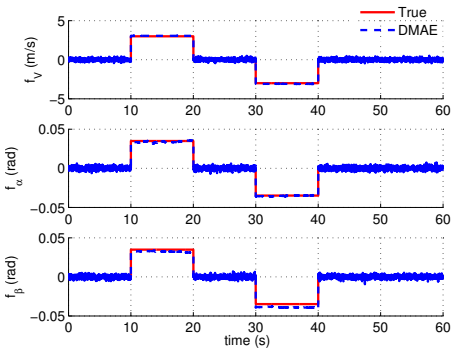
worse. As seen from Figures 7.10(c) and 7.10(d), the fault estimation for scenarios 3 and 4 is significantly influenced by the turbulence.

7.6.2. RESULTS USING THE DMAE WITH EXTENSION

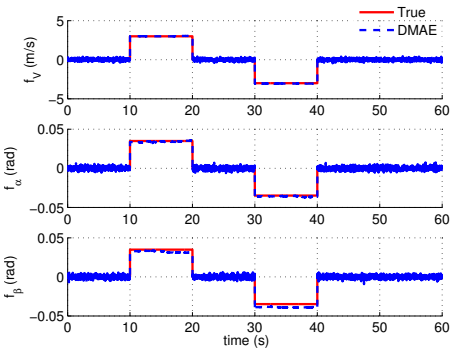
In contrast, true and estimated model probabilities in four scenarios using the DMAE with extension are shown in Figure 7.11. The performance is still performance despite the intensities and scale lengths are different. No false alarms are observed even in scenario 4. This demonstrates the improvement of using the DMAE with extension.

True and estimated f_V, f_α and f_β in four scenarios using the DMAE with extension are shown Figure 7.12. All the faults are estimated in four scenarios. It is also noticed that the variances of the fault estimation, when there are no faults, increase when the intensities increase.

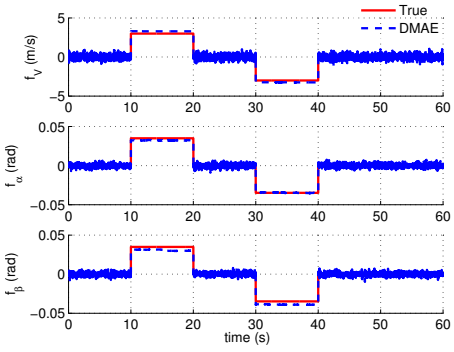
The weighted fault estimates in scenarios 3 and 4 using the DMAE with extension are



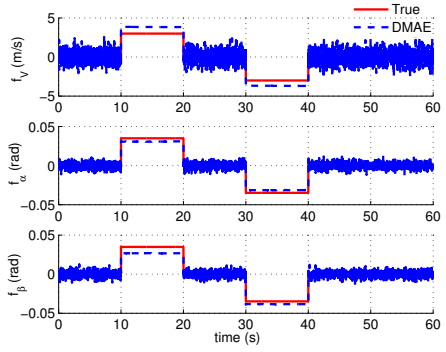
(a) True and estimated f_V, f_α and f_β in scenario 1.



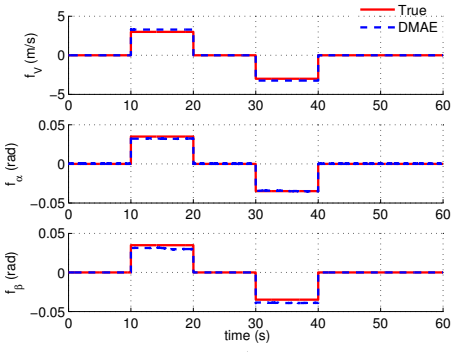
(b) True and estimated f_V, f_α and f_β in scenario 2.



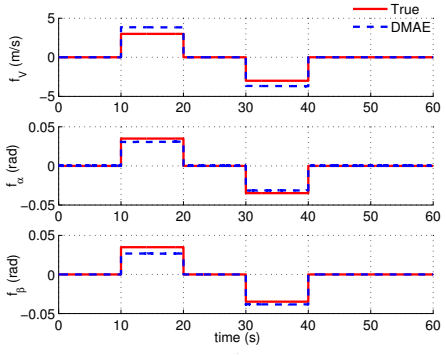
(c) True and estimated f_V, f_α and f_β in scenario 3.



(d) True and estimated f_V, f_α and f_β in scenario 4.

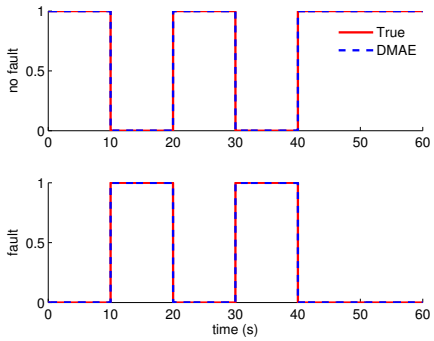


(e) True f_V, f_α and f_β and weighted estimates in scenario 3.

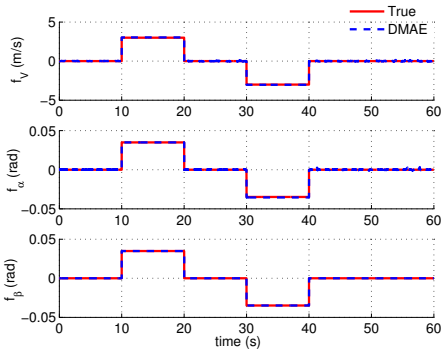


(f) True f_V, f_α and f_β and weighted estimates in scenario 4.

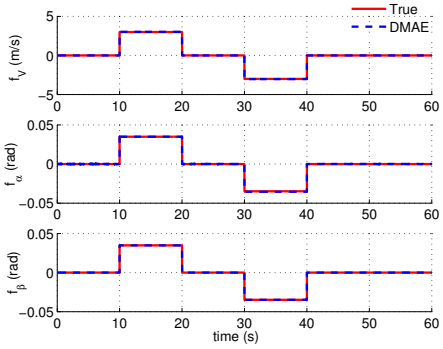
Figure 7.12: True and estimated f_V, f_α and f_β using the DMAE *with* extension



(a) True and estimated model probabilities using real flight data.



(b) True and estimated f_V, f_α and f_β using real flight data.



(c) True f_V, f_α and f_β and weighted estimates using real flight data.

Figure 7.13: Results of using the DMAE *with* extension applied to real flight data recorded during the presence of turbulence.

shown in Figures 7.12(e) and 7.12(f), respectively. Comparing Figure 7.10(c) with Figure 7.12(e), it is found weighted fault estimates approach zero when there are no faults. This is due to the multiplication with the model probabilities shown in Figure 7.9(c). It is noticed in Figure 7.12(f) that the fault estimates are not zero. This is acceptable based on the fact that the turbulence intensities are selected based on the those of the thunderstorm.

7.6.3. APPLICATION TO ADS BIAS FDD USING REAL FLIGHT TEST DATA

In this section, the performance of the proposed DMAE with extension is further validated by real flight data recorded in the presence of turbulence. The flight test data are the same as used in [135]. The fault scenario is the same as in Table 7.3.

The model probabilities using the DMAE with extension in the presence of turbu-

lence are shown in Figure 7.13(a). The performance is similar to those shown in Figure 7.11. No false alarms are observed. This demonstrates that the DMAE with extension will not give false alarms even in the presence of turbulence.

The fault estimates and weighted fault estimates are shown in Figures 7.13(b) and 7.13(c), respectively. All the faults are estimated in an unbiased sense. Therefore, its satisfactory fault detection and diagnosis performance in the presence of turbulence is also validated using real flight data.

7.7. CONCLUSION

In this chapter, the unknown input decoupling problem is extended to the case when the existence condition of traditional unknown input filters is not satisfied. It is proved that the states, disturbances and faults can be estimated using an extended DMAE approach which does not require the existence condition. Therefore, it can be applied to a wider class of systems and applications. Two illustrative examples demonstrate the effectiveness of the extended DMAE approach.

The proposed approach is applied to the ADS FDD in the presence of turbulence. The performance is validated using both aircraft response data in the presence of simulated and real turbulence. The results are satisfactory even when the scale length and intensities of the turbulence change.

III

Actuator Fault-Tolerant Control

8

AIRCRAFT FAULT-TOLERANT TRAJECTORY CONTROL USING INCREMENTAL NONLINEAR DYNAMIC INVERSION

Part I and Part II mainly deal with sensor FDD and FTC. In this part, the flight control in the presence of actuator faults is considered. Aircraft trajectory control in the presence of actuator faults is first addressed in this chapter. The main difficulties of the controller design lie in the design of the flight path and angular rate loop since there are model uncertainties in these two loops. First, the detailed design procedure for the trajectory control is presented. Then, the performance is validated using three scenarios and is compared to other existing approaches such as the Nonlinear Dynamic Inversion with online model identification.

Parts of this chapter are based on:

P Lu, E van Kampen, C C de Visser, Q P Chu. Nonlinear Aircraft Fault-Tolerant Trajectory Control Using Incremental Nonlinear Dynamic Inversion. *Journal of Guidance, Control and Dynamics*, (under review).

This chapter deals with aircraft trajectory control in the presence of model uncertainties and actuator faults. Existing approaches, such as adaptive backstepping and nonlinear dynamic inversion with online model identification, can be applied. However, since there are a number of unknown aerodynamic derivatives, the tuning of parameter update law gains is time-consuming. Methods with online model identification require excitation and the selection of a threshold. Furthermore, to deal with highly nonlinear aircraft dynamics, the aerodynamic model structure needs to be designed. In this chapter, a novel aircraft trajectory controller, which uses the incremental nonlinear dynamic inversion, is proposed to achieve fault-tolerant trajectory control. The design of four control loops is given. The idea is to design the loops with uncertainties using the incremental approach. By doing this, the tuning of the approach is straightforward and there is no requirement for excitation and selection of a threshold. The performance of the proposed trajectory controller using the Incremental Nonlinear Dynamic Inversion (INDI) is compared with existing approaches using three scenarios. The results show that the proposed trajectory controller can follow the reference even when there are model uncertainties and actuator faults.

8.1. INTRODUCTION

Civil aircraft are usually required to follow trajectories in three-dimensional space, such as those imposed by air traffic control [165]. In the design phase of civil aircraft, safety is of critical concern. Many techniques [11, 14, 40, 43, 46, 48, 57, 70, 138, 166] have been proposed to improve the safety level and reduce critical risks. Conventionally, since the aircraft model is nonlinear, flight control systems are designed based on a number of linearized models around certain operating points [40]. Next, a gain scheduling method has to be used to blend the gains in different operating points into one controller.

Using nonlinear control approaches, design of different operating points can be avoided. Linearization-based methods, such as NDI and BS, are nonlinear control methods which can handle nonlinearities in the model. This chapter also makes use of nonlinear control methods to design a trajectory controller. Aircraft trajectory control has been considered by several researchers [165, 167–172]. To design the trajectory controller, the uncertainties of aerodynamic derivatives have to be considered since they can degrade the performance of the nonlinear control approaches. [167] uses a sliding mode adaptive controller to reduce the influence of uncertainties. In [168], a control Lyapunov function approach is applied. In [169], a Command Filtered Backstepping approach which uses adaptive function approximation is applied to design the trajectory controller. [172] deals with model uncertainties using the NDI method. The influence of model uncertainties is decreased by making use of a concurrent learning approach [172]. [170] applies ABS [60, 173] with parameter update laws. However, the computational load of the ABS is intensive and the tuning of the parameter update law gains is time-consuming [63]. In [171, 174], NDI is used to deal with the nonlinearities and the model uncertainties influence is reduced by identifying the unknown parameters online. The method is validated on the SIMONA (SImulation, MOtion and NAvigation) research simulator [171]. However, this method is based on parameter identification and it requires excitation [171] which could limit its performance when there is no excitation. Furthermore, to deal with highly nonlinear aircraft dynamics, an aerodynamic model structure needs

to be designed [175].

The present chapter proposes a novel nonlinear controller for aircraft trajectory control. The NDI-based approach is used to deal with nonlinearities in the model. The solution to cope with uncertain aerodynamic derivatives is to make use of INDI [176, 177]. A control structure with four loops are designed: position control, flight path control, attitude and angular rate control. Through analysis, it is found that there are only model uncertainties in the flight path and angular rate control loops. Therefore, these two control loops are designed based on the INDI control law while the remaining two control loops are designed based on the NDI control law. An additional benefit of using the proposed approach is that there is no need to design the aerodynamic model structure. The overall control architecture of the trajectory controller and the detailed design are presented in the chapter.

The performance of our approach is compared to the approach proposed in [171]. The model-based approach in [171] requires sufficient excitation, which may be difficult to obtain during failure situations. The performance comparison is perform using three scenarios: no fault, model uncertainties, and actuator faults. All three scenarios demonstrate the performance of the proposed trajectory controller.

The faults considered in this chapter only include actuator faults. For FTC in the presence of sensor faults, the reader is referred to [13, 25, 92, 135, 137].

The structure of this chapter is as follows: Section 8.2 presents the aircraft model which is used for designing the trajectory controller. In Section 8.3, the four control loops and the distribution of model uncertainties are introduced. The detailed control law design is given in Section 8.4. The position control and attitude control loops are designed based on the NDI control approach while the flight path control and angular rate control loops are designed based on the INDI control approach. In Section 8.5, the aircraft model and fault scenario are presented. The baseline controller, Nonlinear Dynamic Inversion with online Model Identification (NDI-MI), is also presented. In Section 8.6, the performance of the proposed trajectory controller is compared to existing method which is the NDI with online model identification. Their performances are compared under three scenarios. Finally, the conclusions are given in Section 8.7.

8.2. AIRCRAFT EQUATIONS OF MOTION

In this section, the aircraft model used for designing the control law is described. It is assumed that the Earth is flat and non-rotating. Under this condition, the Earth-center Earth-fixed reference frame is equal to the Earth-fixed inertial reference frame. The body reference frame ($Ox_B y_B z_B$), Earth-center Earth-fixed reference frame ($O_E x_E y_E z_E$), stability reference frame ($Ox_S y_S z_S$) and velocity reference frame ($Ox_V y_V z_V$) are shown in Fig. 8.1.

Define the inertial position vector of the aircraft as follows:

$$\mathbf{x}_0 = [x, y, z]^T \tag{8.1}$$

where x, y and z denote the position of the aircraft in the North, East and down direc-

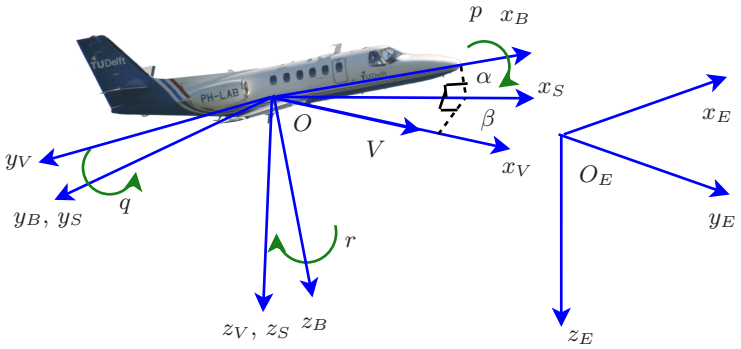


Figure 8.1: Aircraft reference frames. Picture by courtesy of J. A. Pascoe.

tions. The derivatives of the position vector are:

$$\dot{\mathbf{x}}_0 = \begin{bmatrix} V_g \cos\chi \cos\gamma \\ V_g \sin\chi \cos\gamma \\ -V_g \sin\gamma \end{bmatrix} \tag{8.2}$$

Define:

$$\mathbf{x}_1 = [V_g, \chi, \gamma]^T \tag{8.3}$$

where V_g is the inertial velocity of the aircraft, χ is the kinematic azimuth angle and γ is the flight path angle. The dynamics of $\mathbf{x}_1$ are [178]:

$$\dot{\mathbf{x}}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{V_g \cos\gamma} & 0 \\ 0 & 0 & -1/V_g \end{bmatrix} \mathbf{T_{VB}} \frac{\mathbf{F}_B}{m} \tag{8.4}$$

$$= \begin{bmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{mV_g \cos\gamma} & 0 \\ 0 & 0 & \frac{-1}{mV_g} \end{bmatrix} \left(\mathbf{T_{VB}} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \mathbf{T_{VE}} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \right) \tag{8.5}$$

where $\mathbf{F}_B$ represent the total forces (including aerodynamic, propulsion and gravitational forces) acting on the aircraft expressed in the body reference frame. X, Y and Z denote the sum of aerodynamic and propulsion forces expressed in the body axes. $\mathbf{T_{VB}}$ is the transformation matrix from the body frame to the velocity frame and $\mathbf{T_{VE}}$ is the transformation matrix from the Earth frame to the velocity frame. These matrices

are given as:

$$\mathbf{T_{VB}} = \begin{bmatrix} \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ -\cos\alpha\sin\beta\cos\mu + \sin\alpha\sin\mu & \cos\beta\cos\mu & -\sin\alpha\sin\beta\cos\mu - \cos\alpha\sin\mu \\ -\cos\alpha\sin\beta\sin\mu - \sin\alpha\cos\mu & \cos\beta\sin\mu & -\sin\alpha\sin\beta\sin\mu + \cos\alpha\cos\mu \end{bmatrix} \tag{8.6}$$

$$\mathbf{T_{VE}} = \begin{bmatrix} \cos\chi\cos\gamma & \sin\chi\cos\gamma & -\sin\gamma \\ -\sin\chi & \cos\chi & 0 \\ \cos\chi\sin\gamma & \sin\chi\sin\gamma & \cos\gamma \end{bmatrix} \tag{8.7}$$

The attitude angles are defined as:

$$\mathbf{x}_2 = [\mu, \alpha, \beta]^T \tag{8.8}$$

where μ is the kinematic bank angle. α and β are the aerodynamic angle of attack and angle of sideslip, respectively. Their kinematics are given as follows [178]:

$$\dot{\mathbf{x}}_2 = \begin{bmatrix} \cos\alpha\cos\beta & 0 & \sin\alpha \\ \sin\beta & 1 & 0 \\ \sin\alpha\cos\beta & 0 & -\cos\alpha \end{bmatrix}^{-1} \left(-\mathbf{T_{BV}} \begin{bmatrix} -\dot{\chi}\sin\gamma \\ \dot{\gamma} \\ \dot{\chi}\cos\gamma \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \tag{8.9}$$

where $\mathbf{T_{BV}} = \mathbf{T_{VB}}^T$. It should be noted that μ is not directly measured onboard. It is integrated from Equation (8.9).

Further define

$$\mathbf{x}_3 = \boldsymbol{\omega} = [p, q, r]^T \tag{8.10}$$

The dynamics of the angular rates of the aircraft are presented as follows [178]:

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}(\mathbf{M} - \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega}) \tag{8.11}$$

where $\mathbf{J}$ is the inertia matrix defined as

$$\mathbf{J} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{8.12}$$

$\mathbf{M}$ are the total moments acting on the aircraft which can be split into two parts:

$$\mathbf{M} = \mathbf{M}_a + \mathbf{M}_u = \mathbf{M}_a + \mathbf{C_{M_u}}\mathbf{u} \tag{8.13}$$

where $\mathbf{M}_u$ are the moments generated by the control surface deflections and $\mathbf{M}_a$ are all moments except for $\mathbf{M}_u$. $\mathbf{C_{M_u}}$ are the coefficients related to $\mathbf{M}_u$, which are denoted as

$$\mathbf{C_{M_u}} = \bar{q}S\mathbf{C}_1 \begin{bmatrix} C_{l_{\delta a}} & 0 & C_{l_{\delta r}} \\ 0 & C_{m_{\delta e}} & 0 \\ C_{n_{\delta a}} & 0 & C_{n_{\delta r}} \end{bmatrix}, \mathbf{C}_1 = \mathrm{diag}(b, \bar{c}, b) \tag{8.14}$$

where $\bar{q}$ is the dynamic pressure, S is the wing area, b the wing span and $\bar{c}$ is the mean aerodynamic chord. The control surface deflections are $\mathbf{u} = [\delta_a \ \ \delta_e \ \ \delta_r]^T$, which are also the input to the system.

Equation (8.11) can be rewritten into the following affine-in-control form:

$$\dot{\mathbf{x}}_3 = \mathbf{f}_3 + \mathbf{G}_3\mathbf{u} \tag{8.15}$$

where $\mathbf{f}_3 = \mathbf{J}^{-1}(\mathbf{M}_a - \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega})$, $\mathbf{G}_3 = \mathbf{J}^{-1}\mathbf{C}_{\mathbf{M_u}}$.

8.3. CONTROL LOOPS AND UNCERTAINTY SOURCES

In this section, the control loops and the relationships among them are given. Then, the uncertainty sources are presented.

For the ease of explanation, define the following variables:

$$\mathbf{x}_0^{ref} = [x^{ref}, y^{ref}, z^{ref}]^T, \ \mathbf{x}_1^{des} = [V_g^{ref}, \chi^{des}, \gamma^{des}]^T \tag{8.16}$$

$$\mathbf{x}_2^{'des} = [PLA^{des}, \mu^{des}, \alpha^{des}]^T, \ \mathbf{x}_2^{des} = [\mu^{des}, \alpha^{des}, \beta^{ref}]^T \tag{8.17}$$

$$\mathbf{x}_3^{des} = [p^{des}, q^{des}, r^{des}]^T, \ \mathbf{u}^{des} = [\delta_a^{des}, \delta_e^{des}, \delta_r^{des}]^T \tag{8.18}$$

where superscript ref and des both denote the reference commands for corresponding variables. The difference between them is that ref denotes the reference commands which are given for the controller to follow and des denotes the desired reference commands generated by the controller.

The control law design is based on the following four loops:

1. Position control loop
   The objective of this control loop is to follow the trajectory reference $\mathbf{x}_0^{ref}$. The desired references χ^{des} and γ^{des} are generated in this loop.

2. Flight path control loop
   The objective of this loop is to steer $\mathbf{x}_1$ to the given references $\mathbf{x}_1^{des}$. It generates the desired commands μ^{des} and α^{des} for the attitude control loop. In addition, to follow the velocity reference V^{des}, it generates the desired power level angle PLA^{des} for the engine.

3. Attitude control loop
   The objective of this control loop is to follow $\mathbf{x}_2^{des}$ given by the flight path control loop. To control the attitude, it provides the desired command $\mathbf{x}_3^{des}$ for the angular rate control loop.

4. Angular rate control loop
   This loop follows $\mathbf{x}_3^{des}$ given by the attitude control loop. To achieve the objective, it provides the desired command $\mathbf{u}^{des}$ to the actuators such that the control goal can be achieved.

To compute the control law, the aerodynamic forces and moments need to be calculated. The aerodynamic forces acting on the aircraft expressed in the body axis, denoted

by $\mathbf{F}_B$, change the flight path, as described in Equation (8.4). The aerodynamic moments generated by the control surfaces (ailerons, elevators and rudders), modify the angular rates, as described in Equation (8.11). The attitude control loop contains no model uncertainties, as seen in Equation (8.9).

The aerodynamic forces can be expressed by Taylor series [64]. For instance, X can be expanded as follows (higher-order terms are neglected for simplicity of explanation):

$$X = \bar{q}SC_X = \bar{q}S(C_{X_0} + C_{X_\alpha}\alpha + C_{X_q}\frac{q\bar{c}}{V} + C_{X_{\delta_e}}\delta_e + C_{X_{PLA}}PLA) \tag{8.19}$$

where δ_e is the deflection angle of the elevator and PLA represents power level angle. C_{X_α} and C_{X_q} are stability derivatives. $C_{X_{\delta_e}}$ and $C_{X_{PLA}}$ are control derivatives. These derivatives can vary according to various flight conditions and are therefore not constant. To obtain these parameters, computer fluid dynamics or wind tunnel tests are usually performed. However, accurate parameters in various flight situations are difficult to obtain, which leads to model uncertainties. Likewise, the aerodynamic moments can also be expressed in Taylor series. M can be expressed as

$$M = \bar{q}SC_M = \bar{q}S(C_{M_0} + C_{M_\alpha}\alpha + C_{M_q}\frac{q\bar{c}}{V} + C_{M_{\delta_e}}\delta_e + C_{M_{PLA}}PLA) \tag{8.20}$$

where C_{M_α} and C_{M_q} are stability derivatives. $C_{M_{\delta_e}}$ and $C_{M_{PLA}}$ are the control derivatives. Inaccurate C_{M_α}, C_{M_q}, $C_{M_{\delta_e}}$ and $C_{M_{PLA}}$ bring model uncertainties in the control law design and can result in performance degradation.

The uncertainties in these parameters must be taken into account when designing the control laws. Researchers have proposed many adaptive controllers [61–63, 173] which consider model uncertainties. For instance, ABS [61, 173] approaches are proposed which estimate the unknown aerodynamic forces and moments by updating the parameter estimate through a specific update law. However, since there are many parameters to be estimated, the tuning of the update law gains is time-consuming [63] and the adaptive controller is computationally intensive [63]. In this chapter, the idea is to design the control loop which has model uncertainties using an incremental control approach. Since model uncertainties only exist in the flight path control loop and the angular rate control loop, the incremental approach design is only applied to these two control loops. This can be seen in Figure 8.2. The other two loops, position control loop and attitude control loop, are designed based on standard NDI approach.

8.4. TRAJECTORY CONTROL LAW DESIGN

In this section, the design of the control law for the four loops is presented. Finally, the overall control architecture is presented.

8.4.1. POSITION CONTROL LOOP

The position control contains no model uncertainties and is designed based on the NDI control law (see Appendix C). It is assumed that the reference trajectory is generated

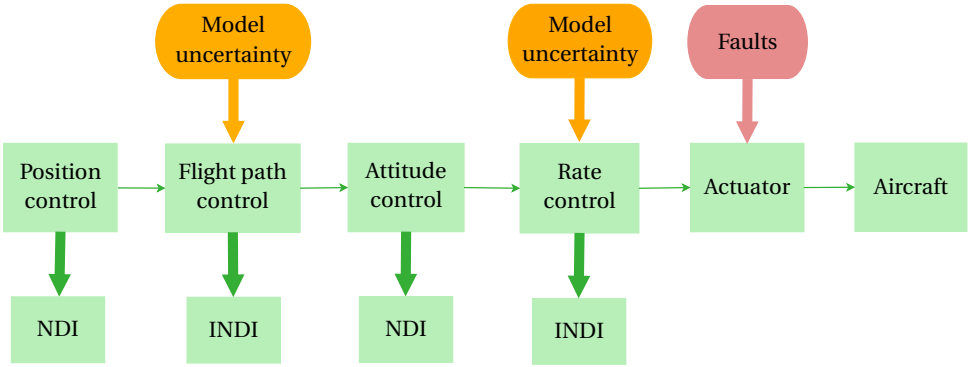


Figure 8.2: Distributions of the model uncertainties and controllers

through

$$\begin{bmatrix} \dot{x}^{ref} \\ \dot{y}^{ref} \\ \dot{z}^{ref} \end{bmatrix} = \begin{bmatrix} V^{ref}\cos\chi^{ref}\cos\gamma^{ref} \\ V^{ref}\sin\chi^{ref}\cos\gamma^{ref} \\ -V^{ref}\sin\gamma^{ref} \end{bmatrix} \tag{8.21}$$

In this chapter, we also try to follow the reference V^{ref}. Through the last two equations of Eq. (8.2), the desired reference commands for χ and γ can be calculated based on the standard NDI approach [179] as follows (see Appendix C for more details):

$$\begin{bmatrix} \chi^{des} \\ \gamma^{des} \end{bmatrix} = \begin{bmatrix} \arcsin(v_y/(V^{ref}\cos\gamma)) \\ -\arcsin(v_z/(V^{ref})) \end{bmatrix} \tag{8.22}$$

where v_y and v_z are virtual controls which can be designed using classical linear control such as a Proportional-Integral-Derivative control as follows:

$$v_y(t) = K_P^p e_y(t) + K_I^p \int_0^t e_y(\tau)d\tau + K_D^p \frac{de_y(t)}{dt} \tag{8.23}$$

$$v_z(t) = K_P^p e_z(t) + K_I^p \int_0^t e_z(\tau)d\tau + K_D^p \frac{de_z(t)}{dt} \tag{8.24}$$

where

$$e_y(t) = y^{ref}(t) - y(t) \tag{8.25}$$

$$e_z(t) = z^{ref}(t) - z(t). \tag{8.26}$$

and K_P^p, K_I^p and K_D^p are the gains for the position control loop.

8.4.2. FLIGHT PATH CONTROL LOOP

The flight path control loop requires the calculation of the aerodynamic forces which contains model uncertainties. Therefore, it needs to be designed based on the incremental approach.

DESIGN OF μ^{des}

In the flight path control loop, the objective is to steer V_g, χ and γ to their references denoted as V_g^{ref}, χ^{des} and γ^{des}. The virtual control variables are μ, α and PLA. However, μ, α and PLA do not appear affine in Equation (8.5). Therefore, the equation needs to be rewritten to be in the affine-in-control form.

Express the forces X and Z as follows:

$$X = \bar{q}SC_X = \bar{q}S(C_{X_0} + C_{X_\alpha}\alpha + C_{X_q}\frac{q\bar{c}}{V} + C_{X_{\delta_e}}\delta_e + C_{X_{PLA}}PLA) \tag{8.27}$$

$$Z = \bar{q}SC_Z = \bar{q}S(C_{Z_0} + C_{Z_\alpha}\alpha + C_{Z_q}\frac{q\bar{c}}{V} + C_{Z_{\delta_e}}\delta_e + C_{Z_{PLA}}PLA) \tag{8.28}$$

Since the virtual control variables are μ, α and PLA, Equations (8.27) and (8.28) can be rewritten into the following:

$$X = \bar{q}SC_X = \bar{q}S(\bar{C}_{X_0} + C_{X_\alpha}\alpha + C_{X_{PLA}}PLA) \tag{8.29}$$

$$Z = \bar{q}SC_Z = \bar{q}S(\bar{C}_{Z_0} + C_{Z_\alpha}\alpha + C_{Z_{PLA}}PLA) \tag{8.30}$$

where

$$\bar{C}_{X_0} = C_X - C_{X_\alpha}\alpha - C_{X_{PLA}}PLA \tag{8.31}$$

$$\bar{C}_{Z_0} = C_Z - C_{Z_\alpha}\alpha - C_{Z_{PLA}}PLA \tag{8.32}$$

Express Y as $Y = mA_y$ and substitute it with Equations (8.29) and (8.30) into Equation (8.4), it follows

$$\dot{\mathbf{x}}_1 = \begin{bmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{mV_g\cos\gamma} & 0 \\ 0 & 0 & \frac{-1}{mV_g} \end{bmatrix} \left(\mathbf{T_{VB}} \begin{bmatrix} \bar{q}S(\bar{C}_{X_0} + C_{X_{PLA}}PLA + C_{X_\alpha}\alpha) \\ mA_y \\ \bar{q}S(\bar{C}_{Z_0} + C_{Z_{PLA}}PLA + C_{Z_\alpha}\alpha) \end{bmatrix} + \mathbf{T_{VE}} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \right) \tag{8.33}$$

In the attitude control loop, β should be kept at zero. Therefore, it is reasonable to assume that β is close to zero. Under this condition, Equation (8.33) can be further simplified into

$$\dot{\mathbf{x}}_1 = \begin{bmatrix} \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{mV_g\cos\gamma} & 0 \\ 0 & 0 & -\frac{1}{mV_g} \end{bmatrix} \left(\begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ \sin\alpha\sin\mu & \cos\mu & -\cos\alpha\sin\mu \\ -\sin\alpha\cos\mu & \sin\mu & \cos\alpha\cos\mu \end{bmatrix} \begin{bmatrix} \bar{q}S(\bar{C}_{X_0} + C_{X_{PLA}}PLA + C_{X_\alpha}\alpha) \\ mA_y \\ \bar{q}S(\bar{C}_{Z_0} + C_{Z_{PLA}}PLA + C_{Z_\alpha}\alpha) \end{bmatrix} + \begin{bmatrix} -mg\sin\gamma \\ 0 \\ mg\cos\gamma \end{bmatrix} \right) \tag{8.34}$$

First, let us derive μ^{des}. Due to $X = mA_x$ and $Z = mA_z$, the last two equations of Equation (8.34) can be written as:

$$\dot{\chi} = \frac{1}{V_g \cos\gamma}(A_x \sin\alpha \sin\mu + A_y \cos\mu - A_z \cos\alpha \sin\mu) \tag{8.35}$$

$$\dot{\gamma} = \frac{1}{V_g}(A_x \sin\alpha \cos\mu - A_y \sin\mu - A_z \cos\alpha \cos\mu - g\cos\gamma) \tag{8.36}$$

Rewrite Equations (8.35) and (8.36) into the following form:

$$(A_x \sin\alpha - A_z \cos\alpha)\sin\mu = V_g \cos\gamma \dot{\chi} - A_y \cos\mu \tag{8.37}$$

$$(A_x \sin\alpha - A_z \cos\alpha)\cos\mu = V_g \dot{\gamma} + A_y \sin\mu + g\cos\gamma \tag{8.38}$$

The desired command μ^{des} can be designed as follows:

$$\mu^{des} = \arctan\frac{\nu_\chi V_g \cos\gamma - A_y \cos\mu}{\nu_\gamma V_g + A_y \sin\mu + g\cos\gamma} \tag{8.39}$$

where ν_χ and ν_γ are the virtual controls which are designed in the same way as ν_y and ν_z which are given in Equations (8.23) and (8.24). If further assume that A_y is small, the control law can be further simplified as follows:

$$\mu^{des} = \arctan\frac{\nu_\chi V_g \cos\gamma}{\nu_\gamma V_g + g\cos\gamma} \tag{8.40}$$

DESIGN OF PLA^{des} AND α^{des}

Next, the design of the desired commands PLA^{des} and α^{des} is presented. The design of PLA^{des} and α^{des} is different from that of μ^{des}. This is because there are model uncertainties in the first and third equations of Equation (8.34). Therefore, they should be designed using the incremental approach.

In order to design the incremental control law, the system dynamics need to be written into the incremental control form. First, write the first and third equations of Equation (8.34) into:

$$\dot{\bar{\mathbf{x}}}_1 = \bar{\mathbf{f}}_1(\mathbf{x}_1, \mathbf{x}_2) + \bar{\mathbf{g}}_1 \bar{\mathbf{x}}_2 \tag{8.41}$$

where

$$\bar{\mathbf{x}}_1 = [V_g, \gamma]^T, \bar{\mathbf{x}}_2 = [PLA, \alpha]^T \tag{8.42}$$

$$\bar{\mathbf{f}}_1 = \begin{bmatrix} -g\sin\gamma + \frac{\bar{q}S}{m}(\bar{C}_{X_0}\cos\alpha + \bar{C}_{Z_0}\sin\alpha) \\ -\frac{A_y \sin\mu}{V_g} - \frac{g\cos\gamma}{V_g} - \frac{\bar{q}S}{mV_g}(-\bar{C}_{X_0}\sin\alpha\cos\mu + \bar{C}_{Z_0}\cos\alpha\cos\mu) \end{bmatrix} \tag{8.43}$$

$$\bar{\mathbf{g}}_1 = \frac{\bar{q}S}{mV_g}\begin{bmatrix} V_g(C_{X_{PLA}}\cos\alpha + C_{Z_{PLA}}\sin\alpha) & V_g(C_{X_\alpha}\cos\alpha + C_{Z_\alpha}\sin\alpha) \\ C_{X_{PLA}}\sin\alpha\cos\mu - C_{Z_{PLA}}\cos\alpha\cos\mu & C_{X_\alpha}\sin\alpha\cos\mu - C_{Z_\alpha}\cos\alpha\cos\mu \end{bmatrix} \tag{8.44}$$

According to [176, 177, 180], Equation (8.41) can be written into the following incremental control form:

$$\dot{\bar{\mathbf{x}}}_1 = \dot{\bar{\mathbf{x}}}_{1,0} + \bar{\mathbf{g}}_1 \Delta\bar{\mathbf{x}}_2 \tag{8.45}$$

where

$$\dot{\bar{\mathbf{x}}}_{1,0} = \bar{\mathbf{f}}_1(\bar{\mathbf{x}}_{1,0}, \bar{\mathbf{x}}_{2,0}) + \bar{\mathbf{g}}_1 \bar{\mathbf{x}}_{2,0} \tag{8.46}$$

where $\bar{\mathbf{x}}_{1,0}$ and $\bar{\mathbf{x}}_{2,0}$ denote $\bar{\mathbf{x}}_1$ and $\bar{\mathbf{x}}_2$ at the last time step. $\dot{\bar{\mathbf{x}}}_{1,0}$ denotes the derivative of $\bar{\mathbf{x}}_1$ at the last time step.

Then the desired virtual incremental input $\Delta\bar{\mathbf{x}}_2$ can be designed by

$$\Delta\bar{\mathbf{x}}_2 = \bar{\mathbf{g}}_1^{-1}(\nu_{\bar{\mathbf{x}}_1} - \dot{\bar{\mathbf{x}}}_{1,0}) \tag{8.47}$$

where $\nu_{\bar{\mathbf{x}}_1} = [\nu_{V_g}, \nu_\gamma]^T$ is designed based on the errors between V_g, γ and V_g^{ref}, γ^{des}.

The final desired command for $\bar{\mathbf{x}}_2$ can be given by

$$\bar{\mathbf{x}}_2^{des} = \bar{\mathbf{x}}_{2,0} + \Delta\bar{\mathbf{x}}_2 \tag{8.48}$$

There are two engines, of which the desired references PLA_1^{des} and PLA_2^{des} are both equal to PLA^{des}.

CALCULATION OF THE DERIVATIVES $\dot{\bar{x}}_{1,0}$

$\dot{\bar{\mathbf{x}}}_{1,0}$ can be calculated using different approaches. It can be calculated using the sliding mode differentiator [127]. Alternatively, it can also be obtained using a second-order low-pass filter. According to [176], $\dot{\bar{\mathbf{x}}}_{1,0}$ can be computed by passing $\bar{\mathbf{x}}_{1,0}$ through the following washout filter:

$$\frac{s\omega_n^2}{s^2 + 2\zeta_n\omega_n s + \omega_n^2} \tag{8.49}$$

with $\zeta_n = 0.8$, $\omega_n = 25$ rad/s. These are selected by trial and error to reduce the influence of noise [176].

The overall block diagram of the flight path control loop is shown in Figure 8.3. It can be noted in the figure that to calculate $\dot{\bar{x}}_{2,0}$, a second-order filter is used to counteract the influence of phase delay [181].

8.4.3. ATTITUDE CONTROL LOOP

In this control loop, the objective is to follow μ^{des}, α^{des} and β^{ref}. μ^{des}, α^{des} have already been given in the flight path control loop. β^{ref} is set to zero to enable coordinated turns. The virtual control variables are the angular rates $\boldsymbol{\omega}$. In this control loop, there is no model uncertainty. Therefore, there is no need to design an incremental control law.

The desired virtual control law, based on the standard NDI approach [179] (see Appendix C for more details), is given as follows:

$$\begin{bmatrix} p^{des} \\ q^{des} \\ r^{des} \end{bmatrix} = \begin{bmatrix} \cos\alpha\cos\beta & 0 & \sin\alpha \\ \sin\beta & 1 & 0 \\ \sin\alpha\cos\beta & 0 & -\cos\alpha \end{bmatrix} \begin{bmatrix} \nu_\mu \\ \nu_\alpha \\ \nu_\beta \end{bmatrix} + \mathbf{T_{BV}} \begin{bmatrix} -\dot{\chi}\sin\gamma \\ \dot{\gamma} \\ \dot{\chi}\cos\gamma \end{bmatrix} \tag{8.50}$$

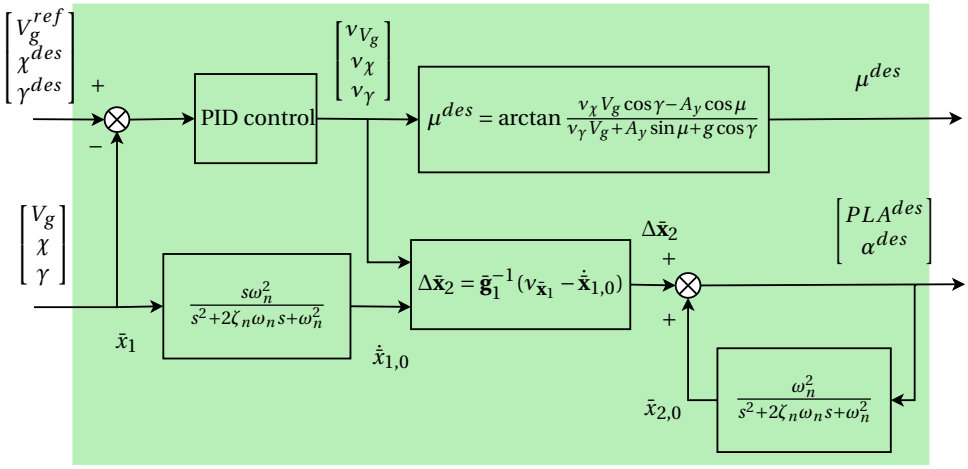


Figure 8.3: Block diagram of the flight path control loop

where ν_μ, ν_α and ν_β are virtual controls designed based on the errors between μ^{des}, α^{des}, β^{ref} and μ, α, β. It is seen from Eq. (8.50) that in order to calculate the virtual control law, $\dot{\chi}$ and $\dot{\gamma}$ are required. This can be calculated using Eqs. (8.35) and (8.36). In these equations, A_x, A_y and A_z are directly measured by the accelerometers of the aircraft.

8.4.4. ANGULAR RATE CONTROL LOOP

In this control loop, the control objective is to follow the angular rate references generated by the attitude control loop denoted as p^{des}, q^{des} and r^{des}. The control variables are the control surface deflections denoted as $\mathbf{u}^{des}$ (δ_a^{des}, δ_e^{des} and δ_r^{des}). As mentioned, there are also model uncertainties in this control loop. Therefore, this loop also has to be designed based on the incremental control approach.

As done in the flight path control loop, Equation (8.15) can be rewritten into the following incremental form:

$$\dot{\mathbf{x}}_3 = \dot{\mathbf{x}}_{3,0} + \mathbf{G}_3 \Delta\mathbf{u} \tag{8.51}$$

with

$$\dot{\mathbf{x}}_{3,0} = \mathbf{f}_{3,0} + \mathbf{G}_3 \mathbf{u}_0 \tag{8.52}$$

where

$$\dot{\mathbf{x}}_{3,0} = \begin{bmatrix} \dot{p}_0 \\ \dot{q}_0 \\ \dot{r}_0 \end{bmatrix}, \mathbf{u}_0 = \begin{bmatrix} \delta_{a,0} \\ \delta_{e,0} \\ \delta_{r,0} \end{bmatrix} \tag{8.53}$$

$\dot{\mathbf{x}}_{3,0}$ denotes the derivatives of $\mathbf{x}_3$ in the previous time step. $\mathbf{f}_{3,0}$ and $\mathbf{u}_0$ denotes $\mathbf{f}_3$ and $\mathbf{u}$ in the previous time step. The calculation of $\dot{\mathbf{x}}_{3,0}$ can be done in the same way as that of $\dot{\bar{\mathbf{x}}}_{1,0}$ through the use of (8.49).

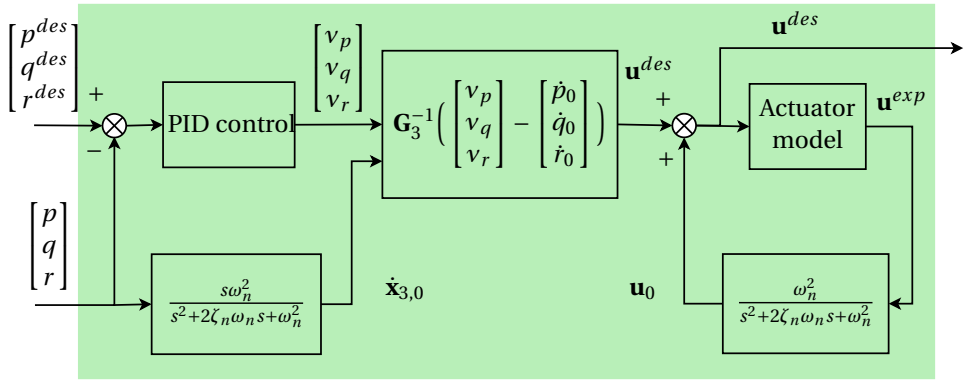


Figure 8.4: Block diagram of the angular rate control loop

Now the desired incremental control law can be presented as follows:

$$\Delta\mathbf{u}^{des} = \begin{bmatrix} \Delta\delta_a^{des} \\ \Delta\delta_e^{des} \\ \Delta\delta_r^{des} \end{bmatrix} = \mathbf{G}_3^{-1}\left(\begin{bmatrix} \nu_p \\ \nu_q \\ \nu_r \end{bmatrix} - \begin{bmatrix} \dot{p}_0 \\ \dot{q}_0 \\ \dot{r}_0 \end{bmatrix} \right) \tag{8.54}$$

where ν_p, ν_q and ν_r are virtual controls designed based on the errors between p^{des}, q^{des}, r^{des} and p, q, r.

The final desired control law $\mathbf{u}^{des}$ is computed as follows:

$$\begin{bmatrix} \delta_a^{des} \\ \delta_e^{des} \\ \delta_r^{des} \end{bmatrix} = \begin{bmatrix} \delta_{a,0} \\ \delta_{e,0} \\ \delta_{r,0} \end{bmatrix} + \begin{bmatrix} \Delta\delta_a^{des} \\ \Delta\delta_e^{des} \\ \Delta\delta_r^{des} \end{bmatrix} \tag{8.55}$$

The block diagram of the angular rate control loop is shown in Figure 8.4. In the figure, $\mathbf{u}^{exp}$ are the expected control surface deflections and will be discussed in more details in Section 8.5.

8.4.5. CONTROL STRUCTURE

The block diagram of the control loops is given in Figure 8.5. In the figure, the reference commands and the desired reference commands generated by each control loop are shown. The performance of this trajectory controller will be demonstrated in the following sections.

8.5. AIRCRAFT MODEL AND BASELINE CONTROLLER

In this section, the aircraft model, the actuators and the control allocation scheme are introduced first in Section 8.5.1. In Section 8.5.2, the baseline controller, which is propose in [171], is briefly introduced.

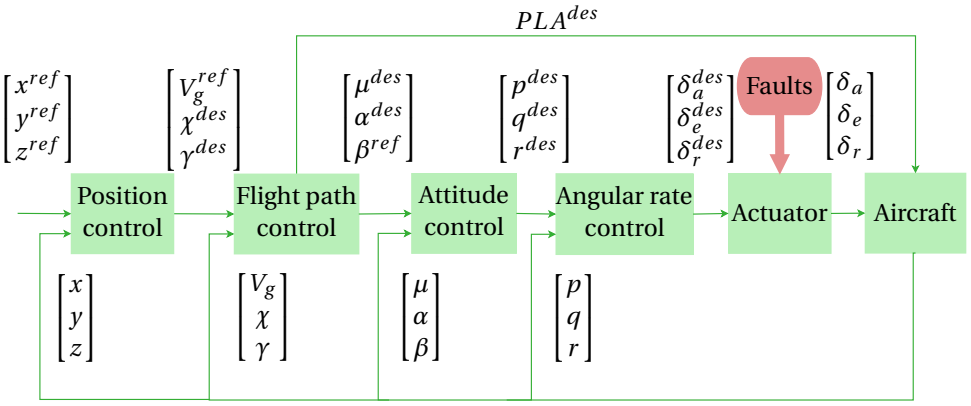


Figure 8.5: Block diagram of the proposed trajectory controller

8.5.1. AIRCRAFT MODEL AND CONTROL ALLOCATION

The aircraft model used is the Cessna Citation aircraft [182] shown in Fig. 8.1. The actuator dynamics are modeled as first-order low-pass filters $1/(\tau s+1)$. The parameter τ and the position limits of the actuators are given in Table 8.1.

There are two engines onboard the aircraft. The thrust is controlled by the throttle (power level angle). The power level angle is within [0.2356, 1.0820] rad.

Table 8.1: Time constants and position limits of the actuators

Actuator	$1/\tau$	Minimum position limit	Maximum position limit	Unit
elevator	13	-0.35	0.26	[rad]
aileron (left/right)	13	-0.65	0.65	[rad]
rudder (upper/lower)	13	-0.38	0.38	[rad]

For reconfiguring the controller, the left and right ailerons are assumed to be actuated by independent actuation systems such that they can deflect independently. Furthermore, the rudder is split into an upper part and a lower part in a way such that the effectiveness of these two parts are the same. Such a configuration is also considered in other papers such as [183, 184].

The positive deflections are defined in the conventional way. For example, positive δ_a means the left aileron deflects upwards and the right aileron deflects downwards.

Although there are two independent rudders and two independent ailerons, the controller design does not consider them as independent. Since the control surface commands computed by the INDI controller are $[\delta_a^{des}, \delta_e^{des}, \delta_r^{des}]^T$, the desired commands

of the two rudders and two ailerons are as follows:

$$\delta_{al}^{des} = \delta_{ar}^{des} = \delta_{a}^{des} \tag{8.56}$$
$$\delta_{ru}^{des} = \delta_{rl}^{des} = \delta_{r}^{des} \tag{8.57}$$

where δ_{al}^{des}, δ_{ar}^{des}, δ_{ru}^{des} and δ_{rl}^{des} denote the commands for the left aileron, right aileron, upper rudder and lower rudder, respectively. Take the upper rudder for example, even if it is stuck, the command generated for its deflection is still δ_{r}^{des}. The effectiveness of the control surface pairs are assumed to be the same such that the parameters in Equation (8.14) can be denoted as:

$$C_{l\delta_a} = C_{l\delta_{al}} + C_{l\delta_{ar}} = 2C_{l\delta_{al}} \tag{8.58}$$
$$C_{l\delta_r} = C_{l\delta_{ru}} + C_{l\delta_{rl}} = 2C_{l\delta_{ru}} \tag{8.59}$$
$$C_{n\delta_a} = C_{n\delta_{al}} + C_{n\delta_{ar}} = 2C_{n\delta_{al}} \tag{8.60}$$
$$C_{n\delta_r} = C_{n\delta_{ru}} + C_{n\delta_{rl}} = 2C_{n\delta_{ru}} \tag{8.61}$$

8.5.2. BASELINE CONTROL APPROACH: NDI-MI

In this section, a baseline control approach used for comparison, NDI-MI [171], is briefly introduced. The NDI-MI uses Eq. (8.5) for flight path control and Eq. (8.15) for angular rate control while the INDI uses Eq. (8.45) for flight path control and Eq. (8.51) for angular rate control. The core of the method is to identify the unknown parameters online to reduce the model uncertainty. Take the lateral motion for example, the lateral force and moments are assumed to be expressed as follows:

$$Y = \bar{q}SC_y = \bar{q}S(C_{y_0} + C_{y_\beta}\beta + C_{y_p}\frac{pb}{2V} + C_{y_r}\frac{rb}{2V} + C_{y_{\delta_a}}\delta_a^{exp} + C_{y_{\delta_r}}\delta_r^{exp}) \tag{8.62}$$
$$L = \bar{q}SC_l = \bar{q}S(C_{l_0} + C_{l_\beta}\beta + C_{l_p}\frac{pb}{2V} + C_{l_r}\frac{rb}{2V} + C_{l_{\delta_a}}\delta_a^{exp} + C_{l_{\delta_r}}\delta_r^{exp}) \tag{8.63}$$
$$N = \bar{q}SC_n = \bar{q}S(C_{n_0} + C_{n_\beta}\beta + C_{n_p}\frac{pb}{2V} + C_{n_r}\frac{rb}{2V} + C_{n_{\delta_a}}\delta_a^{exp} + C_{n_{\delta_r}}\delta_r^{exp}) \tag{8.64}$$

where δ_a^{exp} and δ_r^{exp}, which can be obtained using the actuator model as shown in Figure 8.6, denote the expected actuator deflections. The NDI-MI also treats the control surface pairs as one control surface. For instance, it does not identify $C_{l\delta_{al}}$ and $C_{l\delta_{ar}}$, but only identifies $C_{l\delta_a}$. The aerodynamic model structure in Equations (8.62)-(8.64) is a simplified structure. To deal with aircraft dynamics with significant nonlinearities, a more complex structure should be designed [169, 175]. However, this is not required if the proposed approach, which uses the INDI, is used. The difference between δ_a^{exp}, δ_e^{exp}, δ_r^{exp} and δ_a, δ_e, δ_r is shown in Figure 8.6. It is seen that the expected deflections δ_a^{exp}, δ_e^{exp}, δ_r^{exp} are free from the influence of faults.

First, the lateral force coefficient C_y and lateral moment coefficients C_l and C_n are calculated [171]. Then the parameters can be identified using parameter estimation methods such as the recursive least squares method [171]. The identified parameters

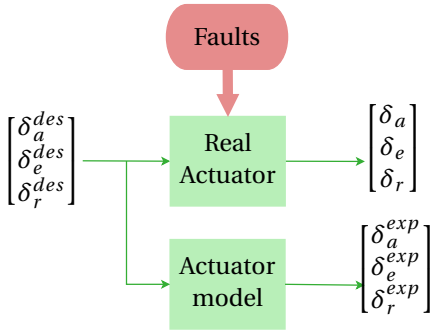


Figure 8.6: True and expected actuator deflections

are denoted as:

$$\hat{C}_{y_0}, \hat{C}_{y_\beta}, \hat{C}_{y_p}, \hat{C}_{y_r}, \hat{C}_{y_{\delta_a}}, \hat{C}_{y_{\delta_r}} \tag{8.65}$$

$$\hat{C}_{l_0}, \hat{C}_{l_\beta}, \hat{C}_{l_p}, \hat{C}_{l_r}, \hat{C}_{l_{\delta_a}}, \hat{C}_{l_{\delta_r}} \tag{8.66}$$

$$\hat{C}_{n_0}, \hat{C}_{n_\beta}, \hat{C}_{n_p}, \hat{C}_{n_r}, \hat{C}_{n_{\delta_a}}, \hat{C}_{n_{\delta_r}} \tag{8.67}$$

When an actuator fault occurs, the above parameters can change due to the difference between the true and expected actuator deflections. In this case, the parameter estimation process needs to be re-initialized in order to identify the change of the parameters. The re-initialization is triggered by statistics monitoring [171]. Define

$$M_{lat} = \frac{1}{N_w} \sum_{i=0}^{N_w} (\Delta C_{y,i}^2 + \Delta C_{l,i}^2 + \Delta C_{n,i}^2) \tag{8.68}$$

where N_w is a pre-selected size of a moving window and

$$\Delta C_{y,i} = C_y - (\hat{C}_{y_0} + \hat{C}_{y_\beta}\beta + \hat{C}_{y_p}\frac{pb}{2V} + \hat{C}_{y_r}\frac{rb}{2V} + \hat{C}_{y_{\delta_a}}\delta_a^{exp} + \hat{C}_{y_{\delta_r}}\delta_r^{exp}) \tag{8.69}$$

$$\Delta C_{l,i} = C_l - (\hat{C}_{l_0} + \hat{C}_{l_\beta}\beta + \hat{C}_{l_p}\frac{pb}{2V} + \hat{C}_{l_r}\frac{rb}{2V} + \hat{C}_{l_{\delta_a}}\delta_a^{exp} + \hat{C}_{l_{\delta_r}}\delta_r^{exp}) \tag{8.70}$$

$$\Delta C_{n,i} = C_n - (\hat{C}_{n_0} + \hat{C}_{n_\beta}\beta + \hat{C}_{n_p}\frac{pb}{2V} + \hat{C}_{n_r}\frac{rb}{2V} + \hat{C}_{n_{\delta_a}}\delta_a^{exp} + \hat{C}_{n_{\delta_r}}\delta_r^{exp}) \tag{8.71}$$

The parameter estimation needs to be re-initialized when the following is satisfied:

$$M_{lat} > T_{M_{lat}} \tag{8.72}$$

where $T_{M_{lat}}$ is the threshold to trigger the re-initialization of the parameter identification. Furthermore, define the following difference between the real values of C_y, C_l, C_n

and the estimated ones using the NDI-MI as follows:

$$\Delta C_y = C_{y,real} - (\hat{C}_{y_0} + \hat{C}_{y_\beta}\beta + \hat{C}_{y_p}\frac{pb}{2V} + \hat{C}_{y_r}\frac{rb}{2V} + \hat{C}_{y_{\delta_a}}\delta_a^{exp} + \hat{C}_{y_{\delta_r}}\delta_r^{exp}) \tag{8.73}$$

$$\Delta C_l = C_{l,real} - (\hat{C}_{l_0} + \hat{C}_{l_\beta}\beta + \hat{C}_{l_p}\frac{pb}{2V} + \hat{C}_{l_r}\frac{rb}{2V} + \hat{C}_{l_{\delta_a}}\delta_a^{exp} + \hat{C}_{l_{\delta_r}}\delta_r^{exp}) \tag{8.74}$$

$$\Delta C_n = C_{n,real} - (\hat{C}_{n_0} + \hat{C}_{n_\beta}\beta + \hat{C}_{n_p}\frac{pb}{2V} + \hat{C}_{n_r}\frac{rb}{2V} + \hat{C}_{n_{\delta_a}}\delta_a^{exp} + \hat{C}_{n_{\delta_r}}\delta_r^{exp}) \tag{8.75}$$

where $C_{y,real}$, $C_{l,real}$, $C_{n,real}$ are the real values of C_y, C_l and C_n, which is obtained from the Citation model [182]. ΔC_y, ΔC_l and ΔC_n can be used to evaluate the identification performance of the NDI-MI [174].

Similarly, the re-initialization of the parameter identification for the longitudinal motion is triggered when

$$M_{long} > T_{M_{long}} \tag{8.76}$$

where M_{long} and $T_{M_{long}}$ are defined the same way as M_{lat} and $T_{M_{lat}}$. $T_{M_{long}}$ is the threshold to trigger the re-initialization of the parameter identification. ΔC_x, ΔC_z and ΔC_m are defined to denote the differences between the real values of C_x, C_z, C_m and the estimated values using the NDI-MI.

A comparison of NDI-MI with the proposed approach will be demonstrated in the following section.

Remark: Since the NDI-MI uses Eqs. (8.5) and (8.15), it requires the aerodynamic model structure (described by Eqs. (8.62), (8.63) and (8.64)). This aerodynamic model structure can change during the whole flight envelope [170]. It also changes when there are actuator faults. Consequently, the performance of the NDI-MI degrades when the aerodynamic model structure is not accurate. However, the proposed method using the INDI is less sensitive to the the aerodynamic model structure since it does not require the complete aerodynamic model as can be seen in Eq. (8.45) and (8.51).

8.6. PERFORMANCE VALIDATION RESULTS

In this section, the performance of the trajectory controller using the INDI will be demonstrated. The trajectory reference in three-dimensional space is given in Fig 8.7(a). x^{ref}, y^{ref} and z^{ref} are given in Figure 8.7(b). Besides, $V^{ref} = 90$ m/s and $\beta^{ref} = 0$ rad. This means that the aircraft should fly at a constant speed and perform coordinated turns.

To demonstrate the performance of the proposed trajectory controller, the NDI-MI is also applied to control the trajectory of the aircraft. The control gains of the four loops used for both controllers are given in Table 8.2. In the table, K_P, K_I and K_D are the PID gains for each loop.

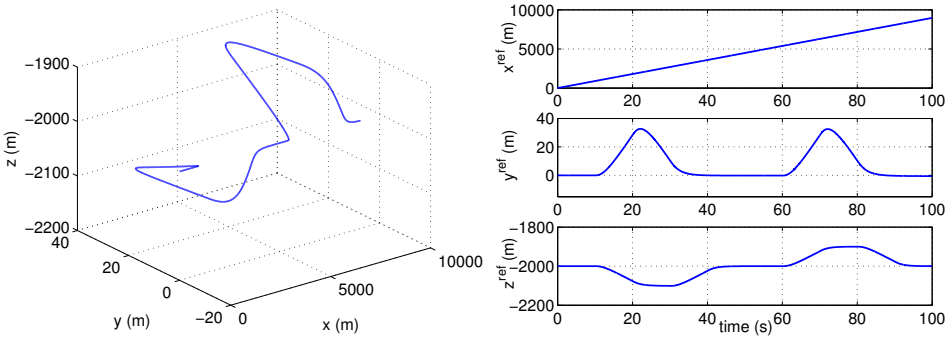
The initial states of the aircraft is:

$$z = -2000 \text{ m}, \ \gamma = -3\pi/180 \text{ rad},$$
$$V_g = 90 \text{ m/s}, \ \ \alpha = 3.2\pi/180 \text{ rad}$$

Other states are close to zero and are not shown.

Table 8.2: Control gains for four loops

Gains	Position	Flight path	Attitude	Angular rate
K_P	1	1.5	2.5	5
K_I	0.01	0.01	0.5	0.5
K_D	0.05	0.05	0.5	0.5



(a) Trajectory reference in three-dimensional space.

(b) x^{ref}, y^{ref} and z^{ref}.

Figure 8.7: Results of aircraft trajectory control without faults

In Section 8.6.1, the actuator fault scenario is introduced. The performance of the proposed trajectory controller with the one using the NDI-MI is compared under three scenarios which is presented in Sections 8.6.2, 8.6.3 and 8.6.4.

8

8.6.1. ACTUATOR FAULT SCENARIO AND MODELING OF THE FAULTS

In this section, the modeling of the actuator faults are presented.

The actuator faults considered in this chapter are jamming faults where the control surfaces are stuck at a certain position. The specific fault occurrence time and stuck position is given in Table 8.3.

It can be seen that during 25 s $< t <$ 100 s, the left aileron is stuck. During 50 s $< t <$ 100 s, the left aileron and the upper rudder both fail. Therefore, simultaneous actuator faults are taken into account in this chapter.

Table 8.3: Actuator faults

Time interval	Actuator	Fault type	Stuck position	Fault unit
$t > 25$ s	left aileron	Jamming	0.56	[rad]
$t > 50$ s	upper rudder	Jamming	0.2	[rad]

8.6.2. VALIDATION IN THE ABSENCE OF FAULTS

In this section, the performance of trajectory control using the NDI-MI and the INDI is compared. In this section, there are no actuator faults.

Both controller require initial values for the aerodynamic parameters (control and stability derivatives). For the INDI, once the initial parameters are given, it is considered fixed during the whole flight. For the NDI-MI, although it requires initial parameters, the initial parameters will be changed due to the on-line model identification. This scenario is to test the performance of the controllers in a nominal situation when there are no model uncertainties and faults. Therefore, the initial parameters given to the controllers are both the real parameters of the aircraft at the initial condition.

RESULTS OF USING THE NDI-MI

The trajectory control results using the NDI-MI are shown in Figure 8.8. As can be seen from Figure 8.8(a), the trajectory can follow the reference. The true and desired V_g, χ and γ are shown in Figure 8.8(b). As can be seen, the response is satisfactory. The velocity is kept within 85-95 m/s. This is acceptable since the engine of the aircraft is not as powerful as a fighter. The engine power level angles are shown in Figure 8.8(c). It is seen that saturations are observed. The fluctuations in the beginning of γ response is caused by the fact that the initial γ is not zero and the aircraft needs to reach a new equilibrium to hold the altitude.

The control inputs are presented. Figure 8.8(c) shows PLA_1 and PLA_2. The control surface deflections δ_{ar}, δ_e and δ_{rl} are shown in Figure 8.8(d). The control surface deflections δ_{al} and δ_{ru} are the same as δ_{ar} and δ_{rl} respectively and are therefore not shown.

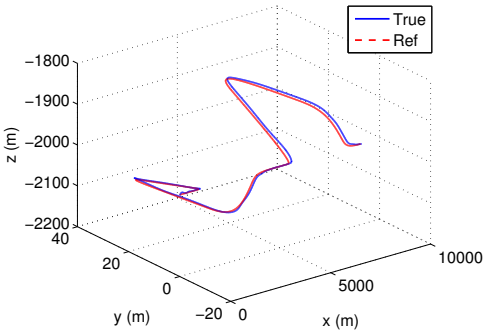
The differences between the real C_x, C_y, C_z, C_l, C_m, C_n and their estimates using the NDI-MI are shown in Figures 8.8(e) and 8.8(f) respectively. It is seen that all the differences are close to zero-mean, which indicates that the performance of the on-line model identification is satisfactory. This performance can be further improved by the re-identification, which is introduced in the following section.

RESULTS OF USING THE INDI

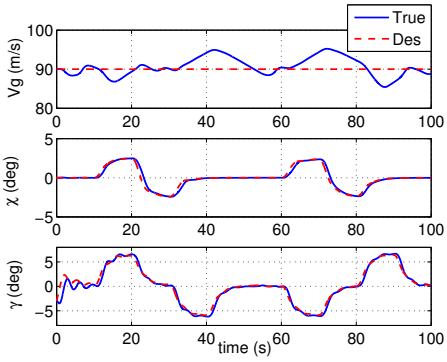
The trajectory control results using the INDI are shown in Figure 8.9. It is seen from Figure 8.9(a) that the trajectory can follow the reference. The tracking performance is satisfactory. V_g, χ and γ can follow the desired reference generated by the position control loop, as shown in Figure 8.9(b).

The attitude response using the INDI is shown in Figure 8.9(c). μ, α can follow their desired references and β is smaller than 0.2 deg. The desired references for p, q and r are shown in Figure 8.9(d). It is seen that the references are tracked well and no oscillations are observed.

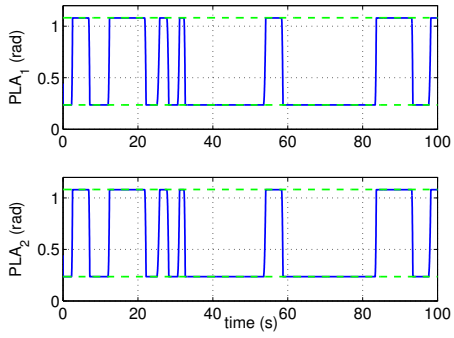
Finally, the control inputs using the INDI are shown in Figures 8.9(e) and 8.9(f). It is seen from Figure 8.9(e) that PLA_1 and PLA_2 also reach their limits. The deflections δ_{ar}, δ_e and δ_{rl} are shown in Figure 8.9(f). It is seen that no oscillations are present. Control surface deflections δ_{al} and δ_{ru} are the same as δ_{ar} and δ_{rl} respectively and are therefore not shown.



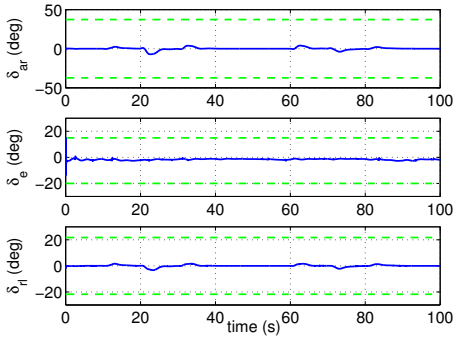
(a) Desired and true x, y and z



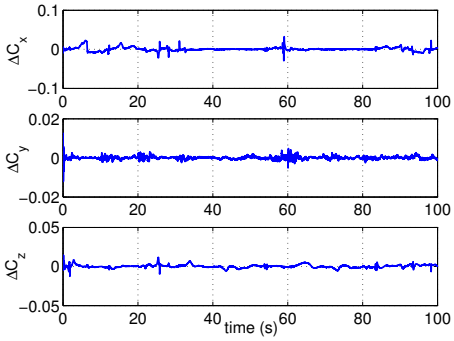
(b) Desired and true V_g, χ and γ



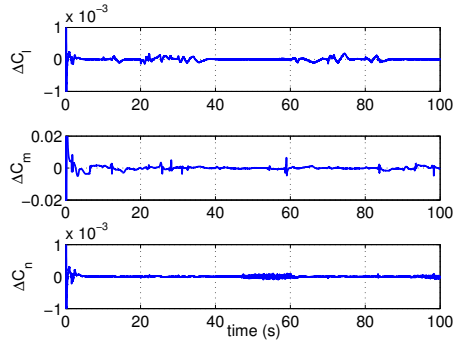
(c) PLA_1 and PLA_2, dashed lines represent the limits.



(d) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.

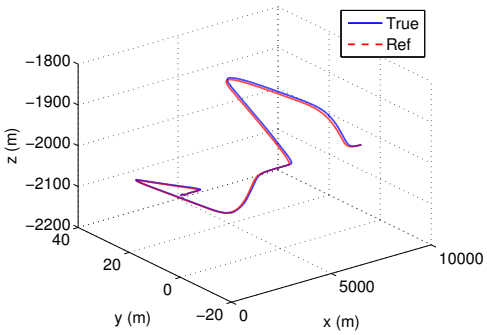


(e) ΔC_x, ΔC_y and ΔC_z

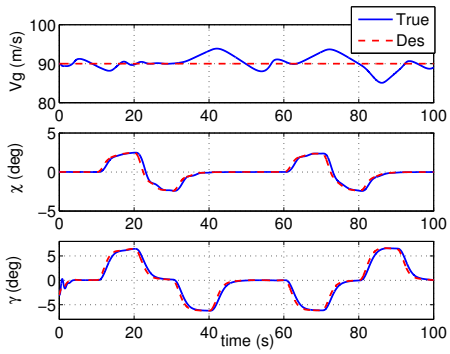


(f) ΔC_l, ΔC_m and ΔC_n

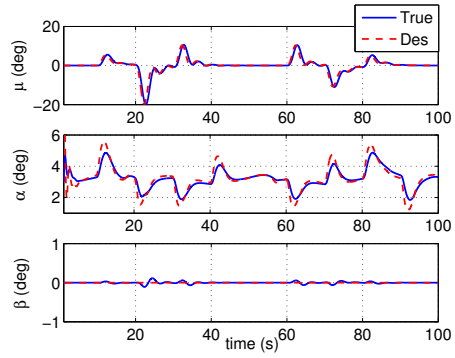
Figure 8.8: Results of aircraft trajectory control using the NDI-MI without faults.



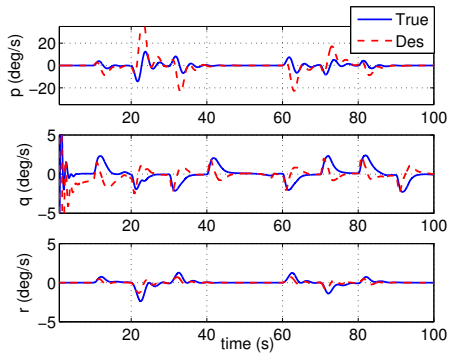
(a) Desired and true x, y and z



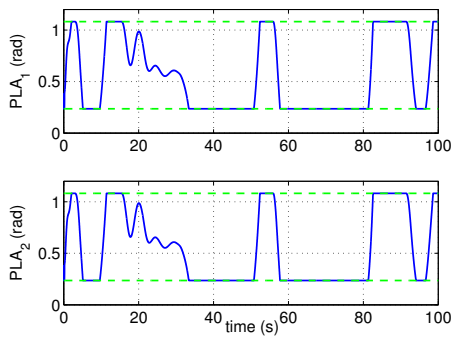
(b) Desired and true V_g, χ and γ



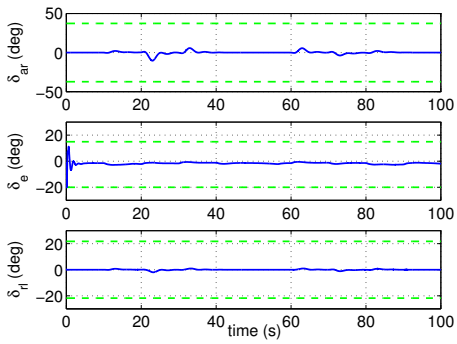
(c) Desired and true μ, α and β



(d) Desired and true p, q and r



(e) PLA_1 and PLA_2, dashed lines represent the limits.



(f) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.

Figure 8.9: Results of aircraft trajectory control using the INDI without faults

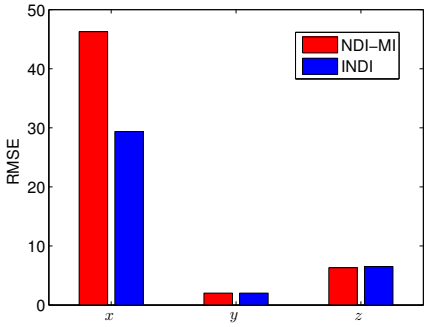


Figure 8.10: RMSE of x, y and z using the INDI and the NDI-MI

COMPARISON IN THE ABSENCE OF FAULTS

In this scenario, the performance of both the INDI and NDI-MI is satisfactory. The RMSEs of the tracking errors of x, y, z are shown in Figure 8.10. It is seen that the RMSEs of both controllers are close, especially the RMSEs of y and z.

8.6.3. VALIDATION IN THE PRESENCE OF UNCERTAINTIES

In this scenario, the initial parameters given to the two controllers are modified to generate model uncertainties. Then these parameters with model uncertainties will be given to the controllers to test the performance of the controllers in the presence of model uncertainties. It should be noted that the model uncertainties can also be viewed as the results of structural faults.

Persistent excitation is important in system identification. Although the requirement of persistent excitation is weaker using the NDI-MI [171], excitation is still needed in order to identify the derivatives. To ensure proper identification, a trigger for re-identification is required. To trigger the re-identification, thresholds ($T_{M_{long}}$ and $T_{M_{lat}}$) have to be chosen.

RESULTS OF USING THE NDI-MI

First of all, all the parameters are multiplied by 80%. The result using the NDI-MI are shown in Figure 8.11.

The trajectory of the aircraft is shown in Figure 8.11(a). It is seen that the aircraft trajectory can follow the reference well despite the model uncertainties. The control inputs are shown in Figure 8.11(b). No saturations are observed from the figure.

The reason why the NDI-MI worked in the presence of model uncertainties is due to the triggered re-identification. To trigger the re-identification, choosing the thresholds is critical. If the threshold is "large", it could lead to failure of the model identification (this will be demonstrated later on). In this case, the following thresholds are chosen:

$$T_{M_{long}} = 0.003,\ T_{M_{lat}} = 9 \times 10^{-4} \tag{8.77}$$

M_{long} and M_{lat} together with the thresholds using the NDI-MI are shown in Figure 8.11(c). It is seen that both M_{long} and M_{lat} exceed their thresholds and triggered the

re-identification. The re-identified parameters are shown in Figure 8.11(d).

To further evaluate the performance of the NDI-MI, the differences between the real C_x, C_y, C_z, C_l, C_m, C_n and their estimation using the NDI-MI are shown in Figures 8.11(e) and 8.11(f) respectively. It is seen that all the differences are zero-mean, which indicates that the performance of the on-line model identification is satisfactory.

In the following, the performance of the NDI-MI, when different thresholds are chosen, is demonstrated. The thresholds are slightly increased as follows:

$$T_{M_{long}} = 0.004,\ T_{M_{lat}} = 1.25 \times 10^{-3} \tag{8.78}$$

The results of using the NDI-MI are shown in Figure 8.12. In this case, M_{long} and M_{lat} together with the thresholds using the NDI-MI are shown in Figure 8.12(c). It is seen that the thresholds are not exceeded until after $t = 62$ s. In the first 60 s, the aircraft can still follow the trajectory reference, as shown in Figure 8.12(a). However, it can be seen from the control surface deflections (Figure 8.12(b)) that after $t = 10$ s, significant oscillations are observed. This is caused by the model uncertainties. After $t = 10$ s, there are longitudinal and lateral maneuvers and the aerodynamic parameters can further change due to the change of the flight condition. Although the parameters are identified on-line, without the re-identification, convergence of parameter estimation is not guaranteed. Consequently, incorrect estimated parameters (shown in Figure 8.12(d)) lead to the oscillations of the control surfaces.

The differences between the real C_x, C_y, C_z, C_l, C_m, C_n and their estimated values using the NDI-MI, when there are no re-identification, are shown in Figures 8.12(e) and 8.12(f). As can be seen from the figures, the differences are no longer zero-mean, which confirms that the estimated parameters are not correct.

The results when different thresholds are chosen demonstrate the importance of choosing a reasonable threshold. Further simulations indicate that when the parameters are multiplied with different constants, different thresholds should be chosen in order to trigger the re-identification. However, there are no rules for choosing the thresholds.

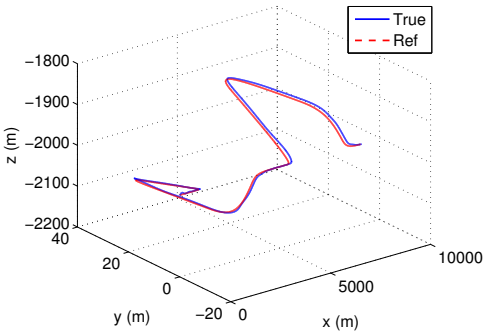
In the next, the model uncertainties are increased. All the parameters are multiplied with 50%. In this case, the aircraft is no longer able to follow the trajectory even when the parameter re-identification is triggered and the simulation is terminated around $t = 5.7$ s, as shown in Figure 8.13(a). The control surface deflections are shown in Figure 8.13(b). It is seen from the figure that the elevator reaches its position limit.

RESULTS OF USING THE INDI

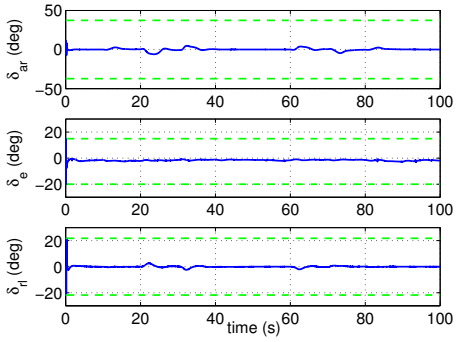
In contrast, the performances of using the INDI, when all the parameters are multiplied with 80% and 50% respectively, remain the same as the case when there is no model uncertainty and the results are not shown. This also demonstrates the benefits of the INDI. Its performance is still satisfactory when there are model uncertainties.

COMPARISON IN THE PRESENCE OF MODEL UNCERTAINTIES

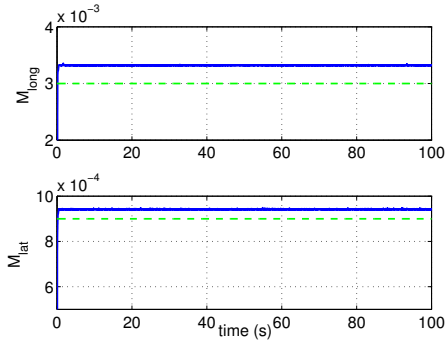
In this section, the performance of both controllers when there are model uncertainties is compared. It is shown that the performance of the controller using the NDI-MI depends on the chosen thresholds. The controller using the INDI does not require a threshold. When the initial parameters are multiplied with 80%, the performances of



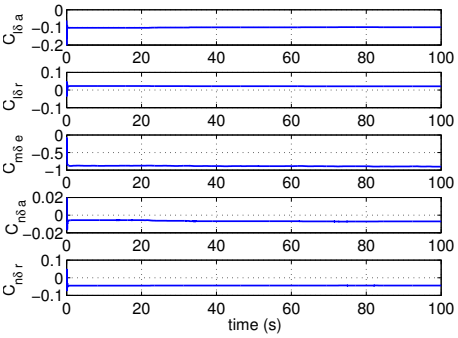
(a) Desired and true x, y and z



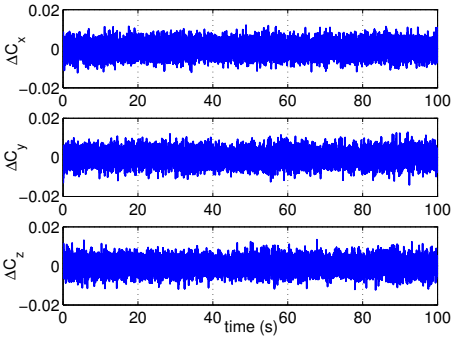
(b) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.



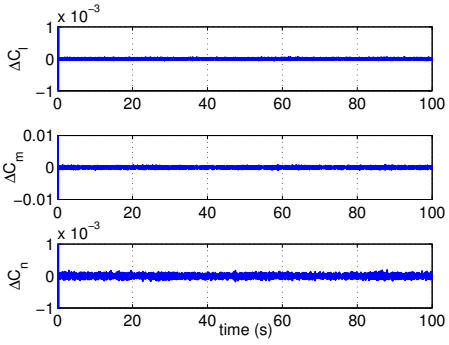
(c) M_{long} and M_{lat}, dashed lines represent the thresholds.



(d) Identified $C_{l\delta_a}$, $C_{l\delta_r}$, $C_{m\delta_e}$, $C_{n\delta_a}$ and $C_{n\delta_r}$

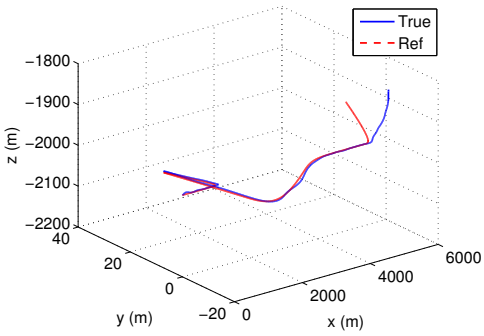


(e) ΔC_x, ΔC_y and ΔC_z

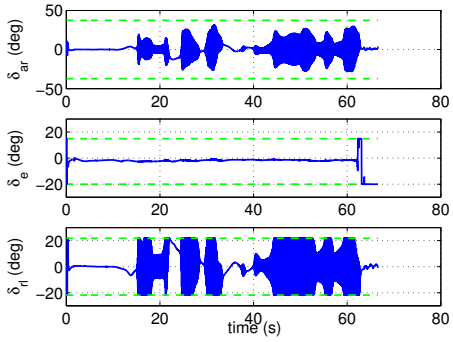


(f) ΔC_l, ΔC_m and ΔC_n

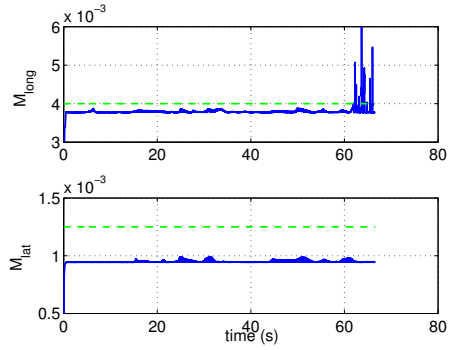
Figure 8.11: Results of aircraft trajectory control using the NDI-MI with thresholds (8.77) in the presence of model uncertainties.



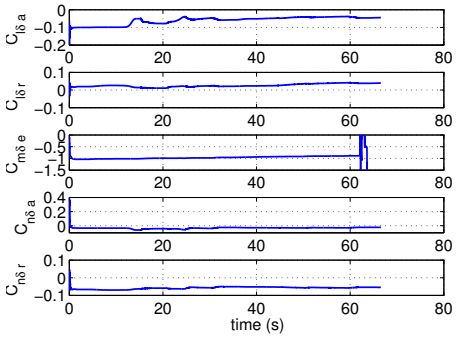
(a) Desired and true x, y and z



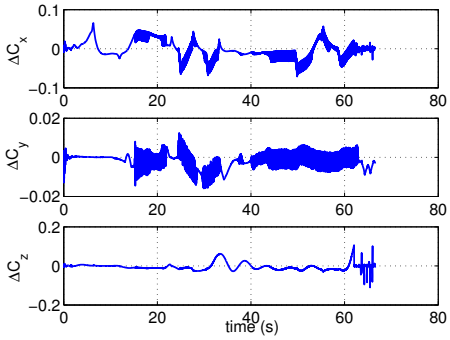
(b) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.



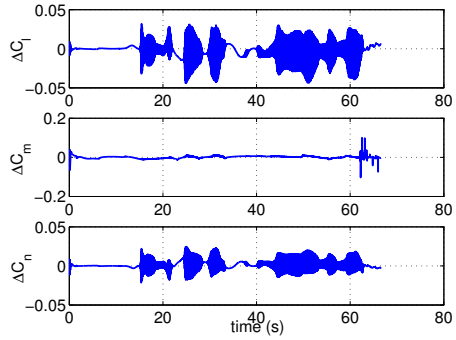
(c) M_{long} and M_{lat}, dashed lines represent the thresholds.



(d) Identified $C_{l\delta_a}$, $C_{l\delta_r}$, $C_{m\delta_e}$, $C_{n\delta_a}$ and $C_{n\delta_r}$

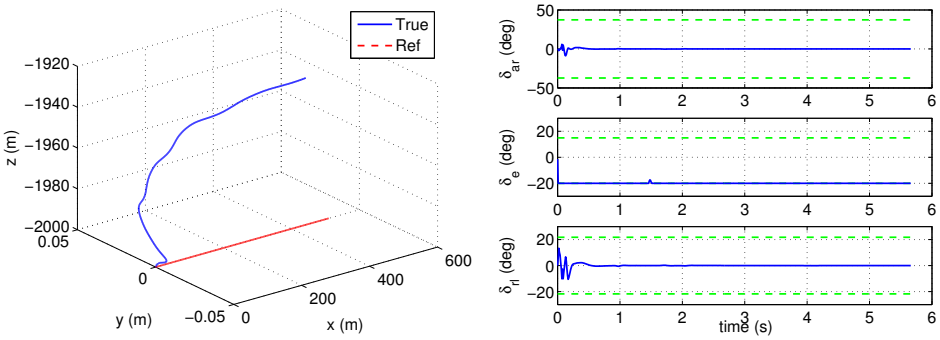


(e) ΔC_x, ΔC_y and ΔC_z



(f) ΔC_l, ΔC_m and ΔC_n

Figure 8.12: Results of aircraft trajectory control using the NDI-MI with thresholds (8.78) in the presence of model uncertainties.



(a) Desired and true x, y and z

(b) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.

Figure 8.13: Results of aircraft trajectory control using the NDI-MI in the presence of model uncertainties (parameters multiplied with 50%).

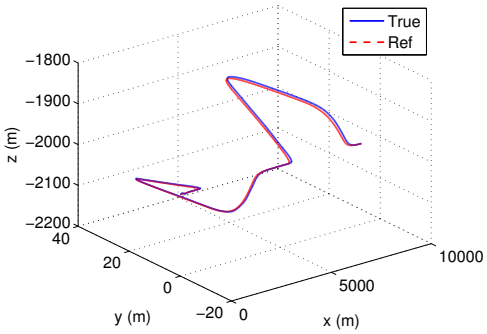
both controllers are similar. When the parameters are multiplied with 50%, the controller using the NDI-MI can no longer follow the reference whereas that using the INDI can still follow the reference well. This demonstrates one benefit of using the INDI: it does not require choosing a threshold.

8.6.4. VALIDATION IN THE PRESENCE OF ACTUATOR FAULTS

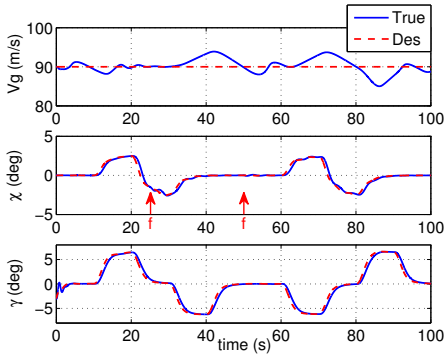
In this final scenario, a more challenging problem is considered. The model uncertainties are included in the parameters by multiplying each parameter with 50%. In addition, the actuators faults, which are given in Table 8.3, are also considered. Since the NDI-MI does not work when the parameters are all multiplied with 50%, its performance is not shown in this section.

The results using the INDI are shown in Figure 8.14. No significant difference between the trajectory and its reference is found in Figure 8.14(a). Its performance is comparable with the scenario when there are no faults or model uncertainties. However, slight differences and performance degradation can still be observed. At $t = 25$ s and $t = 50$ s, χ, μ are influenced by the faults, as can be seen from Figures 8.14(b) and 8.14(c). Especially, β is influenced. Two peaks are generated due to the faults. β, which is around 0.1 deg when there are no faults, reaches 0.6 deg in this scenario. p and r are also influenced and cannot follow their references well when the faults occur, as observed from Figure 8.14(d).

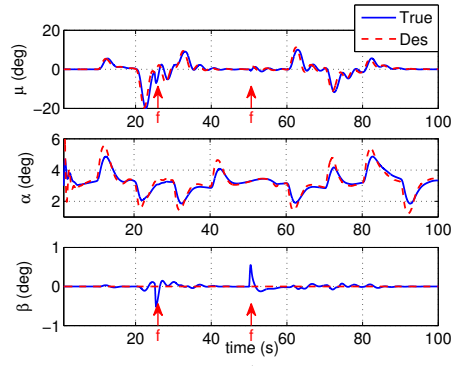
The control surface deflections δ_{al} and δ_{ru} are shown in Figure 8.14(e). It is seen that they are stuck at $t = 25$ s and $t = 50$ s respectively. The control surface deflections δ_{ar}, δ_e and δ_{rl}, which provide the remaining control authority for reconfiguration, are shown in Figure 8.14(f). In order to compensate the influence of the faults, δ_{ar} and δ_{rl} are controlled to deflect opposite angles of which δ_{al} and δ_{ru} are stuck at.



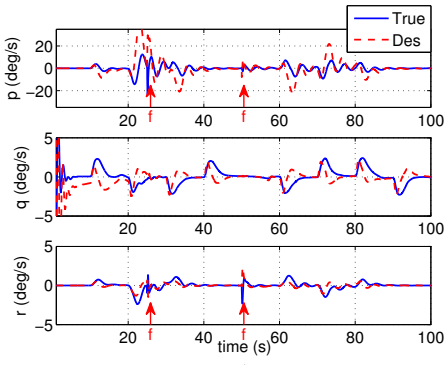
(a) Desired and true x, y and z



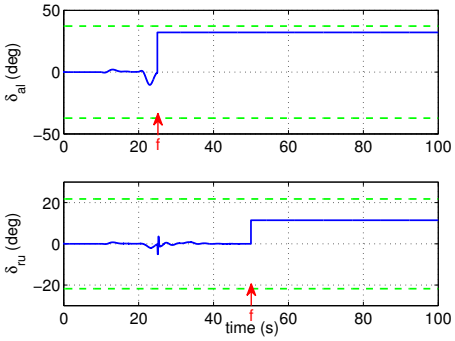
(b) Desired and true V_g, χ and γ



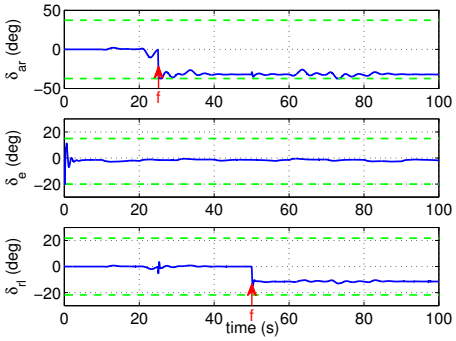
(c) Desired and true μ, α and β



(d) Desired and true p, q and r



(e) δ_{al} and δ_{ru}, dashed lines represent the position limits.



(f) δ_{ar}, δ_e and δ_{rl}, dashed lines represent the position limits.

Figure 8.14: Results of aircraft trajectory control using the INDI in the presence of model uncertainties and actuator faults.

8

8.7. CONCLUSIONS

This chapter proposes an aircraft fault-tolerant trajectory controller. The nonlinearities in the aircraft model are dealt with using a NDI approach. The uncertainties in the aircraft model are treated by the INDI approach. The trajectory controller is split into four control loops: position control, flight path control, attitude control and angular rate control. The detailed control law of the flight path control and the angular rate control loop, which use the INDI approach, are presented in the chapter. The other two loops, position control and attitude control loops are designed based on the standard NDI approach.

The performance of the proposed trajectory controller is compared to an existing control approach which is NDI-MI. Their performances are compared using three scenarios. The drawback of the NDI-MI is that it requires excitation and a selection of thresholds. The results of the comparison demonstrate the superior performance of the proposed controller which does not require sufficient excitation and choosing thresholds. Therefore, the proposed controller is more robust to model uncertainties in increasing the safety of the aircraft in the presence of actuator faults.

In the future, an experiment of the proposed approach on a fixed-wing unmanned aerial vehicle is highly recommended. Through doing this, the performance of the proposed approach can be validated under real-life model uncertainties and fault scenarios.

9

A Framework for Simultaneous Sensor and Actuator Fault-Tolerant Flight Control

In previous chapters, sensor and actuator FTC is considered. However, simultaneous sensor and FTC is not dealt with. Existing research only considers simultaneous sensor and actuator FDD. In this chapter, a framework is proposed to deal with simultaneous sensor and actuator FTC. The structure of the proposed controller is presented first, followed by the simulation validation.

Parts of this chapter are based on:

P Lu, E van Kampen, C C de Visser, Q P Chu. Nonlinear Simultaneous Aircraft Sensor and Actuator Fault-Tolerant Flight Control. *Journal of Guidance, Control and Dynamics*, (under review).

9.1. INTRODUCTION

Safety is of paramount importance in aviation, especially for civil aircraft. In civil aviation, many developments focus on improving of safety levels and reducing the risks that critical faults occur [171]. According to the report of the International Civil Aviation Organization, loss of control in-flight is one of the three high-risk accident categories [185]. Some recent accidents caused by actuator faults include [4] and [5]. To avoid these accidents, the flight control systems should be reconfigurable in the presence of actuator faults. This motivates the development of actuator FTC systems.

Apart from the actuator faults, recent airliner accidents indicate that sensor faults can result in critical failures despite the fact that most aircraft contain sensor redundancy [73]. The ADSs, which measure the airspeed, angle of attack and angle of sideslip, are exposed outside of the fuselage. This makes it prone to fail due to the external environment. Some recent accidents caused by sensor failures include [1] and [2]. These examples highlight the importance of sensor FDD systems. In the past few decades, many promising sensor FDD approaches have been proposed [11, 13, 14, 25, 65, 83].

Most sensor FDD approaches make use of the aerodynamic model of the aircraft, which makes their performance sensitive to model uncertainties. Furthermore, these FDD approaches are usually designed based on linear time invariant systems [186]. However, the aerodynamic forces and moments acting on the aircraft can change significantly under different flight conditions, which results in uncertainty in the model. In order to reduce the influence of model uncertainty during the whole flight, one solution is to design the LPV system. LPV models may explicitly contain information about the aerodynamic coefficient variations over the flight envelope [113]. However, the modeling of this LPV system can be time-consuming. Alternatively, the aircraft kinematic model [65, 83, 92, 114], which does not require the modeling of an LPV system, can be applied. The core of the kinematic model is that it uses measured aerodynamic forces to reduce model uncertainties. This model has been used for sensor FDD and shows great potential.

The FDD system provides more reliable state information for the FTC system. For actuator FTC systems, various reconfigurable control methods are also proposed for aerospace applications [11, 40]. [41] use model predictive control which can handle system input and output constraints. Sliding mode control [18] is also applied to FTC due to its fast convergence property [18]. H_∞ control approach has also been proposed to design FTC controllers [187]. Additionally, BS is one of the most popular Lyapunov-based control approaches which can be used for designing FTC systems. However, standard BS is sensitive to model uncertainties. To cope with model uncertainties, Adaptive Backstepping [60, 173] and Command Filtering Backstepping [61, 62] were proposed. More recently, Incremental Backstepping (IBS) [180, 188, 189], which can cope with model uncertainties, is proposed.

Although several sensor FDD systems and actuator FTC systems are proposed, research on simultaneous sensor and actuator FTC system is limited. One reason is that state-of-the-art sensor FDD systems make use of the actuator information such as commanded control surface deflections. This can be problematic when the actuator fails resulting in incorrect information of the actuator. Recently, Zhang and li [99], Marzat et. al [140] and Alwi et. al [186] consider simultaneous sensor and actuator FDD. However,

simultaneous sensor and actuator FTC is not dealt with. Furthermore, performance of existing sensor FDD systems is only shown in off-line simulation. In this chapter, simultaneous sensor and actuator FTC are considered. To the best of the authors' knowledge, the present chapter is one of the first few studies which consider simultaneous aircraft sensor and actuator FTC.

The present chapter aims at proposing a framework of design FTC systems which are able to deal with simultaneous sensor and actuator faults. The FTC system is composed of a sensor FDD system and a reconfigurable controller. The sensor FDD system is designed by making use of aircraft kinematic model which does not require the information of the actuator. The detection and estimation of the sensor faults are resolved using an ATS-UKF [65] which can estimate the state and fault in an unbiased sense. The unbiased state estimation is provided to the reconfigurable controller such that FTC in the presence of sensor faults is achieved. Regarding actuator and process faults, the IBS approach is applied to reconfigure the controller. By doing this, simultaneous sensor and actuator FTC is achieved.

The performance of the proposed FTC system is validated using simulation. The control objective is to control the attitude of the aircraft. An attitude controller and an angular rate controller are designed based on the IBS approach. The command filters are implemented to take the actuator physical limits into consideration. The aircraft sensor faults contain ADS and Attitude Heading and Reference System (AHRS) faults. The sensor fault types include bias, drift as well as oscillatory faults. The stuck actuator faults considered in this chapter. The FTC system is applied to address the sensor and actuator FTC simultaneously and maintain controlled flight.

9.2. SENSOR FAULT RECONSTRUCTION SYSTEM DESIGN

In this section, first the model used for sensor fault reconstruction is introduced in section 9.2.1. Then the detailed sensor fault reconstruction system design is presented in section 9.2.2.

9.2.1. NONLINEAR AIRCRAFT KINEMATIC MODEL INCLUDING ADS AND AHRS FAULTS

The model used for sensor fault reconstruction is the aircraft kinematic model. It should be noted that this model does not require the information of the actuator. The process

model including ADS and AHRS faults is presented as follows [65]:

$$\dot{V} = (A_{xm} - w_{Ax} - g\sin\theta)\cos\alpha\cos\beta + (A_{ym} - w_{Ay} + g\sin\phi\cos\theta)\sin\beta + (A_{zm} - w_{Az} + g\cos\phi\cos\theta)\sin\alpha\cos\beta \tag{9.1}$$

$$\dot{\alpha} = \frac{1}{V\cos\beta}\big[-(A_{xm} - w_{Ax})\sin\alpha + (A_{zm} - w_{Az})\cos\alpha + g\cos\phi\cos\theta\cos\alpha + g\sin\theta\sin\alpha\big] + q_m - w_q - [(p_m - w_p)\cos\alpha + (r_m - w_r)\sin\alpha]\tan\beta \tag{9.2}$$

$$\dot{\beta} = \frac{1}{V}\big[-(A_{xm} - w_{Ax} - g\sin\theta)\cos\alpha\sin\beta + (A_{ym} - w_{Ay} + g\sin\phi\cos\theta)\cos\beta - (A_{zm} - w_{Az} + g\cos\phi\cos\theta)\sin\alpha\sin\beta\big] + (p_m - w_p)\sin\alpha - (r_m - w_r)\cos\alpha \tag{9.3}$$

$$\dot{\phi} = (p_m - w_p) + (q_m - w_q)\sin\phi\tan\theta + (r_m - w_r)\cos\phi\tan\theta \tag{9.4}$$

$$\dot{\theta} = (q_m - w_q)\cos\phi - (r_m - w_r)\sin\phi \tag{9.5}$$

$$\dot{\psi} = (q_m - w_q)\frac{\sin\phi}{\cos\theta} + (r_m - w_r)\frac{\cos\phi}{\cos\theta} \tag{9.6}$$

Define:

$$\boldsymbol{x} = [V\ \alpha\ \beta\ \phi\ \theta\ \psi]^T,\ \boldsymbol{w} = [w_{Ax}\ w_{Ay}\ w_{Az}\ w_p\ w_q\ w_r]^T \tag{9.7}$$

$$\boldsymbol{u}_m = [A_{xm}\ A_{ym}\ A_{zm}\ p_m\ q_m\ r_m]^T = [A_x\ A_y\ A_z\ p\ q\ r]^T + \boldsymbol{w} \tag{9.8}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u}_m \in \mathbb{R}^l$ is the measured input vector. V is the true airspeed, α is the angle of attack, β is the angle of sideslip. ϕ, θ and ψ are the roll angle, pitch angle and yaw angle, respectively. The input vector, denoted as u_m, is the measurement of IMU sensors. The subscript "m" means that it is measured. A_x, A_y, A_z are the specific forces, p, q, and r are the roll rate, pitch rate and yaw rate, respectively. w is the noise in the IMU sensor with covariance matrix defined as: $E[\boldsymbol{w}(t)\boldsymbol{w}^T(t_\tau)] = \mathbf{Q}\delta(t-\tau)$.

The above process model can be rewritten into the following vector form:

$$\dot{\boldsymbol{x}}(t) = \bar{\boldsymbol{f}}(\boldsymbol{x}(t), \boldsymbol{u}_m(t), t) + \mathbf{G}(\boldsymbol{x}(t))\boldsymbol{w}(t) \tag{9.9}$$

Further define the following:

$$\boldsymbol{y}_m = [V_m\ \alpha_m\ \beta_m\ \phi_m\ \theta_m\ \psi_m]^T,\ \boldsymbol{v} = [v_V\ v_\alpha\ v_\beta\ v_\phi\ v_\theta\ v_\psi]^T \tag{9.10}$$

$$\boldsymbol{f}^o = [f_V\ f_\alpha\ f_\beta\ f_\phi\ f_\theta\ f_\psi]^T \tag{9.11}$$

where $\boldsymbol{y}_m \in \mathbb{R}^m$ is the measurement vector, $\boldsymbol{f}^o \in \mathbb{R}^p$ is the fault vector. $[f_V\ f_\alpha\ f_\beta]^T$ denote the faults in the ADSs and $[f_\phi\ f_\theta\ f_\psi]^T$ denote the faults in the AHRS. v is the output noise vector with $E[\boldsymbol{v}(t_i)\boldsymbol{v}^T(t_j)] = \mathbf{R}\delta(t_i - t_j)$.

The measurement model including the ADS and AHRS faults is:

$$\boldsymbol{y}_m(t) = \boldsymbol{h}(\boldsymbol{x}(t)) + \mathbf{F}(t)\boldsymbol{f}^o(t) + \boldsymbol{v}(t) \tag{9.12}$$

$$= \mathbf{H}(t)\boldsymbol{x}(t) + \mathbf{F}(t)\boldsymbol{f}^o(t) + \boldsymbol{v}(t),\ \ t = t_i,\ i = 1, 2, \ldots \tag{9.13}$$

where $\mathbf{H} = \mathbf{F} = \mathbf{I}_{6\times 6}$.

9.2.2. DESIGN OF THE SENSOR FAULT RECONSTRUCTION SYSTEM

The ATS-UKF differs from the UKF [27, 86] in that the measurement update is different. Furthermore, the ATS-UKF contains a fault detection and identification step.

To design the ATS-UKF, first the condition for fault reconstruction is checked. In this study, $m = 6$, $p = 6$ and rank $F_k = 6$. According to [65, 84], the condition is satisfied and an ATS-UKF can be designed. The detailed steps of estimating the states and ADS and AHRS faults using the ATS-UKF [65] is presented as follows:

Step1 Sigma points calculation and time update

At time step k, the following sigma points are generated [27]:

$$\mathscr{X}_{0,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} \tag{9.14a}$$

$$\mathscr{X}_{i,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} - \left(\sqrt{(n+\gamma_0)\mathbf{P}_{k-1|k-1}}\right)_i, \quad i = 1,2,...,n \tag{9.14b}$$

$$\mathscr{X}_{i,k-1} = \hat{\boldsymbol{x}}_{k-1|k-1} + \left(\sqrt{(n+\gamma_0)\mathbf{P}_{k-1|k-1}}\right)_{i-n}, \quad i = n+1, n+2,...,2n \tag{9.14c}$$

with $\mathscr{X}_{i,k-1}$ the sigma points of the state (dimension n). $\hat{\boldsymbol{x}}_{k-1|k-1}$ and $\mathbf{P}_{k-1|k-1}$ are the state estimation and covariance of estimation error at time step $k-1$. $\boldsymbol{\gamma}_0$ is a scaling factor [86].

The predicted mean and covariance are computed as follows

$$\mathscr{X}_{i,k|k-1} = \mathscr{X}_{i,k-1} + \int_{k-1}^{k} \bar{f}(\mathscr{X}_{i,k-1}, \boldsymbol{u}(t), t)\,dt \tag{9.15}$$

$$\hat{\boldsymbol{x}}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(m)} \mathscr{X}_{i,k|k-1} \tag{9.16}$$

$$\mathbf{P}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(c)} [\mathscr{X}_{i,k|k-1} - \hat{\boldsymbol{x}}_{k|k-1}][\mathscr{X}_{i,k|k-1} - \hat{\boldsymbol{x}}_{k|k-1}]^T + \mathbf{Q}_d \tag{9.17}$$

$$\mathscr{Y}_{i,k|k-1} = \boldsymbol{h}(\mathscr{X}_{i,k|k-1}) \tag{9.18}$$

$$\hat{\boldsymbol{y}}_k = \sum_{i=0}^{2n} w_i^{(m)} \mathscr{Y}_{i,k|k-1} \tag{9.19}$$

$$\mathbf{P}_{xy,k} = \sum_{i=0}^{2n} w_i^{(c)} [\mathscr{X}_{i,k|k-1} - \hat{\boldsymbol{x}}_{k|k-1}][\mathscr{Y}_{i,k|k-1} - \hat{\boldsymbol{y}}_k]^T \tag{9.20}$$

$$\mathbf{P}_{yy,k} = \sum_{i=0}^{2n} w_i^{(c)} [\mathscr{Y}_{i,k|k-1} - \hat{\boldsymbol{y}}_k][\mathscr{Y}_{i,k|k-1} - \hat{\boldsymbol{y}}_k]^T + \mathbf{R} \tag{9.21}$$

where $w_i^{(m)}$ and $w_i^{(c)}$ are the weights which can be found in [65, 86]. $\hat{y}_k$ is the estimate of the measurements. $\mathbf{Q}_d$ is approximated by $\mathbf{G}(\hat{x}_{k|k-1})\mathbf{Q}\mathbf{G}(\hat{\boldsymbol{x}}_{k|k-1})^T\Delta t$ and Δt is the time step.

Step2 Detection, isolation and estimation of the faults

Define $\mathbf{C}_k$ as follows:

$$\mathbf{C}_k := \frac{1}{N} \sum_{j=k-N+1}^{k} \boldsymbol{\gamma}_j \boldsymbol{\gamma}_j^T \tag{9.22}$$

where $\boldsymbol{\gamma}_j = (\boldsymbol{y}_j - \hat{\boldsymbol{y}}_j)$, denotes the innovation at time step j.

Let $C_{ii,k}$, $i = 1,2,...,m$ denote the ith diagonal elements of $\mathbf{C}_k$ at time step k. The fault detection and isolation is performed through the following:

$$\text{If } C_{ii,k} > T_i,\ F_{A_i} = 1.\ \text{otherwise } F_{A_i} = 0,\ \text{i = 1,2,...,m.}$$

where $\boldsymbol{F}_A = [F_{A_V}\ F_{A_\alpha}\ F_{A_\beta}\ F_{A_\phi}\ F_{A_\theta}\ F_{A_\psi}]^T$ are the binary alarm indicators. $\boldsymbol{T}_i$ are the thresholds which are designed to detect the faults in the ADS and AHRS respectively. These thresholds are designed based on fault-free cases.

To cope with initial condition errors, we need to calculate the change of the innovation covariance $\Delta C_{ii,k}$ as follow:[65]

$$\Delta C_{ii,k} := C_{ii,k} - C_{ii,k-1}, \quad i = 1,2,...,m. \tag{9.23}$$

When the following inequality holds, the initial measurement update can be regarded as sufficient. The inequality is

$$\Delta C_{ii,k} < \eta_i, \ \ i = 1,2,...,m. \tag{9.24}$$

where η_i, $i = 1,2,...,m$ are pre-defined constants which can be tuned to stop the initial measurement update. It should be noted that this initial measurement update is only performed in the first few time steps denoted by k^*. In this chapter, $k^* = 20$.

If either of the following two conditions is satisfied:

(a) There are no faults detected (i.e., $\|F_A\| = 0$)

(b) $k < k^*$ and $\Delta C_{ii,k} > \eta_i$ for any i

the estimates of the fault and its error covariance matrix are:

$$\hat{\boldsymbol{f}}_k^o = \mathbf{0},\ \mathbf{P}_k^f = \mathbf{0} \tag{9.25}$$

$$\mathbf{K}_k = \mathbf{P}_{xy,k}\mathbf{P}_{yy,k}^{-1} \tag{9.26}$$

If neither of conditions (a) and (b) is satisfied, the estimates of the fault and its error covariance matrix are calculated as follows:

$$\mathbf{N}_k = (\mathbf{F}_k^T\mathbf{P}_{yy,k}^{-1}\mathbf{F}_k)^{-1}\mathbf{F}_k^T\mathbf{P}_{yy,k}^{-1} \tag{9.27}$$

$$\hat{\boldsymbol{f}}_k^o = \mathbf{N}_k\gamma_k,\ \mathbf{P}_k^f = (\mathbf{F}_k^T\mathbf{P}_{yy,k}^{-1}\mathbf{F}_k)^{-1} \tag{9.28}$$

$$\mathbf{K}_k = \mathbf{P}_{xy,k}\mathbf{P}_{yy,k}^{-1}(\mathbf{I}_{6\times 6} - \mathbf{F}_k\mathbf{N}_k) \tag{9.29}$$

where $\boldsymbol{\gamma}_k$ is the innovation at time step k, $\hat{\boldsymbol{f}}_k^o$ is the estimation of $\boldsymbol{f}_k^o$ and $\mathbf{P}_k^f$ is its error covariance matrix. $\mathbf{N}_k$ is the gain matrix which can achieve an unbiased estimation of $\boldsymbol{f}_k^o$.

Step3 Measurement update

$$\mathbf{K}_k = \mathbf{P}_{xy,k}\mathbf{P}_{yy,k}^{-1} \tag{9.30}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \mathbf{K}_k(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k - \mathbf{F}_k\hat{f}_k^o) \tag{9.31}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k(\mathbf{P}_{yy,k} - \mathbf{F}_k\mathbf{P}_k^f\mathbf{F}_k^T)\mathbf{K}_k^T \tag{9.32}$$

By substituting the functions $\bar{\boldsymbol{f}}$ and $\boldsymbol{h}$ given in Equations (9.9) and (9.12) into Equations (9.15) and (9.18) respectively, the fault estimation and state estimation can be achieved which are given in Equations (9.28) (or (9.25)) and (9.31) respectively. The state and fault estimation results are used by the controller which is designed in the following section.

9.3. RECONFIGURABLE CONTROL: INCREMENTAL BACKSTEPPING

This section will introduce the reconfigurable control approach for the actuator FTC system. First, in section 9.3.1, the dynamics of the aircraft attitude and angular rates are presented. Then, the design of the IBS controller for actuator FTC is presented in section 9.3.2.

9.3.1. AIRCRAFT ATTITUDE AND ANGULAR RATE DYNAMICS

In this section, the attitude dynamics and angular rate dynamics of the aircraft is presented, which is the model used for designing the IBS controller.

The kinematics of the Euler angles are:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \end{bmatrix} \boldsymbol{\omega} \tag{9.33}$$

where $\boldsymbol{\omega} = [p\ q\ r]^T$ are the rotational rates of the aircraft in the body reference frame. Furthermore, the sideslip angle β has to be kept at zero. The dynamics of the sideslip angle are given in Equation (9.3).

Ignore the noise in the accelerometers, Equations (9.33) and (9.3) can be rewritten into the following affine-in-control form:

$$\dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_2 \tag{9.34}$$

where

$$\begin{aligned}
&\boldsymbol{x}_1 = [\phi, \theta, \beta]^T, \boldsymbol{x}_2 = \boldsymbol{\omega}, \\
&\boldsymbol{f}_1 = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{V}\left[-(A_{xm} - g\mathrm{s}\theta)\mathrm{c}\alpha\mathrm{s}\beta + (A_{ym} + g\mathrm{s}\phi\mathrm{c}\theta)\mathrm{c}\beta - (A_{zm} + g\mathrm{c}\phi\mathrm{c}\theta)\mathrm{s}\alpha\mathrm{s}\beta\right] \end{bmatrix} \\
&\boldsymbol{g}_1 = \begin{bmatrix} 1 & \mathrm{s}\phi\mathrm{t}\theta & \mathrm{c}\phi\mathrm{t}\theta \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ \mathrm{s}\alpha & 0 & -\mathrm{c}\alpha \end{bmatrix}
\end{aligned} \tag{9.35}$$

where s(•), c(•) and t(•) denote the trigonometric functions sin(•), cos(•) and tan(•) respectively.

The angular rate dynamics of the aircraft are:

$$\dot{\boldsymbol{\omega}} = \mathbf{J}^{-1}(\boldsymbol{M} - \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega}) \tag{9.36}$$

where $\mathbf{J}$ is the inertia tensor. $\boldsymbol{M}$ are the moments acting on the aircraft which can be described as:

$$\boldsymbol{M} = \boldsymbol{M}_a + \boldsymbol{M}_u = \boldsymbol{M}_a + \mathbf{C}_{M_u}\boldsymbol{u} \tag{9.37}$$

where $\boldsymbol{M}_a$ are the moments except for the moments generated by the control surface deflections. $\boldsymbol{M}_u$ are the moments generated by the control surface deflections. $\mathbf{C}_{\boldsymbol{M}_u}$ are the coefficients related to $\boldsymbol{M}_u$, which are denoted as

$$M_a = \bar{q}SC_1 \begin{bmatrix} C_l(\beta, p, r, M_a) \\ C_m(\alpha, \dot{\alpha}, q, M_a) \\ C_n(\beta, p, r, M_a) \end{bmatrix}, \mathbf{C}_{\boldsymbol{M}_u} = \bar{q}SC_1 \begin{bmatrix} C_{l_{\delta a}} & 0 & C_{l_{\delta r}} \\ 0 & C_{m_{\delta e}} & 0 \\ C_{n_{\delta a}} & 0 & C_{n_{\delta r}} \end{bmatrix}, \mathbf{C}_1 = \mathrm{diag}(b, \bar{c}, b) \tag{9.38}$$

where $\bar{q}$ is the dynamic pressure, M_a is the Mach number, S is the wing area, b the wing span and $\bar{c}$ is the mean aerodynamic chord. The control surface deflections are $\boldsymbol{u} = [\delta_a, \delta_e, \delta_r]^T$, which are also the input to the system.

We can rewrite Equation (9.36) into the following affine-in-control form:

$$\dot{\boldsymbol{x}}_2 = \boldsymbol{f}_2(\boldsymbol{x}_1, \boldsymbol{x}_2) + \boldsymbol{g}_2\boldsymbol{u} \tag{9.39}$$

where $\boldsymbol{f}_2 = \mathbf{J}^{-1}(\boldsymbol{M}_a - \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega})$, $\boldsymbol{g}_2 = \mathbf{J}^{-1}\mathbf{C}_{\boldsymbol{M}_u}$.

9.3.2. INCREMENTAL BACKSTEPPING CONTROLLER DESIGN

In order to design the IBS controller, we need to rewrite Equation (9.34) into an incremental form. Denote the actuator deflections in the previous time step as u_0 and the

incremental deflections in the current step as Δu, then the actuator deflections in the current step are:

$$\boldsymbol{u} = \boldsymbol{u}_0 + \Delta \boldsymbol{u} \tag{9.40}$$

where $\boldsymbol{u}_0 = [\delta_{a0}, \delta_{e0}, \delta_{r0}]^T$ and $\Delta \boldsymbol{u} = [\Delta\delta_a, \Delta\delta_e, \Delta\delta_r]^T$. $\boldsymbol{u}_0$ can be obtained using the actuator model [189]. According to [176, 180], Equation (9.39) can be rewritten into the following:

$$\dot{\boldsymbol{x}}_2 = \dot{\boldsymbol{x}}_{2,0} + \boldsymbol{g}_2 \Delta \boldsymbol{u} \tag{9.41}$$

where $\dot{\boldsymbol{x}}_{2,0}$, the derivative of the angular rates in the previous time step, is defined as:

$$\dot{\boldsymbol{x}}_{2,0} := \boldsymbol{f}_2(\boldsymbol{x}_{10}, \boldsymbol{x}_{20}) + \boldsymbol{g}_2 \boldsymbol{u}_0 \tag{9.42}$$

According to [176], $\dot{\boldsymbol{x}}_{2,0}$ can be computed by passing $\boldsymbol{\omega}_m$ ($[p_m, q_m, r_m]^T$) through the following filter:

$$\frac{s\omega_n^2}{s^2 + 2\zeta_n \omega_n s + \omega_n^2} \tag{9.43}$$

with $\zeta_n = 0.8$, $\omega_n = 25$ rad/s which are chosen by trial and error to reduce the influence of noise [176].

Now the complete model used for the controller design can be given as follows:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_2 & (9.44) \\ \dot{\boldsymbol{x}}_2 = \dot{\boldsymbol{x}}_{2,0} + \boldsymbol{g}_2 \Delta \boldsymbol{u} & (9.45) \end{cases}$$

Now we can present the design of the controller based on the model Equations (9.44) and (9.45). The control task is to steer $\boldsymbol{x}_1$ towards a given reference $\boldsymbol{y}_r = [\phi^{ref}, \theta^{ref}, \beta^{ref}]^T$ with bounded and known derivatives ($\dot{\boldsymbol{y}}_r$, $\ddot{\boldsymbol{y}}_r$, ...). Define the following tracking errors:

$$\boldsymbol{z}_1 = \boldsymbol{x}_1 - \boldsymbol{y}_r \tag{9.46}$$

$$\boldsymbol{z}_2 = \boldsymbol{x}_2 - \boldsymbol{\alpha}_1 \tag{9.47}$$

where $\boldsymbol{\alpha}_1$ is the virtual control for the state $\boldsymbol{x}_2$. The derivatives of Equations (9.46) and (9.47) are:

$$\dot{\boldsymbol{z}}_1 = \dot{\boldsymbol{x}}_1 - \dot{\boldsymbol{y}}_r \tag{9.48}$$

$$\dot{\boldsymbol{z}}_2 = \dot{\boldsymbol{x}}_2 - \dot{\boldsymbol{\alpha}}_1 \tag{9.49}$$

Substituting Equations (9.44) and (9.45) into Equations(9.48) and (9.49), it follows that

$$\dot{\boldsymbol{z}}_1 = \boldsymbol{f}_1(\boldsymbol{x}_1) + \boldsymbol{g}_1(\boldsymbol{x}_1)\boldsymbol{x}_2 - \dot{\boldsymbol{y}}_r \tag{9.50}$$

$$\dot{\boldsymbol{z}}_2 = \dot{\boldsymbol{x}}_{2,0} + \boldsymbol{g}_2 \Delta \boldsymbol{u} - \dot{\boldsymbol{\alpha}}_1 \tag{9.51}$$

Define $\bar{\boldsymbol{z}}_1 = \boldsymbol{z}_1 - \boldsymbol{\chi}_1$ and $\bar{\boldsymbol{z}}_2 = \boldsymbol{z}_2 - \boldsymbol{\chi}_2$. By using the control Lyapunov function $V = \frac{1}{2}\bar{\boldsymbol{z}}_1^T \bar{\boldsymbol{z}}_1 + \frac{1}{2}\bar{\boldsymbol{z}}_2^T \bar{\boldsymbol{z}}_2$, the desired control laws are defined as:

$$\boldsymbol{\alpha}_1 = \boldsymbol{g}_1^{-1}(-c_1 \boldsymbol{z}_1 - \boldsymbol{f}_1(\boldsymbol{x}_1) + \dot{\boldsymbol{y}}_r) \tag{9.52}$$

$$\Delta \boldsymbol{u}^{des,0} = \boldsymbol{g}_2^{-1}(-\boldsymbol{g}_1(\boldsymbol{x}_1)\bar{\boldsymbol{z}}_1 - c_2 \boldsymbol{z}_2 + \dot{\boldsymbol{x}}_2^{des} - \dot{\boldsymbol{x}}_{2,0}) \tag{9.53}$$

where c_1 and c_2 are the controller gains. $\dot{\boldsymbol{x}}_2^{des}$ is the filtered derivative of $\boldsymbol{x}_2^{des,0}$ which is defined as:

$$\boldsymbol{x}_2^{des,0} = [p^{des,0}, q^{des,0}, r^{des,0}]^T = \alpha_1 - \chi_2 \tag{9.54}$$

$\boldsymbol{\chi}_1$ and $\boldsymbol{\chi}_2$ are computed as:

$$\dot{\boldsymbol{\chi}}_1 = -c_1\boldsymbol{\chi}_1 + \boldsymbol{g}_1(\boldsymbol{x}_2^{des} - \boldsymbol{x}_2^{des,0}) \tag{9.55}$$

$$\dot{\boldsymbol{\chi}}_2 = -c_2\boldsymbol{\chi}_2 + \boldsymbol{g}_2(\Delta\boldsymbol{u}^{des} - \Delta\boldsymbol{u}^{des,0}) \tag{9.56}$$

Define the following variables:

$$\boldsymbol{x}_2^{des,0} = [p^{des,0}, q^{des,0}, r^{des,0}]^T,\ \boldsymbol{x}_2^{des} = [p^{des}, q^{des}, r^{des}]^T \tag{9.57}$$

$$\Delta\boldsymbol{u}^{des,0} = [\Delta\delta_a^{des,0}, \Delta\delta_e^{des,0}, \Delta\delta_r^{des,0}]^T,\ \Delta\boldsymbol{u}^{des} = [\Delta\delta_a^{des}, \Delta\delta_e^{des}, \Delta\delta_r^{des}]^T \tag{9.58}$$

$\boldsymbol{x}_2^{des,0}$ is filtered through second-order command filters [61, 63, 180] to compute $\boldsymbol{x}_2^{des}$ and $\dot{\boldsymbol{x}}_2^{des}$. $\Delta\boldsymbol{u}^{des,0}$ is filtered through second-order command filters to compute $\Delta\boldsymbol{u}^{des}$. Take $\boldsymbol{x}_2^{des,0}$ for example, the command filter is as follows [61–63]:

$$\begin{bmatrix}\dot{\boldsymbol{q}}_1\\ \dot{\boldsymbol{q}}_2\end{bmatrix} = \begin{bmatrix}\boldsymbol{q}_2\\ 2\zeta_c\omega_c\left[S_R(\frac{\omega_c^2}{2\zeta_c\omega_c}[S_P(\boldsymbol{x}_2^{des,0}) - \boldsymbol{q}_1]) - \boldsymbol{q}_2\right]\end{bmatrix} \tag{9.59}$$

where S_P and S_R are position and rate limit functions which can be defined as in Equation (9.60) [61–63]:

$$S_P(z) = \begin{cases} P, \text{ if } z \le P \\ z, \text{ if } |z| < P \\ -P, \text{ if } z \ge -P \end{cases} \tag{9.60}$$

Then $\boldsymbol{x}_2^{des} = \boldsymbol{q}_1$ and $\dot{\boldsymbol{x}}_2^{des} = \boldsymbol{q}_2$.

The final desired control input $\boldsymbol{u}^{des}$ is given by:

$$\boldsymbol{u}^{des} = [\delta_a^{des}, \delta_e^{des}, \delta_r^{des}]^T = \boldsymbol{u}_0 + \Delta\boldsymbol{u}^{des} \tag{9.61}$$

This concludes the design for the IBS for the feedback system. Note that there are only two loops in this design. More outer loops can be added outside the outer loop.

The IBS approach is robust to the uncertainties in the plant dynamics term since it does not require the information of the plant dynamics. The robustness with respect to the uncertainties in the control effectiveness is analyzed in [180].

9.4. FAULT-TOLERANT CONTROL SYSTEM FOR DEALING WITH SIMULTANEOUS SENSOR AND ACTUATOR FAULTS

The proposed FTC system for dealing with simultaneous sensor and actuator faults is presented in Figure 9.1. As can be seen from the figure, there are two control loops: attitude control loop and rate control loop. The attitude controller follows the commands

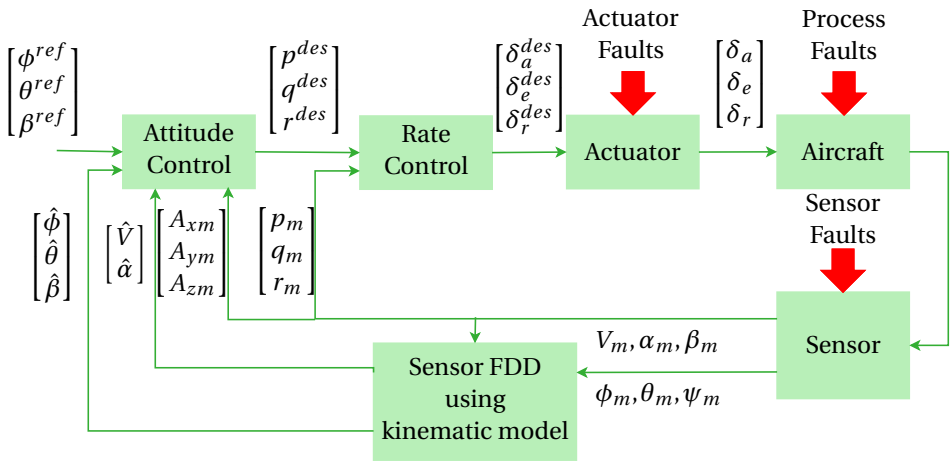


Figure 9.1: Block diagram of the FTC system in the presence of sensor and actuator faults

ϕ^{ref}, θ^{ref} and β^{ref} and is designed based on the BS control law. The attitude control loop generates the reference commands p^{des}, q^{des} and r^{des} for the angular rate control loop. The rate control loop is designed based on the IBS approach and it generates the command for the control surface deflections denoted as δ_a^{des}, δ_e^{des} and δ_r^{des}. The actual control surface deflections are denoted as δ_a, δ_e and δ_r.

The key point of this simultaneous sensor and actuator FTC system is that the sensor FDD makes use of the kinematic model. In Figure 9.1, it is seen that the sensor FDD system does not require any information about the actuators such as commanded actuator deflections. In most aircraft sensor FDD system designs, sensor FDD is achieved using the commanded or measured control surface deflections. In some cases, measurements of control surfaces are not available and are derived from the actuator rod positions [39]. Wrong actuator deflections can be obtained when the control surface is disconnected from the actuator rod. Consequently, it can lead to performance degradation of the sensor FDD system.

Reconfigurable controllers can deal with actuator faults. However, it is difficult for these controllers to deal with sensor faults. Therefore, the sensor FDD plays a critical role in this FTC system. The sensor FDD system not only provides unbiased fault estimation ($\hat{f}_V$, $\hat{f}_\alpha$, $\hat{f}_\beta$, and $\hat{f}_\phi$, $\hat{f}_\theta$, $\hat{f}_\psi$) but also unbiased state estimation ($\hat{V}$, $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\phi}$, $\hat{\theta}$, $\hat{\psi}$) such that the performance of the actuator FTC system is not influenced by sensor faults.

9.5. SIMULATION EXAMPLES

In this section, the proposed FTC system against sensor and actuator faults will be validated. In Section 9.5.1, the aircraft model and the fault scenarios are presented. The design parameters used in the simulation are presented in Section 9.5.2. The performance of the sensor FDD system compared to that without the sensor FDD system in the presence of sensor faults is shown in Section 9.5.3. The performance of the FTC sys-

tem in the presence of simultaneous sensor and actuator faults is shown in Section 9.5.4. Some discussions are presented in Section 9.5.5.

9.5.1. AIRCRAFT MODEL AND FAULT SCENARIO

The aircraft model used in this chapter is a generic aircraft [65]. The actuators are modeled as first-order low-pass filters. The control surfaces are left aileron, right aileron, upper rudder, lower rudder and elevator and the deflections are denoted as δ_{al}, δ_{ar}, δ_{ru}, δ_{rl} and δ_e respectively. The desired commands of the ailerons and rudders are as follows:

$$\delta_{al}^{des} = \delta_{ar}^{des} = \delta_a^{des} \tag{9.62}$$

$$\delta_{ru}^{des} = \delta_{rl}^{des} = \delta_r^{des} \tag{9.63}$$

This chapter considers both sensor faults and actuator faults. The sensor measurements contain noise. The noise covariances of the sensors in the aircraft can be found in [65]. Sensor faults include ADS sensor faults and AHRS faults. All the sensor faults occur simultaneously. The fault scenario of the ADS and AHRS faults is given in Table 9.1 respectively. The actuator faults considered in this chapter are stuck at a position fault. The specific fault occurrence time and stuck position is given in Table 9.2.

Table 9.1: Simultaneous ADS and AHRS faults

Time interval	Sensor fault	Fault type	Fault magnitude	Fault unit
20 s $< t <$ 40 s	f_V	oscillatory	$10\sin(0.5\pi t)$	m/s
	f_α	drift	$0.01t$	[rad]
	f_β	oscillatory	$5\pi\sin(0.5\pi t)/180$	[rad]
	f_ϕ	oscillatory	$2\pi\sin(\pi t)/180$	[rad]
	f_θ	bias	0.2	[rad]
	f_ψ	drift	$0.005t$	[rad]
60 s $< t <$ 80 s	f_V	drift	$-0.5t$	[m/s]
	f_α	drift	$-0.01t$	[rad]
	f_β	oscillatory	$5\pi\sin(\pi t)/180$	[rad]
	f_ϕ	bias	$-2\pi/180$	[rad]
	f_θ	drift	$-0.005t$	[rad]
	f_ψ	oscillatory	$2\pi\sin(\pi t)/180$	[rad]

It can be seen that during 25 s $< t <$ 40 s, the left aileron and the ADS and the AHRS fail. During 75 s $< t <$ 80 s, the left aileron, the upper rudder and the ADS and AHRS all fail. Therefore, simultaneous sensor and actuator faults are considered in this chapter.

Table 9.2: Actuator faults

Time interval	Actuator	Fault type	Stuck position	Fault unit
$t > 25$ s	left aileron	Jamming	0.56	[rad]
$t > 75$ s	upper rudder	Jamming	0.2	[rad]

9.5.2. FTC SYSTEM DESIGN PARAMETERS

In this section, the parameters used for the design of the FTC system is presented. For the sensor FDD, η, which is used in Equation (9.24) to terminate the initial measurement update, is chosen as:

$$\eta = [2\times10^{-2}, 1\times10^{-4}, 1\times10^{-5}, 1\times10^{-5}, 1\times10^{-5}, 1\times10^{-5}]^T \tag{9.64}$$

The threshold T used for fault detection is chosen as:

$$T = [8\times10^{-1}, 1\times10^{-4}, 1\times10^{-4}, 5\times10^{-4}, 1\times10^{-4}, 2\times10^{-4}]^T \tag{9.65}$$

For the IBS controller, the control gains are chosen as:

$$c_1 = \text{diag}(5,5,5), \qquad c_2 = \text{diag}(2,2,2) \tag{9.66}$$

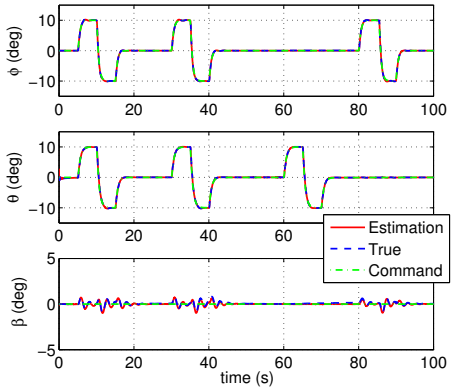
9.5.3. VALIDATION OF THE FTC SYSTEM IN THE PRESENCE OF SENSOR FAULTS

In this section, the performance of the FTC system in the presence of sensor faults is validated. The sensor faults are given in Table 9.1. It is seen that during 20 s $< t <$ 40 s and 60 s $< t <$ 80 s, all the ADS and AHRS fail simultaneously. The sensor FDD system of the FTC system will be used to reconstruct the sensor faults in order to provide unbiased state estimation for the feedback.

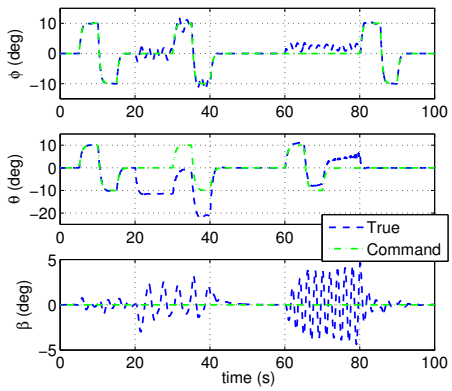
For comparison, the IBS controller without the sensor FDD system is also applied. For this controller, the measurements are directly used by the controller. In this case, the state feedback used by the controller is biased due to the sensor faults.

The results of the IBS controller with and without using the sensor FDD system are given in Figures 9.2(a) and 9.2(b) respectively. As can be seen from Figure 9.2(a), even in the presence of sensor faults, ϕ, θ and β of the aircraft can still follow the reference commands well. In contrast, when the sensor FDD is not used, ϕ, θ and β of the aircraft can not follow the reference commands when there are sensor faults, as shown in Figure 9.2(b). The response of the aircraft is obviously influenced by the wrong sensor measurements. For example, during 20 s $< t <$ 30 s, the reference command for θ is zero. But the aircraft pitches down to -10 deg. This is caused by the θ sensor fault, which is a bias fault. Furthermore, during 60 s $< t <$ 80 s, β reaches 5 deg even when there are no lateral maneuvers. This is not desirable for civil aircraft. The results demonstrate the importance of a sensor FDD system.

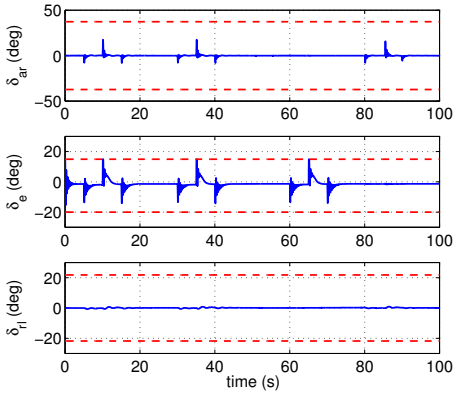
The state estimation performance of the sensor FDD system is shown in Figure 9.2(a). It is seen that the solid lines can still follow the dashed lines closely. This demonstrates that the sensor FDD system can provide unbiased state estimation even in the presence of sensor faults.



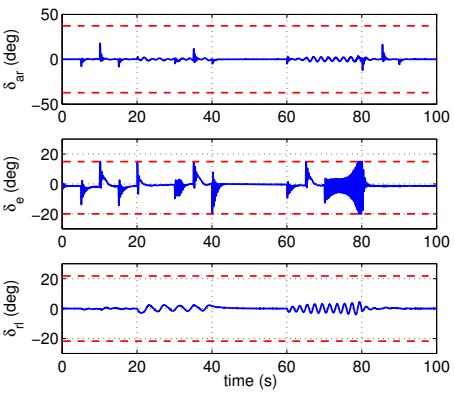
(a) Estimated, true and commanded ϕ, θ and β in the presence of sensor faults, with the sensor FDD system



(b) True and commanded ϕ, θ and β in the presence of sensor faults, without the sensor FDD system



(c) Actuator deflections in the presence of sensor faults with the sensor FDD system. Dashed lines denote the actuator maximum and minimum position limits

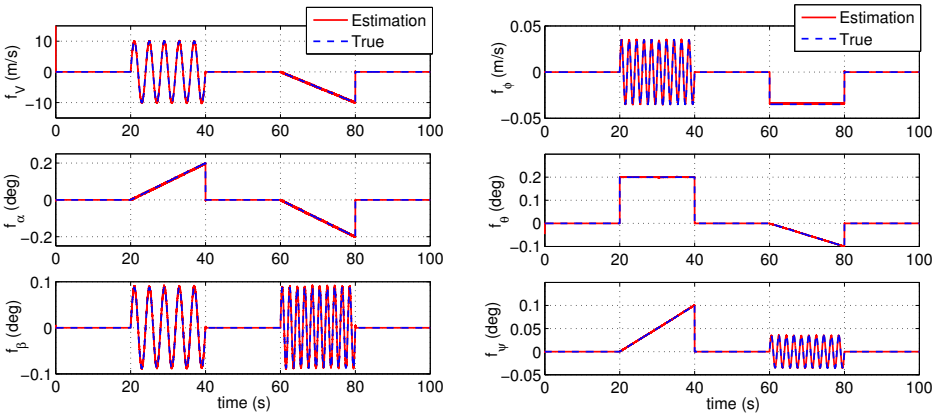


(d) Actuator deflections in the presence of sensor faults without the sensor FDD system. Dashed lines denote the actuator maximum and minimum position limits

Figure 9.2: Results of the IBS control with and without the sensor FDD system in the presence of sensor faults

The actuator control surface deflections δ_{ar}, δ_e and δ_{rl} of the FTC system with the sensor FDD system are given in Figure 9.2(c). It is seen that all the control surface deflections are within the actuator position limits. The control surface deflections δ_{ar}, δ_e and δ_{rl} of the FTC system without using the sensor FDD system is shown in Figure 9.2(d). As can be seen from the figure, the control surfaces oscillate in the presence of sensor faults. During 70 s $< t <$ 80 s, the elevator oscillates frequently, which can even damage the elevator itself.

For the FTC system with sensor FDD, sensor FTC is achieved. The fault reconstruction results using the sensor FDD system are given in Figures 9.3(a) and 9.3(b). It is seen



(a) True and estimated f_V, f_α and f_β in the presence of sensor faults

(b) True and estimated f_ϕ, f_θ and f_ψ in the presence of sensor faults

Figure 9.3: Fault reconstruction of the sensor FDD system in the presence of sensor faults

from the figure, although the fault types are different, all faults are reconstructed in an unbiased sense. The oscillatory frequency of the oscillatory faults are also different. This demonstrates the performance of the sensor FDD system which makes use of the ATS-UKF.

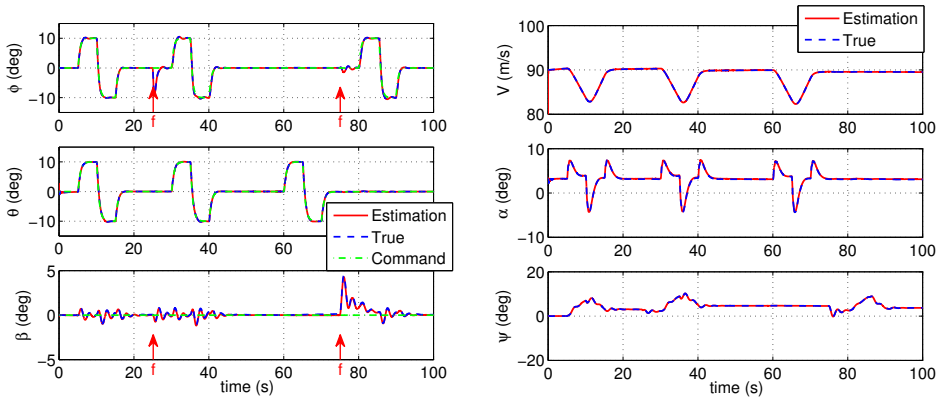
9.5.4. VALIDATION OF THE FTC SYSTEM IN THE PRESENCE OF SIMULTANEOUS SENSOR AND ACTUATOR FAULTS

In this section, the performance of the FTC system in the presence of simultaneous sensor and actuator faults is validated. The sensor faults are given in Table 9.1 and the actuator faults are given in Table 9.2. The sensor faults are coped with by the sensor FDD system. The actuator faults are dealt with by the IBS approach. The results of the FTC system are given in Figure 9.4.

The response of ϕ, θ and β of the aircraft using the FTC system is shown in Figure 9.4(a). As can be seen from the figure, when the actuators are stuck, ϕ and β will be influenced and deviate from zero. However, ϕ and β are controlled back to zero thanks to the IBS controller.

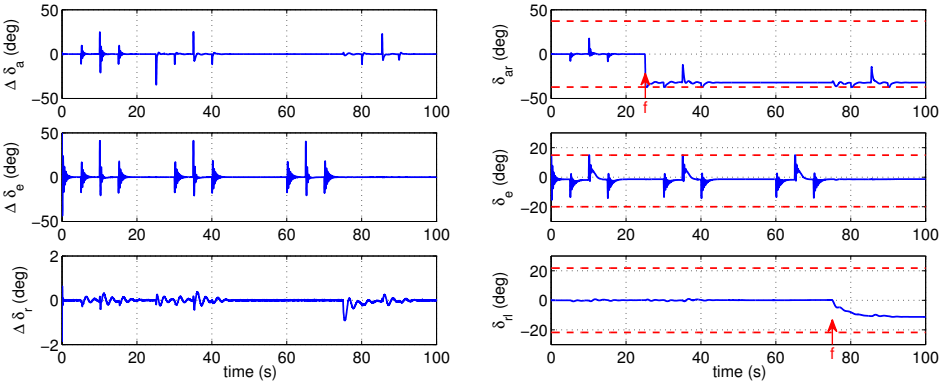
The state estimation performance remains satisfactory. Estimated and true ϕ, θ and β are shown in Figure 9.4(a) while those of V, α and ψ are shown in Figure 9.4(b). It can be seen that the state estimation is unbiased despite the sensor and actuator faults.

The incremental control surface deflections generated by the IBS controller are shown in Figure 9.4(c). The actual control surface deflections δ_{ar}, δ_e and δ_{rl} are shown in Figure 9.4(d). It is seen from the figure that the control surface deflections change after the occurrence of the faults. Take the ailerons for example, after the left aileron is stuck at a positive position at $t = 50$ s, the IBS controller takes effect. The desired command δ_a^{des} generated by the IBS decreases immediately. Then the right aileron deflects to a negative



(a) Estimated, true and commanded ϕ, θ and β in the presence of simultaneous sensor and actuator faults. The arrows denote the occurrence of the faults.

(b) Estimated and true V, α and ψ in the presence of simultaneous sensor and actuator faults



(c) Incremental actuator deflections in the presence of simultaneous sensor and actuator faults

(d) Actuator deflections in the presence of simultaneous sensor and actuator faults. Dashed lines denote position limits. Arrows denote the occurrence of the faults.

Figure 9.4: Results of the FTC system in the presence of simultaneous sensor and actuator faults

position. By doing this, the moments generated by the stuck left aileron is counteracted by the right aileron which deflects to a negative position.

It is also obvious that the remaining control authority will decrease due to the stuck aileron. Since the left aileron is stuck at a positive position, it generates a negative rolling moment all the time. Therefore, the control authority of rolling to the right is decreased. Therefore, extreme maneuvers should be performed with caution after the actuator faults.

The fault reconstruction using the sensor FDD system is still satisfactory even in the presence of simultaneous sensor and actuator faults. They are the same as those in Fig-

ures 9.3(a) and 9.3(b). Therefore, they are not shown again.

9.5.5. DISCUSSION

Through the validations in the previous sections, the following conclusions can be made:

1. The sensor FDD system is critical to maintain the performance in the presence of sensor faults.
   In Section 9.5.3, the reconfigurable controller (IBS) has no ability to recover the control in the presence of sensor faults if the sensor FDD is not included. This is expected since the controller relies on these measurements to calculate the error between the reference command and the controlled states. The control objective is to minimize the tracking errors such as those in Equation (9.46). Let z_θ denote the tracking error of θ and assume the wrong measurement is directly used by the controller, then

   $$z_\theta = \theta_m - \theta^{ref} = \theta + f_\theta + v_\theta - \theta^{ref} \tag{9.67}$$

   Assume z_θ is minimized to zero by the controller, it can be readily inferred that the true state θ deviates from the command θ^{ref} by $f_\theta + v_\theta$.

2. Simultaneous sensor and actuator FTC can be achieved by decoupling the sensor FTC from the actuator information. Through the validation in Section 9.5.4, simultaneous sensor and actuator FTC is achieved by the sensor FDD system and the reconfigurable controller. Since the sensor FDD system does not require the actuator information, it can perform state and fault estimation without being influenced by the actuator faults. The sensor FDD system provides unbiased state and fault estimation such that the reconfigurable controller can perform the reconfiguration without being influenced by sensor faults.

9.6. CONCLUSIONS

This chapter proposes an aircraft FTC system, which can maintain the controlled flight in the presence of simultaneous sensor and actuator faults. The proposed system contains a sensor FDD system and a reconfigurable controller. The sensor FDD system makes use of the aircraft kinematic model and does not require any information from the actuator. The ATS-UKF is used to reconstruct the state and fault in an unbiased sense. The state estimation is used by the controller to achieve sensor FTC. The actuator faults are dealt with by the reconfigurable controller which is the IBS in this chapter.

The proposed FTC system is validated by simulation examples. The simulated aircraft contains ADS and AHRS faults as well as stuck actuator faults. The simulation results demonstrate that the proposed FTC system is able to deal with the sensor faults as well as actuator faults and maintain the controlled flight.

The simulation results show that sensor and actuator FTC can be achieved simultaneously using model-based analytical redundancy. This FTC system has a potential to be applied to the aircraft to enhance the safety of the aircraft.

10

CONCLUSIONS AND RECOMMENDATIONS

This chapter summarises the main achievements of the research in Section 10.1, followed by a discussion in Section 10.2. In Section 10.3 the limitations of this dissertation, as well as recommendations for future work are presented.

10.1. CONCLUSIONS

This dissertation has been dedicated to increase aircraft safety through improving the robustness of FDD and FTC approaches. Both sensor and actuator faults, which have caused a number of airliner accidents, were considered. The following main research question was proposed at the start of this dissertation:

Research question

How to increase the robustness of the sensor and actuator FDD and FTC approaches with respect to real-life model uncertainties and internal and external disturbances?

To answer the main research question, three sub-questions or research goals were defined. The answers to the research sub-questions are presented in the following:

SENSOR FDD

The goal was to increase the robustness of sensor FDD with respect to model uncertainties and disturbances.

The considered sensors include IMU (accelerometers and rate gyros) and ADS (airspeed, AOA and angle of sideslip sensors). The fault types include bias and drift faults since these are common in sensors like the IMU. Furthermore, oscillatory faults [39] are also considered in this thesis. First, the airspeed-based kinematic model is taken as the

mathematical model that drives the FDD algorithm. The kinematic model uses the measured specific forces to compute the aerodynamic forces, thus reducing the influence of model uncertainties. To estimate the states and faults, which are important to realize fault-tolerance, advanced filtering techniques are proposed.

- Regarding IMU FDD, since the IMU faults are the input faults (faults in the input), an IOTSEKF, which is a modified version of the OTSEKF, was proposed for the state and fault estimation. By iteratively performing the measurement update in the first few time steps [73, 123], the IOTSEKF is less sensitive to initial condition errors than the OTSEKF [123]. Various types of faults, such as biases and drifts in the rate gyros, were implemented to validate the fault reconstruction performance using the IOTSEKF. Results demonstrate that all the bias and drift faults in the accelerometers and rate gyros are estimated in an unbiased sense [123].

- Regarding ADS FDD, since the ADS faults are the output faults (faults in the output), an ATS-UKF was proposed for the state and fault estimation. The RTS-UKF was applied and found to be sensitive to initial condition errors. It was analyzed that the sensitivity is caused by the fact that the estimates of the states measured by faulty measurements are not updated by the measurements [65]. Therefore, the ATS-UKF was proposed and overcomes the drawbacks of the RTS-UKF by performing an initial measurement update. [65]. Real flight data, in which bias and drift as well as oscillatory faults were injected, were used to test the performance of the ATS-UKF. All the faults in the ADSs can be estimated including the oscillatory faults which have faster dynamics.

- Furthermore, for fast fault detection speed, several MMAE approaches have been considered. There are several drawbacks related to existing MMAE approaches, such as their slow response to detect whether faults have been removed or not. To cope with these drawbacks, an SRMMAE approach was proposed. It significantly reduces the false alarms of existing MMAE approaches by developing three selective reinitialization algorithms [83]. The selective reinitialization algorithms perform reinitialization for different filters based on their model probabilities.

- Although the false alarms using the proposed SRMMAE are reduced, the computational load of MMAE and SRMMAE approaches is still intensive since they run a number of models online. To resolve the high computational load, a DMAE approach is developed which reduces the number of models to two even when dealing with simultaneous faults. Computational load is reduced and the detection speed is still satisfactory as compared to the conventional MMAE methods. The performance (false alarm rate, detection speed and computational load) of all proposed algorithms was validated using real flight test data, which contain unknown and realistic model uncertainties. A comparison among the MMAE, SRMMAE and DMAE is presented in Table 10.1.

Disturbances, especially external disturbances such as time-varying wind and atmospheric turbulence, are difficult to cope with within the field of sensor FDD. In this dissertation, the IMU FDD in the presence of external disturbances is solved in two steps.

	MMAE	SRMMAE	DMAE
False alarm rate			
Detection speed			
Computational load			

Table 10.1: Comparison among the MMAE, SRMMAE and DMAE. Green color indicates that the performance is best, yellow color denotes the second and orange color denotes the worst. Red color indicates that the performance is unacceptable.

	OTSEKF	IOTSEKF	ATSEKF
Sensitivity to initial condition errors			
Sensitivity to the choice of covariance matrices			

Table 10.2: Comparison among the OTSEKF, IOTSEKF and ATSEKF. Green color indicates that the performance is better and red color indicates that the performance is worse.

First, a different kinematic model, the ground speed-based kinematic model, was proposed as the mathematical model. Then, an ATSEKF was proposed, which improves the estimation performance of the IOTSEKF, for state and fault estimation. The estimation performance of the IOTSEKF degrades when the covariance matrices for fault estimate are not chosen wisely. The proposed ATSEKF can adaptively update the covariance matrices based on the innovation information [135] and can thus improve the performance of the IOTSEKF. A comparison among the OTSEKF, IOTSEKF and ATSEKF is presented in Table 10.2. To validate the proposed approach, more flight tests were performed in order to collect the aircraft response data recorded during the presence of atmospheric turbulence. IMU faults (bias, drift and oscillatory faults in the accelerometers and rate gyros) were injected into the recorded real flight data. Results show that the ATSEKF together with the GS-KM can indeed effectively estimate the sensor faults despite the presence of external disturbances such as turbulence. The aircraft can maintain the controlled flight since the estimated states are unbiased even in the presence of faults in the accelerometers and rate gyros.

Dealing with ADS FDD in the presence of turbulence is more difficult. One reason is that the airspeed-based kinematic model still has to be used. However, it cannot decouple the influence of the faults from the disturbances. To cope with that, the proposed DMAE approach is further extended to deal with the situation when the influence of disturbances and faults cannot be decoupled. It is theoretically proven that the faults and disturbances can be decoupled by using the selective-reinitialization algorithms in the DMAE. The FDD of the proposed ADS FDD is also validated using the recorded aircraft response data during the presence of atmospheric turbulence.

ACTUATOR FTC

The goal was to increase the robustness of the actuator FTC approaches in the presence of model uncertainties.

The considered FTC approaches, INDI and IBS, are model inversion-based approaches, which can handle nonlinearities. The control objective is to follow a reference trajectory in three-dimensional space which is usually required by air traffic control. Aircraft nonlinear trajectory control is challenging mainly due to the model uncertainties in the flight path and angular rate control loops. In this dissertation, the analysis of the uncertainties is presented. The flight path control and angular rate control loops are designed using the INDI which can deal with the model uncertainties. Furthermore, the actuator faults were also considered. The jammed actuator faults (one of the rudders and ailerons are both jammed) are implemented as an example. Simulation validation results demonstrate the aircraft can track the reference trajectory in the presence of model uncertainties and actuator faults by using the proposed trajectory controller. The advantages of the proposed approach are that it does not require a parameter update law design or online model identification.

SIMULTANEOUS SENSOR AND ACTUATOR FTC

The goal was to achieve simultaneous sensor and actuator FTC in the presence of model uncertainties.

To achieve simultaneous sensor and actuator FTC is challenging when using existing model-based FDD approaches. The reason is that existing sensor FDD approaches rely on the information of the actuators, which makes it difficult to cope with cases when the actuators themselves fail. This dissertation proposes a framework which can deal with simultaneous sensor and actuator FTC, through conducting sensor FDD without using any actuator information. The actuator faults are handled by reconfigurable control approaches. In this work, the IBS is applied as an example. ADS faults and actuator faults were implemented to validate the proposed fault-tolerant controller. Simulation results show that the proposed FTC system can maintain the controlled flight even in the presence of simultaneous sensor and actuator faults.

10.2. DISCUSSIONS

The contributions demonstrate that combining the aircraft kinematic model with advanced filtering techniques is a very promising approach to achieve sensor FTC. This approach can deal with real-life model uncertainties as well as external and internal disturbances. Different from existing approaches, which are usually only validated using simulated data and only deal with internal disturbances (such as noise), the approaches proposed in this dissertation are validated by real aircraft response data including the presence of turbulence. The proposed approaches brings analytical redundancy closer to real implementation. The validation in this dissertation suggests that real flight test data should be more widely used to validate sensor FTC, since this allows us to test the FTC performance in the presence of real-life, unknown, model uncertainties and disturbances.

The proposed aircraft trajectory controller significantly simplifies controller design compared to existing controllers, such as adaptive Backstepping and Nonlinear Dynamic Inversion with online model identification. One advantage is that it does not need additional effort to select the aerodynamic model structure. This controller structure can be

applied to other mechanical systems such as robots, automotive and spacecraft.

Although fixed-wing aircraft are studied in this dissertation, many proposed approaches can be readily applied to rotary-wing aircraft such as quadrotors, see [190, 191]. The proposed FTC techniques for quadrotors increase their safety and reliability levels and can thus speed up the integration process of, e.g., quadrotor UAVs into the civil and commercial aerospace.

Furthermore, the advanced filters proposed can cope with state estimation with unknown disturbances. Therefore, they can be applied to many other areas related to state estimation such as object tracking. In reality, unknown disturbances are always present and the proposed filters provide significant benefits to deal with real-life unknown disturbances.

10.3. RECOMMENDATIONS

Based on the results presented in this dissertation, the following future research directions are presented:

- ADS FDD in the presence of atmospheric turbulence is challenging since the effects of faults and turbulence are coupled. It was theoretically proven that the faults can be estimated in the presence of turbulence if the extended DMAE is applied. However, the only ADS fault considered in this proof is the bias fault. Future research will need to investigate the potential of the DMAE approach for other types of faults.

- Although the incremental approach can cope with model uncertainties as well as faults, its robustness with respect to external disturbances is not fully investigated. One future topic is to investigate the incremental control approaches for dealing with external disturbances. Furthermore, in the dissertation, only limited actuator fault scenarios (jamming faults) are considered. It would also be interesting to validate the performance of the incremental control approaches under various other fault scenarios such as “floating” actuator faults.

- Computational load is an issue when designing the FDD and FTC approaches due to the limited onboard computational power. Although this dissertation does take the computational load into account as a secondary objective (for instance, the DMAE was proposed to reduce the computational load of the MMAE approaches in Chapter 4), the computational load of other developed algorithms is still intensive. Several solutions can be investigated in this respect. First, one can replace the Unscented Kalman Filter with the Extended Kalman Filter or other filters with lower computational load. Secondly, algorithms with intensive computational load can be run at a lower frequency, at the cost of performance.

- The goal of fault detection approaches is to have low missed and false detection rates [8]. However, for most threshold-based detection approaches (detecting faults by monitoring the threshold), reducing missed detection rates means increasing false detection rates. How to achieve a “good” balance between missed and false detection rates is an interesting topic, which would make the approaches more practical.

- FTC is also important for micro aerial vehicles such as flapping-wing aircraft. Faults can bring detrimental effects to these vehicles and can even terminate its flight. Therefore, research on FTC for these vehicles is also highly recommended.
- In this thesis, it is stated that the kinematic models can reduce the effects of model uncertainties compared to the aerodynamic model. However, the FDD of using the aerodynamic model is not shown. Performance using both models should be compared in future studies. The comparison should be performed under scenarios in the presence of model uncertainties and internal and external disturbances.
- The real application of the proposed FDD and FTC approaches is important. The approaches proposed in this dissertation, which have a potential to be implemented on-board, were not validated in real flight tests. Future work would investigate the performance of the proposed approaches when they are implemented on aircraft.
- Finally, the FDD and FTC approaches proposed in this dissertation mainly aim for increasing the aircraft safety. Apparently, most of the techniques proposed can be applied to other systems to increase safety. For instance, the FDD approaches can be used to monitor various industrial processes such as chemical and power industries.

REFERENCES

[1] BEA, *Final Report on the Accident on 1st June 2009 to the Airbus A330-203 Registered F-GZCP Operated by Air France Flight AF 447 Rio de Janeiro Paris*, Tech. Rep. June 2009 (Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile, 2012).

[2] ATSB, *ATSB Transport Safety Report: In-Flight Upset 154 km West of Learmonth, WA 7 October 2008 VH-QPA Airbus A330-303*, Tech. Rep. October (Australian Transport Safety Bureau, 2008).

[3] F. L. Carpenter, *Summary of Facts- B2 Accident on 23 February 2008*, Tech. Rep. (Accident Investigation Board, USAF, 2008).

[4] M. H. Smaili, J. Breeman, T. Lombaerts, and D. A. Joosten, *A Simulation Benchmark for Integrated Fault Tolerant Flight Control Evaluation,* in *AIAA Modeling and Simulation Technologies Conference and Exhibit* (Keystone, Colorado, 2006) pp. 1–23.

[5] *Uncontrolled Descent and Collision with Terrain, United Airlines Flight 585, Boeing 737-200, N999UA, 4 Miles South of Colorado Springs Municipal Airport*, Tech. Rep. (National transportation safety board, 2001).

[6] *Uncontrolled Descent and Collision With Terrain USAir Flight 427 Boeing 737-300, N513AU Near Aliquippa, Pennsylvania September 8, 1994*, Tech. Rep. (National transportation safety board, 1999).

[7] C. Edwards, T. Lombaerts, and H. Smaili, *Fault Tolerant Flight Control* (Springer-Verlag Berlin Heidelberg, 2010).

[8] J. Chen, *Active Fault-Tolerant Flight Control Systems Design using the Linear Matrix Inequality Method,* **21**, 1 (1999).

[9] P. Goupil, *AIRBUS State of the Art and Practices on FDI and FTC in Flight Control System,* Control Engineering Practice **19**, 524 (2011).

[10] P. M. Frank, *Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge-based Redundancy- A Survey and Some New Results,* **26**, 459 (1990).

[11] R. J. Patton, *Fault-tolerant Control Systems: The 1997 Situation,* in *Proc. of IFAC Symp. on Fault Detection, Supervision and Safety for Technical Processes* (1997) pp. 1033–1054.

[12] R. Isermann, *Model-based Fault-Detection and Diagnosis- Status and Applications,* Annual Reviews in Control **29**, 71 (2005).

[13] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, *Model-based Fault Diagnosis for Aerospace Systems: a Survey,* Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering **226**, 1329 (2012).

[14] A. Zolghadri, *Advanced Model-based FDIR Techniques for Aerospace Systems: Today Challenges and Opportunities,* Progress in Aerospace Sciences **53**, 18 (2012).

[15] K. Watanabe and D. M. Himmelblau, *Instrument fault detection in systems with uncertainties,* International Journal of Systems Science **13**, 137 (1982).

[16] E. Kiyak, A. Kahvecioglu, and F. Caliskan, *Aircraft Sensor and Actuator Fault Detection, Isolation, and Accommodation,* Journal of Aerospace Engineering **24**, 46 (2011).

[17] D. Wang and K.-Y. Lum, *Adaptive unknown input observer approach for aircraft actuator fault detection and isolation,* International Journal of Adaptive Control and Signal Processing **21**, 31 (2007).

[18] V. I. Utkin, *Sliding Modes in Control and Optimization* (Springer-Verlag Berlin Heidelberg, 1992).

[19] C. Edwards, S. Spurgeon, and R. Patton, *Sliding Mode Observers for Fault Detection and Isolation,* Automatica **36**, 541 (2000).

[20] H. Alwi, C. Edwards, and A. Marcos, *Fault Reconstruction Using a LPV Sliding Mode Observer For a Class of LPV Systems,* Journal of the Franklin Institute **349**, 510 (2012).

[21] X. Ding and P. M. Frank, *Fault Detection via Factorization Approach,* Systems & Control Letters **14**, 431 (1990).

[22] R. S. Mangoubi, B. D. Appleby, G. C. Verghese, and W. E. Vandervelde, *A Robust Failure Detection and Isolation Algorithm,* in *Proceedings of the 34th Conference on Decision & Control* (New Orleans, LA, 1995) pp. 2377–2382.

[23] A. Marcos, *Assessment on the ADDSAFE Benchmark Simulator of an H-infinity Fault Detection Design for Aircraft,* in *8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (Mexico, 2012) pp. 1341–1346.

[24] M. Hou and R. Patton, *An LMI approach to H_/H infinity Fault Detection Observers,* in *Proceedings of the UKACC international conference on control*, 427 (1996) pp. 305–310.

[25] P. Freeman, P. Seiler, and G. J. Balas, *Air Data System Fault Modeling and Detection,* Control Engineering Practice **21**, 1290 (2013).

[26] R. Kalman, *A New Approach to Linear Filtering and Prediction Problems,* Journal of Fluids Engineering **82**, 35 (1960).

[27] S. J. Julier and J. K. Uhlmann, *A New Extension of the Kalman Filter to Nonlinear Systems,* in Proc. AeroSense: 11th Int. Symp. Aerospace/Defense Sensing, Simulation and Controls , 182 (1997).

[28] N. Gordan, D. Salmond, and A. Smith, *Novel Approach to Nonlinear/non-Gaussian Bayesian State Estimation,* in *Proc. Inst. Elect. Eng., F*, Vol. 140 (1993) pp. 107–113.

[29] R. Mehra and J. Peschon, *An Innovations Approach to Fault Detection and Diagnosis in Dynamic Systems,* Automatica **7**, 637 (1971).

[30] Q. Xia, M. Rao, Y. Ying, and X. Shen, *Adaptive Fading Kaiman Filter with an Application,* Automatica **30**, 1333 (1994).

[31] R. K. Mehra, *On the Identification of Variances and Adaptive Kalman Filtering,* IEEE Transactions on Automatic Control **15**, 175 (1970).

[32] B. Friedland, *Treatment of Bias in Recursive Filtering,* IEEE Transactions on Automatic Control **14**, 359 (1969).

[33] C.-S. Hsieh and F.-G. Chen, *Optimal Solution of the Two-Stage Kalman Estimator,* IEEE Transactions on Automatic Control **44**, 194 (1999).

[34] D. T. Magill, *Optimal Adaptive Estimation of Sampled Stochastic Processes,* IEEE Transactions on Automatic Control **10**, 434 (1965).

[35] H. A. P. Blom and Y. Bar-Shalom, *The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coeffcients,* IEEE Transactions on Automatic Control **33**, 780 (1988).

[36] P. S. Maybeck and K. P. Hentz, *Investigation of Moving-bank Multiple Model Adaptive Algorithms,* in *Proceedings of 24th Conference on Decison and Control* (Ft. Lauderdale, 1985) pp. 1874–1881.

[37] P. S. Maybeck, *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 **9**, 1051 (1999).

[38] X. R. Li and Y. Bar-Shalom, *Design of an Interacting Multiple Model Algorithm for Air Traffic Control Tracking,* IEEE Transactions on Control Systems Technology **1**, 186 (1993).

[39] P. Goupil, *Oscillatory Failure Case Detection in the A380 Electrical Flight Control System by Analytical Redundancy,* Control Engineering Practice **18**, 1110 (2010).

[40] Y. Zhang and J. Jiang, *Bibliographical Review on Reconfigurable Fault-tolerant Control Systems,* Annual reviews in control , 229 (2008).

[41] M. M. Kale and A. J. Chipperfield, *Stabilized MPC Formulations for Robust Reconfigurable Flight Control,* Control Engineering Practice **13**, 771 (2005).

[42] C. N. Jones, *Reconfigurable Flight Control First Year Report*, Tech. Rep. (University of Cambridge, 2005).

[43] F. A. D. Almeida and D. Leiß ling, *Fault-Tolerant Model Predictive Control with Flight-Test Results,* Journal of Guidance, Control, and Dynamics **33** (2010), 10.2514/1.46108.

[44] J. C. Doyle and G. Stein, *Multivariable Feedback Design: Concepts for a Classical/Modern Synthesis,* IEEE Transactions on Automatic Control **AC-26**, 4 (1981).

[45] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control* (John Wiley & Sons, 2001).

[46] G.-h. Yang, J. L. Wang, and Y. C. Soh, *Reliable H infinity controller design for linear systems,* Automatica **37**, 717 (2001).

[47] Y. Feng, F. Han, and X. Yu, *Chattering Free Full-Order Sliding-Mode Control,* Automatica **50**, 1310 (2014).

[48] Y. Shtessel, J. Buffington, and S. Banda, *Tailless Aircraft Flight Control Using Multiple Time Scale Reconfigurable Sliding Modes,* IEEE Transactions on Control Systems Technology **10**, 288 (2002).

[49] H. Alwi, C. Edwards, and C. P. Tan, *Fault Detection and Fault-Tolerant Control Using Sliding Modes* (Springer-Verlag, 2011).

[50] K. D. Young and V. I. Utkin, *A Control Engineer ' s Guide to Sliding Mode Control,* IEEE Transactions on Control Systems Technology **7**, 328 (1999).

[51] V. Utkin and H. Lee, *Chattering Problem in Sliding Mode Control Systems,* in *Proceedings of the 2006 International Workshop on Variable Structure Systems* (Alghero, Italy, 2006) pp. 346–350.

[52] A. Sabanovic, *Variable Structure Systems With Sliding Modes in Motion Control — A Survey,* IEEE Transactions on Automatic Control **7**, 212 (2011).

[53] C. Cao and N. Hovakimyan, *Design and Analysis of a Novel L 1 Adaptive Controller , Part I :,* in *Proceedings of the 2006 American Control Conference* (Minneapolis, Minnesota, USA, 2006) pp. 3397–3402.

[54] C. Cao and N. Hovakimyan, *Design and Analysis of a Novel L 1 Adaptive Controller , Part II :,* in *Proceedings of the 2006 American Control Conference* (Minneapolis, Minnesota, USA, 2006) pp. 3403–3408.

[55] C. Cao and N. Hovakimyan, *Guaranteed Transient Performance with L1 Adaptive Controller for Systems with Unknown Time-varying Parameters and Bounded Disturbances : Part I,* in *Proceedings of the 2007 American Control COnference* (New York, USA, 2007) pp. 3925–3930.

[56] E. Xargay, N. Hovakimyan, and C. Cao, *L1 Adaptive Controller for Multi-Input Multi-Output Systems in the Presence of Nonlinear Unmatched Uncertainties,* in *American Control Conference* (Marriott Waterfront, Baltimore, MD, USA, 2010) pp. 1–6.

[57] V. Dobrokhodov, I. Kaminer, I. Kitsios, E. Xargay, N. Hovakimyan, C. Cao, I. M. Gregory, and L. Valavani, *Experimental Validation of L1 Adaptive Control: The Rohrs Counterexample in Flight,* Journal of Guidance, Control, and Dynamics **34**, 1311 (2011).

[58] H. K. Khalil, *Nonlinear Systems, 3rd edition* (Prentice Hall, 2002).

[59] T. Lombaerts, H. Huisman, Q. P. Chu, J. A. Mulder, and D. Joosten, *Nonlinear Reconfiguring Flight Control Based on Online Physical Model Identification,* Journal of Guidance, Control, and Dynamics **32**, 727 (2009).

[60] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design* (John Wiley & Sons, Inc., 1995).

[61] J. A. Farrell, M. Polycarpou, M. Sharma, and W. Dong, *Command Filtering Backstepping,* IEEE Transactions on Automatic Control **54**, 1391 (2009).

[62] W. Dong, J. A. Farrell, M. M. Polycarpou, V. Djapic, and M. Sharma, *Command Filtered Adaptive Backstepping,* IEEE Transactions on Control Systems Technology **20**, 566 (2012).

[63] L. Sonneveldt, Q. P. Chu, and J. A. Mulder, *Nonlinear Flight Control Design Using Constrained Adaptive Backstepping,* Journal of Guidance, Control, and Dynamics **30** (2007), 10.2514/1.25834.

[64] D. Mclean, *Automatic Flight Control Systems* (Englewood Cliffs, NJ: Prentice-Hall, 1990).

[65] P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, and Q. P. Chu, *Adaptive Three-Step Kalman Filter for Air Data Sensor Fault Detection and Diagnosis,* Journal of Guidance, Control, and Dynamics , 1 (2015).

[66] J. Chen and R. J. Patton, *Robust Model-based Fault Diagnosis for Dynamic Systems* (Kluwer Academic Publishers, Norwell, MA, USA, 1999).

[67] V. Venkatasubramanian, R. Rengaswamy, and K. Yin, *A Review of Process Fault Detection and Diagnosis Part I : Quantitative Model-based Methods,* Computers and Chemical Engineering **27**, 293 (2003).

[68] C. Hajiyev and F. Caliskan, *Fault Diagnosis and Reconfiguration in Flight Control Systems* (Kluwer Academic Publishers, Boston, 2003).

[69] C. Hajiyev and H. E. Soken, *Robust Adaptive Kalman Filter for Estimation of UAV Dynamics in the Presence of Sensor/Actuator Faults,* Aerospace Science and Technology **28**, 376 (2013).

[70] A. Zolghadri, D. Henry, J. Cieslak, D. Efimov, and P. Goupil, *Fault Diagnosis and Fault-Tolerant Control and Guidance for Aerospace Vehicles, From Theory to Application* (Springer, Series: Advances in Industrial Control, 2013).

[71] D. Houck and L. Atlas, *Air Data Sensor Failure Detection,* in *17th Digital Avionics Systems Conference*, May 1979 (1998) pp. 1–8.

[72] R. D. Eubank, E. M. Atkins, and S. Ogura, *Fault Detection and Fail-Safe Operation with a Multiple-Redundancy Air-Data System,* in *AIAA Guidance, Navigation and Control Conference*, August 2010 (Toronto, Ontario Canada, 2010) pp. 1–14.

[73] T. Lombaerts, *Fault Tolerant Flight Control- A Physical Model Approach*, Ph.D. thesis, Delft University of Technology (2010).

[74] J. C. Ellsworth and S. A. Whitmore, *Reentry Air Data System for a Sub-orbital Spacecraft Based on X-34 Design,* in *AIAA Aerospace Sciences Meeting and Exhibit*, January (Reno, Nevada, 2007) pp. 1–31.

[75] T. J. Rohloff, S. A. Whitmore, and I. Catton, *Fault-Tolerant Neural Network Algorithm for Flush Air Data Sensing,* Journal of Aircraft **36**, 541 (1999).

[76] F. Nebula, R. Palumbo, G. Morani, and F. Corraro, *Virtual Air Data System Architecture for Space Reentry Applications,* Journal of Spacecraft and Rockets **46**, 818 (2009).

[77] F. Nebula, R. Palumbo, and G. Morani, *Virtual Air Data : a Fault-Tolerant Approach Against ADS Failures,* in *AIAA Guidance, Navigation and Control and Co-located Conferences* (Boston, MA, 2013) pp. 1–14.

[78] G. Looye and H.-D. Joos, *Design of Robust Dynamic Inversion Control Laws using Multi-Objective Optimization,* in *AIAA Guidance, Navigation and Control Conferences and Exhibit*, August (Montreal, Canada, 2001).

[79] F. Cervia, E. Denti, R. Galatolo, and F. Schettini, *Air Data Computation in Fly-by-wire Flight-control Systems,* Journal of aircraft **43**, 450 (2006).

[80] L. Van Eykeren and Q. P. Chu, *Fault Detection and Isolation for Inertial Reference Units,* in *AIAA Guidance, Navigation and Control Conference* (Boston, MA, 2013) pp. 1–10.

[81] P. Lu, L. Van Eykeren, E. van Kampen, Q. P. Chu, and B. Yu, *Adaptive Hybrid Unscented Kalman Filter for Aircraft Sensor Fault Detection, Isolation and Reconstruction,* in *AIAA Guidance, Navigation, and Control Conference*, AIAA 2014-1145 (National Harbor, Maryland, 2014) pp. 1–18.

[82] L. Van Eykeren, Q. P. Chu, and J. A. Mulder, *Sensor Fault Detection and Isolation using Adaptive Extended Kalman Filter,* in *8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 1969 (Mexico, 2012) pp. 1155–1160.

[83] P. Lu, L. Van Eykeren, E. van Kampen, and Q. P. Chu, *Selective-Reinitialisation Multiple Model Adaptive Estimation for Fault Detection and Diagnosis,* Journal of Guidance, Control, and Dynamics **38**, 1409 (2015).

[84] S. Gillijns and B. De Moor, *Unbiased Minimum-Variance Input and State Estimation for Linear Discrete-Time Systems with Direct Feedthrough,* Automatica **43**, 111 (2007).

[85] P. Lu and E. van Kampen, *Aircraft Inertial Measurement Unit Fault Identification with Application to Real Flight Data,* in *AIAA Guidance, Navigation and Control Conference*, AIAA 2015-0859 (Kissimmee, Florida, 2015) pp. 1–17.

[86] R. Van Der Merwe and E. A. Wan, *The Square-root Unscented Kalman Filter for State and Parameter-estimation,* in *IEEE International Conference on Acoustics, Speech, and Signal Processing* (2001) pp. 3461–3464.

[87] B. N. Alsuwaidan, J. L. Crassidis, and Y. Cheng, *Convergence Properties of Autocorrelation-Based Generalized Multiple-Model Adaptive Estimation,* in *AIAA Guidance, Navigation and Control Conference and Exhibit*, August (Honolulu, Hawaii).

[88] R. K. Mehra, *Approaches to Adaptive Filtering,* IEEE Transactions on Automatic Control **17**, 693 (1972).

[89] H. E. Soken and C. Hajiyev, *Pico Satellite Attitude Estimation via Robust Unscented Kalman Filter in the Presence of Measurement Faults.* ISA transactions **49**, 249 (2010).

[90] J. A. Mulder, Q. P. Chu, J. K. Sridhar, J. H. Breeman, and M. Laban, *Non-linear Aircraft Flight Path Reconstruction Review and New Advances,* Progress in Aerospace Sciences **35**, 673 (1999).

[91] C. C. de Visser, *Global Nonlinear Model Identification with Multivariate Splines*, Ph.D. thesis, Delft University of Technology (2011).

[92] P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, and Q. P. Chu, *Double-Model Adaptive Fault Detection and Diagnosis Applied to Real Flight Data,* Control Engineering Practice **36**, 39 (2015).

[93] P. Eide and P. S. Maybeck, *An MMAE Failure Detection System for the F-16,* IEEE Transactions on Aerospace and Electronic Systems **32**, 1125 (1996).

[94] R. Isermann, *Fault-Diagnosis Applications* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011) pp. 11–45.

[95] A. Marcos, S. Ganguli, and G. J. Balas, *An Application of Fault Detection and Isolation to A Transport Aircraft,* Control Engineering Practice **13**, 105 (2005).

[96] D. Efimov, A. Zolghadri, and T. Raïssi, *Actuator Fault Detection and Compensation Under Feedback Control,* Automatica **47**, 1699 (2011).

[97] D. Berdjag, J. Cieslak, and A. Zolghadri, *Fault Diagnosis and Monitoring of Oscillatory Failure Case in Aircraft Inertial System,* Control Engineering Practice **20**, 1410 (2012).

[98] H. Alwi and C. Edwards, *An Adaptive Sliding Mode Differentiator for Actuator Oscillatory Failure Case Reconstruction,* Automatica **49**, 642 (2013).

[99] Y. Zhang and X. R. Li, *Detection and Diagnosis of Sensor and Actuator Failures Using IMM Estimator,* IEEE Transactions on Aerospace and Electronic Systems **34**, 1293 (1998).

[100] P. D. Hanlon and P. S. Maybeck, *Multiple-Model Adaptive Estimation Using a Residual Correlation Kalman Filter Bank,* IEEE Transactions on Aerospace and Electronic Systems **36**, 393 (2000).

[101] C. D. Ormsby, J. F. Raquet, and P. S. Maybeck, *A New Generalized Residual Multiple Model Adaptive Estimator of Parameters and States,* Mathematical and Computer Modelling **43**, 1092 (2006).

[102] P. S. Maybeck and M. R. Schore, *Robustness of a Moving-bank Multiple Model Adaptive Algorithm for Control of a Flexible Spacestructure,* in *In Proceedings of the National Aerospace and Electronics Conference* (NAECON 90, Dayton, 1990) pp. 368–374.

[103] G. Ducard and H. P. Geering, *Efficient Nonlinear Actuator Fault Detection and Isolation System for Unmanned Aerial Vehicles,* Journal of Guidance, Control, and Dynamics **31**, 225 (2008).

[104] J. F. Ru and X. R. Li, *Variable-Structure Multiple-Model Approach to Fault Detection, Identification, and Estimation,* IEEE Transactions on Control Systems Technology **16**, 1029 (2008).

[105] X. R. Li and Y. Bar-shalom, *Multiple-Model Estimation with Variable Structure,* IEEE Transactions on Automatic Control **41**, 478 (1996).

[106] J. R. Vasquez and P. S. Maybeck, *Enhanced Motion and Sizing of Bank in Moving-Bank MMAE,* IEEE Transactions on Aerospace and Electronic Systems **40**, 770 (2004).

[107] P. Lu and E. van Kampen, *Selective-Reinitialisation Multiple Model Adaptive Estimation for Fault Detection and Diagnosis,* in *AIAA Guidance, Navigation, and Control Conference*, AIAA 2014-0965 (National Harbor, Maryland, 2014) pp. 1–24.

[108] J. Dunik and M. Simandl, *Estimation of State and Measurement Noise Covariance Matrices by Multi-Step Prediction,* in *Proceedings of the 17th World Congress , the International Federation of Automatic Control* (Seoul, Korea, 2008) pp. 3689–3694.

[109] S. J. Julier and J. K. Uhlmann, *Unscented Filtering and Nonlinear Estimation,* in *Proceedings of the IEEE*, Vol. 92 (2004) pp. 401–422.

[110] S. Särkkä, *On Unscented Kalman Filtering for State Estimation of Continuous-Time Nonlinear Systems,* IEEE Transactions on Automatic Control **52**, 1631 (2007).

[111] J. F. Ru and X. R. Li, *Interacting Multiple Model Algorithm with Maximum Likelihood Estimation for FDI,* in *Proceedings of the 2003 IEEE International Symposium on Intelligent Control ISIC-03*, 2 (Ieee, 2003) pp. 661–666.

[112] P. Goupil, J. Boada-bauxell, A. Marcos, E. Cortet, M. Kerr, and H. Costa, *AIRBUS Efforts Towards Advanced Real-Time Fault Diagnosis and Fault Tolerant Control,* in *The International Federation of Automatic Control World Congress* (Cape Town, South Africa, 2014) pp. 3471–3476.

[113] A. Varga and D. Ossmann, *LPV Model-based Robust Diagnosis of Flight Actuator Faults,* Control Engineering Practice **31**, 135 (2013).

[114] L. Van Eykeren and Q. P. Chu, *Sensor Fault Detection and Isolation for Aircraft Control Systems by Kinematic Relations,* Control Engineering Practice **31**, 200 (2014).

[115] D. Henry, J. Cieslak, A. Zolghadri, and D. Efimov, *A Non-conservative H_/H infinity Solution for Early and Robust Fault Diagnosis in Aircraft Control Surface Servoloops,* Control Engineering Practice **31**, 183 (2014).

[116] H. Alwi and C. Edwards, *Development and Application of Sliding Mode LPV Fault Reconstruction Schemes for the ADDSAFE Benchmark,* Control Engineering Practice **31**, 148 (2014).

[117] L. Chen, R. J. Patton, and P. Goupil, *Robust Fault Estimation and Performance Evaluation based upon the ADDSAFE Benchmark Model,* in *8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (Mexico, 2012) pp. 1364–1369.

[118] B. Vanek, A. Edelmayer, Z. Szabó, and J. Bokor, *Bridging the Gap between Theory and Practice in LPV Fault Detection for Flight Control Actuators,* Control Engineering Practice **31**, 171 (2014).

[119] S. Hecker and H. Pfifer, *Affine LPV-modeling for the ADDSAFE benchmark,* Control Engineering Practice **31**, 126 (2014).

[120] R. Hallouzi, M. Verhaegen, and S. Kanev, *Multiple Model Estimation : A Convex Model Formulation,* International Journal of Adaptive Control and Signal Processing **23**, 217 (2009).

[121] X. R. Li and V. P. Jilkov, *Expected-Mode Augmentation for Multiple-Model Estimation,* in *International Conference on information fusion* (2001).

[122] K. A. Fisher and P. S. Maybeck, *Multiple Model Adaptive Estimation with Filter Spawning,* IEEE Transactions on Aerospace and Electronic Systems **38**, 755 (2002).

[123] P. Lu, L. Van Eykeren, E. van Kampen, C. C. de Visser, and Q. P. Chu, *Aircraft Inertial Measurement Unit Fault Detection and Diagnosis with Application to Real Flight Data,* Journal of Guidance, Control, and Dynamics **38**, 2467 (2015).

[124] P. Goupil and A. Marcos, *The European ADDSAFE Project: Industrial and Academic Efforts Towards Advanced Fault Diagnosis,* Control Engineering Practice **31**, 109 (2014).

[125] H. Alwi and C. Edwards, *Robust Fault Reconstruction for Linear Parameter Varying Systems Using Sliding Mode Observers,* International Journal of Robust and Nonlinear Control **24**, 1947 (2013).

[126] L. Van Eykeren and Q. P. Chu, *Air Data Sensor Fault Detection using Kinematic Relations,* in *Proceedings of the EuroGNC 2013, 2nd CEAS Special Conference on Guidance, Navigation & Control* (2013) pp. 414–428.

[127] A. Levant, *Robust Exact Differentiation via Sliding Mode Technique,* Automatica **34**, 379 (1998).

[128] N. Eva Wu, Y. Zhang, and K. Zhou, *Detection, Estimation, and Accommodation of Loss of Control Effectiveness,* International Journal of Adaptive Control and Signal Processing **14**, 775 (2000).

[129] C. Hajiyev, *Two-stage Kalman Filter-based Actuator / Surface Fault Identification and Reconfigurable Control Applied to F-16 Fighter Dynamics,* International Journal of Adaptive Control and Signal Processing **27**, 755 (2012).

[130] L. Forssell and U. Nilsson, *ADMIRE The Aero-Data Model In a Research Environment*, Tech. Rep. December (2005).

[131] B. Hu and P. Seiler, *Certification Analysis for a Model-Based UAV Fault Detection System,* in *AIAA Guidance, Navigation and Control Conference*, AIAA 2014-0610 (National Harbor, Maryland, 2014) pp. 1–14.

[132] P. Freeman, R. Pandita, N. Srivastava, and G. J. Balas, *Model-Based and Data-Driven Fault Detection Performance for a Small UAV,* IEEE/ASME Transactions on Mechatronics **18**, 1300 (2013).

[133] F. Moschas and S. Stiros, *PLL Bandwidth and Noise in 100 Hz GPS Measurements,* GPS Solutions **19**, 173 (2014).

[134] J. Keller and M. Darouach, *Two-stage Kalman estimator with unknown exogenous inputs,* Automatica **35**, 339 (1999).

[135] P. Lu, E. van Kampen, C. de Visser, and Q. P. Chu, *Nonlinear Aircraft Sensor Fault Reconstruction in the Presence of Disturbances Validated by Real Flight Data,* Control Engineering Practice **49**, 112 (2016).

[136] P. Goupil, J. Boada-bauxell, A. Marcos, P. Rosa, M. Kerr, and L. Dalbies, *An Overview of the FP7 RECONFIGURE Project: Industrial, Scientific and Technological Objectives,* 9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes **48**, 976 (2015).

[137] P. Castaldi, W. Geri, M. Bonfe, S. Simani, and M. Benini, *Design of Residual Generators and Adaptive Filters for the FDI of Aircraft Model Sensors,* Control Engineering Practice **18**, 449 (2010).

[138] P. Castaldi, N. Mimmo, and S. Simani, *Differential Geometry Based Active Fault Tolerant Control for Aircraft,* Control Engineering Practice **32**, 227 (2014).

[139] R. C. Avram, X. Zhang, J. Campbell, and J. Muse, *IMU Sensor Fault Diagnosis and Estimation for Quadrotor UAVs,* in *9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, Vol. 48 (Elsevier Ltd., 2015) pp. 380–385.

[140] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, *Control-based Fault Detection and Isolation for Autonomous Aircraft,* Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering **226**, 510 (2011).

[141] C. C. de Visser, J. A. Mulder, and Q. P. Chu, *Validating the Multidimensional Spline Based Global Aerodynamic Model for the Cessna Citation II,* in *AIAA Atmospheric Flight Mechanics Conference*, AIAA 2011-6356 (Toronto, Ontario Canada, 2010) pp. 1–16.

[142] D. Bates and M. Hagström, *Nonlinear Analysis and Synthesis Techniques for Aircraft Control* (Springer-Verlag Berlin Heidelberg, 2007).

[143] R. Sreedhar, B. Fernandez, and G. Y. Masada, *Robust Fault Detection in Nonlinear Systems Using Sliding Mode Observers,* in *Second IEEE Conference on Control Applications*, 2 (Vancouver, BC, 1993) pp. 715–721.

[144] P. Lu, E. van Kampen, C. de Visser, and Q. P. Chu, *Novel Framework to State and Unknown Input Estimation of Linear Time-varying Systems,* Automatica (2016), 10.1016/j.automatica.2016.07.009.

[145] C.-S. Hsieh and F.-G. Chen, *Optimal solution of the two-stage Kalman filter,* IEEE Transactions on Automatic Control **44**, 194 (1999).

[146] C.-S. Hsieh, *Extension of unbiased minimum-variance input and state estimation for systems with unknown inputs,* Automatica **45**, 2149 (2009).

[147] P. K. Kitanidis, *Unbiased Minimum-variance Linear State Estimation,* Automatica **23**, 775 (1987).

[148] M. Darouach and M. Zasadzinski, *Unbiased Minimum Variance Estimation for Systems with Unknown Exogenous Inputs,* Automatica **33**, 717 (1997).

[149] M. Hou and R. J. Patton, *Optimal Filtering for Systems with Unknown Inputs,* IEEE Transactions on Automatic Control **43**, 445 (1998).

[150] C.-S. Hsieh, *Robust Two-Stage Kalman Filters for Systems with Unknown Inputs,* IEEE Transactions on Automatic Control **45**, 2374 (2000).

[151] J. Chen and R. J. Patton, *Optimal Filtering and Robust Fault Diagnosis of Stochastic Systems with Unknown Disturbances,* IEEE Proceedings Control Theory and Applications **143**, 31 (1996).

[152] M. Darouach, M. Zasadzinski, and M. Boutayeb, *Extension of Minimum Variance Estimation for Systems with Unknown Inputs,* Automatica **39**, 867 (2003).

[153] S. Gillijns and B. De Moor, *Unbiased Minimum-Variance Input and State Estimation for Linear Discrete-Time Systems,* Automatica **43**, 111 (2007).

[154] Y. Cheng, H. Ye, Y. Wang, and D. Zhou, *Unbiased Minimum-Variance State Estimation for Linear Systems with Unknown Input,* Automatica **45**, 485 (2009).

[155] F. Ben Hmida, K. Khémiri, J. Ragot, and M. Gossa, *Unbiased Minimum-Variance Filter for State and Fault Estimation of Linear Time-Varying Systems with Unknown Disturbances,* Mathematical Problems in Engineering **2010**, 1 (2010).

[156] C.-S. Hsieh, *On the Global Optimality of Unbiased Minimum-variance State Estimation for Systems with Unknown Inputs,* Automatica **46**, 708 (2010).

[157] A. Doucet, S. Godsill, and C. Andrieu, *On Sequential Monte Carlo Sampling Methods for Bayesian Filtering,* Statistics and Computing **10**, 197 (2000).

[158] V. Verma, G. Gordon, R. Simmons, and S. Thrun, *Real-Time Fault Diagnosis,* IEEE Robotics & Automation Magzine **11**, 56 (2004).

[159] F. Caron, M. Davy, E. Duflos, and P. Vanheeghe, *Particle Filtering for Multisensor Data Fusion With Switching Observation Models: Application to Land Vehicle Positioning,* IEEE Transactions on Signal Processing **55**, 2703 (2007).

[160] N. D. Freitas, *Rao-Blackwellised Particle Filtering for Fault Diagnosis,* in *Proceedings IEEE Aerospace Conference* (2002) pp. 4–1767–4–1772.

[161] B. Zhao, R. Skjetne, M. Blanke, and F. Dukan, *Particle Filter for Fault Diagnosis and Robust Navigation of Underwater Robot,* IEEE Transactions on Control Systems Technology **22**, 2399 (2014).

[162] S. H. Park, P. S. Kim, O.-k. Kwon, and W. H. Kwon, *Estimation and Detection of Unknown Inputs Using Optimal FIR Filter,* Automatica **36**, 1481 (2000).

[163] A. H. Jazwinski, *Adaptive Filtering,* Automatica **5**, 475 (1969).

[164] M. Darouach, M. Zasadzinski, O. Bassong, and S. Nowakowski, *Kalman Filtering with Unknown Inputs via Optimal State Estimation of Singular Systems,* International Journal of Systems Science **26**, 2015 (1995).

[165] I. Kaminer, A. Pascoal, E. Hallberg, and C. Silvestre, *Trajectory Tracking for Autonomous Vehicles : An Integrated Approach to Guidance and Control,* Journal of Guidance, Control, and Dynamics **21**, 29 (1998).

[166] H. Alwi, C. Edwards, O. Stroosma, and J. A. Mulder, *Evaluation of a Sliding Mode Fault-Tolerant Controller for the El Al Incident,* Journal of Guidance, Control, and Dynamics **33**, 677 (2010).

[167] S. N. Singh, M. L. Steinberg, and A. B. Page, *Nonlinear Adaptive and Sliding Mode Flight Path Control of F / A-18 Model,* IEEE Transactions on Aerospace and Electronic Systems **39**, 1250 (2003).

[168] W. Ren and R. W. Beard, *Trajectory Tracking for Unmanned Air Vehicles With Velocity and Heading Rate Constraints,* IEEE Transactions on Control Systems Technology **12**, 706 (2004).

[169] J. Farrell, M. Sharma, and M. Polycarpou, *Backstepping-Based Flight Control with Adaptive Function Approximation,* Journal of Guidance, Control, and Dynamics **28**, 1089 (2005).

[170] L. Sonneveldt, E. R. Van Oort, Q. P. Chu, and J. A. Mulder, *Nonlinear Adaptive Trajectory Control Applied to an F-16 Model,* Journal of Guidance, Control, and Dynamics **32**, 25 (2009).

[171] T. J. J. Lombaerts, Q. P. Chu, J. A. Mulder, and D. A. Joosten, *Modular flight control reconfiguration design and simulation,* Control Engineering Practice **19**, 540 (2011).

[172] M. Maximilian, J. C. Dauer, and F. Holzapfel, *Adaptive Trajectory Controller for Generic Fixed-Wing Unmanned Aircraft,* in *Proceedings of the EuroGNC 2013, 2nd CEAS Special Conference on Guidance, Navigation & Control* (2013) pp. 236–255.

[173] M. Krstić, I. Kanellakopoulos, and P. Kokotović, *Adaptive Nonlinear Control without Overparametrization,* Systems & Control Letters **19**, 177 (1992).

[174] H. J. Tol, C. C. D. Visser, L. G. Sun, E. V. Kampen, and Q. P. Chu, *Multivariate Spline-Based Adaptive Control of High-Performance Aircraft with Aerodynamic Uncertainties,* Journal of Guidance, Control and Dynamics , 1 (2016).

[175] T. J. J. Lombaerts, E. R. Van Oort, Q. P. Chu, J. A. Mulder, and D. A. Joosten, *Online Aerodynamic Model Structure Selection and Parameter Estimation for Fault-Tolerant Control,* Journal of Guidance, Control and Dynamics **33**, 707 (2010).

[176] B. J. Bacon, A. J. Ostroff, and S. M. Joshi, *Reconfigurable NDI Controller Using Inertial Sensor Failure Detection & Isolation,* IEEE Transactions on Aerospace and Electronic Systems **37**, 1373 (2001).

[177] S. Sieberling, Q. P. Chu, and J. A. Mulder, *Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction,* Journal of Guidance, Control, and Dynamics **33**, 1732 (2010).

[178] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation* (Wiley, New York, 1992).

[179] H. K. Khalil, *Nonlinear Systems Third Edition* (Prentice Hall, 2002).

[180] P. Lu, E. van Kampen, and Q. P. Chu, *Robustness and Tuning of Incremental Backstepping,* in *AIAA Guidance, Navigation and Control Conference*, AIAA 2015-1762 (Kissimmee, Florida, 2015) pp. 1–15.

[181] E. J. J. Smeur, Q. P. Chu, and G. C. H. E. D. Croon, *Adaptive Incremental Nonlinear Dynamic Inversion for Attitude Control of Micro Air Vehicles,* Journal of Guidance, Control, and Dynamics , 1 (2016).

[182] C. van der Linden, *DASMAT-Delft University Aircraft Simulation Model and Analysis Tool* (Delft University Press, 1998).

[183] R. Venkataraman, M. Lukatsi, B. Vanek, and P. Seiler, *Reliability Assessment Reliability Assessment of Actuator Architectures for Unmanned Aircraft,* in *9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* (2015) pp. 398–403.

[184] P. Lu and E. van Kampen, *Active Fault-Tolerant Control System using Incremental Backstepping Approach,* in *AIAA Guidance, Navigation and Control Conference*, AIAA 2015-1312 (Kissimmee, Florida, 2015) pp. 1–17.

[185] ICAO, *Safety Report*, Tech. Rep. (International Civil Aviation Organization, 2015).

[186] H. Alwi, L. Chen, and C. Edwards, *Reconstruction of Simultaneous Actuator and Sensor Faults for the RECONFIGURE Benchmark Using a Sliding Mode Observer,* in *The International Federation of Automatic Control World Congress*, 2 (Cape Town, South Africa, 2014) pp. 3497–3502.

[187] J. Cieslak, D. Henry, A. Zolghadri, and P. Goupil, *Development of an Active Fault-Tolerant Flight Control Strategy,* Journal of Guidance, Control, and Dynamics **31**, 135 (2008).

[188] A. A. H. Ali, *Adaptive Incremental Backstepping using Immersion and Invariance*, Master thesis, Delft University of Technology (2013).

[189] P. Acquatella, E.-J. van Kampen, and Q. P. Chu, *Incremental Backstepping for Robust Nonlinear Flight Control,* in *Proceedings of the EuroGNC 2013, 2nd CEAS Special Conference on Guidance, Navigation & Control* (2013) pp. 1444–1463.

[190] P. Lu and E. van Kampen, *Active Fault-Tolerant Control for Quadrotors Subjected to a Complete Rotor Failure,* in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2015)* (Hamburg, Germany, 2015) pp. 4698–4703.

[191] P. Lu, E. van Kampen, and Q. P. Chu, *Nonlinear Quadrotor Control with Online Model Identification,* in *Advances in Aerospace Guidance, Navigation and Control* (Springer, 2015) pp. 81–98.

A

Reference Frame Definitions

In this appendix, some of the reference frames used in this thesis are introduced. The Earth-fixed reference frame F_E and body-fixed reference frame F_B are given in Figure A.1. The origin of F_B is fixed to the aircraft.

In the figure, p, q and r represent the rotational rates of the body-fixed reference frame F_B with respect to the Earth-fixed reference frame F_E. The aircraft photo with more reference frames can be found in Figure 8.1.

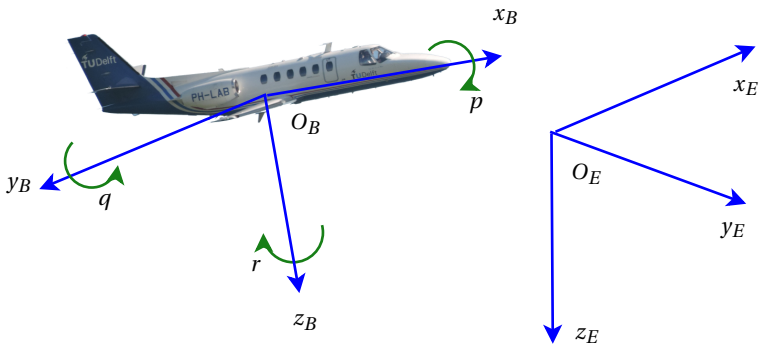


Figure A.1: The Earth-fixed reference frame F_E and body-fixed reference frame F_B. Picture by courtesy of J. A. Pascoe.

A

The transformation matrix from body-fixed reference frame to Earth-fixed reference frame $\mathbf{T}_{eb}$, which is used in Chapter 5, is as follows:

$$\mathbf{T}_{eb} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}$$

where ϕ, θ and ψ are the roll, pitch and yaw angles of the aircraft.

The transformation matrix from Earth-fixed reference frame to body-fixed reference frame $\mathbf{T}_{be} = \mathbf{T}_{eb}^T$.

B

AIRCRAFT AIRSPEED-BASED KINEMATIC MODEL

B.1. AIRSPEED-BASED KINEMATIC MODEL WITHOUT CONSIDERING THE EFFECTS OF FAULTS

In this section, the airspeed-based kinematic model without incorporating the faults are presented. It should be noted that in this appendix, the effects of time-varying wind are not included.

The process model of an airspeed-based kinematic model, which uses u_a, v_a, w_a, ϕ, θ and ψ as the states, is given as follows [90]:

$$\dot{u}_a = v_a r - w_a q - g\sin\theta + A_x \tag{B.1}$$
$$\dot{v}_a = -u_a r + w_a p + g\cos\theta\sin\phi + A_y \tag{B.2}$$
$$\dot{w}_a = u_a q - v_a p + g\cos\theta\cos\phi + A_z \tag{B.3}$$
$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{B.4}$$
$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{B.5}$$
$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{B.6}$$

where u_a, v_a, w_a are the air velocity components defined in the body-fixed reference frame. A_x, A_y, A_z are the specific forces (the difference between the inertial acceleration and the gravitational acceleration) along the body axis, ϕ, θ and ψ are the Euler angles and p, q, r are the rotational rates of the aircraft.

Substitute Equations (B.1)-(B.3) into the following equations:

$$\dot{V} = \dot{u}_a \cos\alpha\cos\beta + \dot{v}_a \sin\beta + \dot{w}_a \sin\alpha\cos\beta \tag{B.7}$$

$$\dot{\alpha} = \frac{\dot{w}_a \cos\alpha - \dot{u}_a \sin\alpha}{V\cos\beta} \tag{B.8}$$

$$\dot{\beta} = \frac{1}{V}(-\dot{u}_a \cos\alpha\sin\beta + \dot{v}_a \cos\beta - \dot{w}_a \sin\alpha\sin\beta) \tag{B.9}$$

One then obtains the following airspeed-based kinematic model which uses V, α, β, ϕ, θ and ψ as the states [126]:

$$\begin{aligned}\dot{V} = {} & (A_x - g\sin\theta)\cos\alpha\cos\beta + (A_y + g\sin\phi\cos\theta)\sin\beta \\ & + (A_z + g\cos\phi\cos\theta)\sin\alpha\cos\beta\end{aligned} \tag{B.10}$$

$$\begin{aligned}\dot{\alpha} = {} & \frac{1}{V\cos\beta}(-A_x\sin\alpha + A_z\cos\alpha + g\cos\phi\cos\theta\cos\alpha \\ & + g\sin\theta\sin\alpha) + q - (p\cos\alpha + r\sin\alpha)\tan\beta\end{aligned} \tag{B.11}$$

$$\begin{aligned}\dot{\beta} = {} & \frac{1}{V}[-(A_x - g\sin\theta)\cos\alpha\sin\beta + (A_y + g\sin\phi\cos\theta)\cos\beta \\ & - (A_z + g\cos\phi\cos\theta)\sin\alpha\sin\beta] + p\sin\alpha - r\cos\alpha\end{aligned} \tag{B.12}$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{B.13}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{B.14}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{B.15}$$

where V, α and β are the true airspeed, angle of attack and angle of sideslip of the aircraft. It should be noted that these equations can also be used to design flight control laws to reduce the influence of model uncertainties.

B.2. AIRSPEED-BASED KINEMATIC MODEL INCORPORATING THE EFFECTS OF IMU FAULTS

In this section, the airspeed-based kinematic model including the influence of IMU faults is presented.

The measurements of the IMU including faults are given as follows:

$$A_{xm} = A_x + \nu_{A_x} + f_{A_x} \tag{B.16}$$

$$A_{ym} = A_y + \nu_{A_y} + f_{A_y} \tag{B.17}$$

$$A_{zm} = A_z + \nu_{A_z} + f_{A_z} \tag{B.18}$$

$$p_m = p + \nu_p + f_p \tag{B.19}$$

$$q_m = q + \nu_q + f_q \tag{B.20}$$

$$r_m = r + \nu_r + f_r \tag{B.21}$$

where the subscript m denotes the measured variables, $\nu_{[\bullet]}$ denotes the noise in the corresponding sensors. f_{Ax}, f_{Ay} and f_{Az} represent the faults in the accelerometers. f_p, f_q and f_r represent the faults in the rate gyros.

These equations can be rewritten as follows:

$$A_x = A_{xm} - \nu_{A_x} - f_{A_x} \tag{B.22}$$
$$A_y = A_{ym} - \nu_{A_y} - f_{A_y} \tag{B.23}$$
$$A_z = A_{zm} - \nu_{A_z} - f_{A_z} \tag{B.24}$$
$$p = p_m - \nu_p - f_p \tag{B.25}$$
$$q = q_m - \nu_q - f_q \tag{B.26}$$
$$r = r_m - \nu_r - f_r \tag{B.27}$$

Substitute Equations (B.22)-(B.27) into Equations (B.1)-(B.6), we then obtain the airspeed-based kinematic model including IMU faults, which uses u_a, v_a, w_a, ϕ, θ and ψ as the states, as follows:

$$\dot{u}_a = v_a(r_m - f_r - \nu_r) - w_a(q_m - f_q - \nu_q) - g\sin\theta + (A_{xm} - f_{Ax} - \nu_{A_x}) \tag{B.28}$$
$$\dot{v}_a = -u_a(r_m - f_r - \nu_r) + w_a(p_m - f_p - \nu_p) + g\cos\theta\sin\phi + (A_{ym} - f_{Ay} - \nu_{A_y}) \tag{B.29}$$
$$\dot{w}_a = u_a(q_m - f_q - \nu_q) - v_a(p_m - f_p - \nu_p) + g\cos\theta\cos\phi + (A_{zm} - f_{Az} - \nu_{A_z}) \tag{B.30}$$
$$\dot{\phi} = (p_m - f_p - \nu_p) + (q_m - f_q - \nu_q)\sin\phi\tan\theta + (r_m - f_r - \nu_r)\cos\phi\tan\theta \tag{B.31}$$
$$\dot{\theta} = (q_m - f_q - \nu_q)\cos\phi - (r_m - f_r - \nu_r)\sin\phi \tag{B.32}$$
$$\dot{\psi} = (q_m - f_q - \nu_q)\frac{\sin\phi}{\cos\theta} + (r_m - f_r - \nu_r)\frac{\cos\phi}{\cos\theta} \tag{B.33}$$

The noise distribution matrix $\boldsymbol{G}$ can be readily extracted from these equations. The corresponding measurement model can be given as follows:

$$V_m = \sqrt{u_a^2 + v_a^2 + w_a^2} + \nu_V \tag{B.34}$$
$$\alpha_m = \text{atan}\left(\frac{w_a}{u_a}\right) + \nu_\alpha \tag{B.35}$$
$$\beta_m = \text{atan}\left(\frac{v_a}{\sqrt{u_a^2 + w_a^2}}\right) + \nu_\beta \tag{B.36}$$
$$\phi_m = \phi + \nu_\phi \tag{B.37}$$
$$\theta_m = \theta + \nu_\theta \tag{B.38}$$
$$\psi_m = \psi + \nu_\psi \tag{B.39}$$

where the subscript m denotes the measured variables, $\nu_{[\bullet]}$ denotes the noise in the corresponding sensors.

Similarly, the airspeed-based kinematic model including the IMU faults, which uses

V, α, β, ϕ, θ and ψ as the states, can be presented as follows:

$$\begin{aligned}\dot{V} &= (A_{xm} - w_{Ax} - f_{Ax} - g\sin\theta)\cos\alpha\cos\beta + (A_{ym} - w_{Ay} - f_{Ay} + g\sin\phi\cos\theta)\sin\beta \\ &\quad + (A_{zm} - w_{Az} - f_{Az} + g\cos\phi\cos\theta)\sin\alpha\cos\beta \end{aligned} \tag{B.40}$$

$$\begin{aligned}\dot{\alpha} &= \frac{1}{V\cos\beta}\big[-(A_{xm} - w_{Ax} - f_{Ax})\sin\alpha + (A_{zm} - w_{Az} - f_{Az})\cos\alpha + g\cos\phi\cos\theta\cos\alpha \\ &\quad + g\sin\theta\sin\alpha\big] - [(p_m - w_p - f_p)\cos\alpha + (r_m - w_r - f_r)\sin\alpha]\tan\beta \\ &\quad + q_m - w_q - f_q \end{aligned} \tag{B.41}$$

$$\begin{aligned}\dot{\beta} &= \frac{1}{V}\big[-(A_{xm} - w_{Ax} - f_{Ax} - g\sin\theta)\cos\alpha\sin\beta + (A_{ym} - w_{Ay} - f_{Ay} + g\sin\phi\cos\theta)\cos\beta \\ &\quad - (A_{zm} - w_{Az} - f_{Az} + g\cos\phi\cos\theta)\sin\alpha\sin\beta\big] + (p_m - w_p - f_p)\sin\alpha \\ &\quad - (r_m - w_r - f_r)\cos\alpha \end{aligned} \tag{B.42}$$

$$\dot{\phi} = (p_m - w_p - f_p) + (q_m - w_q - f_q)\sin\phi\tan\theta + (r_m - w_r - f_r)\cos\phi\tan\theta \tag{B.43}$$

$$\dot{\theta} = (q_m - w_q - f_q)\cos\phi - (r_m - w_r - f_r)\sin\phi \tag{B.44}$$

$$\dot{\psi} = (q_m - w_q - f_q)\frac{\sin\phi}{\cos\theta} + (r_m - w_r - f_r)\frac{\cos\phi}{\cos\theta} \tag{B.45}$$

The noise distribution matrix can be readily extracted and can be found in Chapter 2. The corresponding measurement model can be given as follows:

$$V_m = V + \nu_V \tag{B.46}$$

$$\alpha_m = \alpha + \nu_\alpha \tag{B.47}$$

$$\beta_m = \beta + \nu_\beta \tag{B.48}$$

$$\phi_m = \phi + \nu_\phi \tag{B.49}$$

$$\theta_m = \theta + \nu_\theta \tag{B.50}$$

$$\psi_m = \psi + \nu_\psi \tag{B.51}$$

B.3. AIRSPEED-BASED KINEMATIC MODEL INCORPORATING THE EFFECTS OF ADS FAULTS

In this section, the airspeed-based kinematic model including the influence of ADS faults is presented. The IMU sensors are assumed to be fault-free, which leads to the following condition:

$$f_{A_x} = f_{A_y} = f_{A_z} = f_p = f_q = f_r = 0. \tag{B.52}$$

First, if u_a, v_a, w_a, ϕ, θ and ψ are used as the states, the process model can be obtained by inserting Equation (B.52) into Equations (B.28)-(B.33).

The corresponding measurement model can be given as follows:

$$V_m = \sqrt{u_a^2 + v_a^2 + w_a^2} + f_V + v_V \tag{B.53}$$

$$\alpha_m = \text{atan}\left(\frac{w_a}{u_a}\right) + f_\alpha + v_\alpha \tag{B.54}$$

$$\beta_m = \text{atan}\left(\frac{v_a}{\sqrt{u_a^2 + w_a^2}}\right) + f_\beta + v_\beta \tag{B.55}$$

$$\phi_m = \phi + v_\phi \tag{B.56}$$

$$\theta_m = \theta + v_\theta \tag{B.57}$$

$$\psi_m = \psi + v_\psi \tag{B.58}$$

Secondly, if V, α, β, ϕ, θ and ψ are used as the states, the process model can be obtained by inserting Equation (B.52) into Equations (B.40)-(B.45).

The corresponding measurement model can be given as follows:

$$V_m = V + f_V + v_V \tag{B.59}$$

$$\alpha_m = \alpha + f_\alpha + v_\alpha \tag{B.60}$$

$$\beta_m = \beta + f_\beta + v_\beta \tag{B.61}$$

$$\phi_m = \phi + v_\phi \tag{B.62}$$

$$\theta_m = \theta + v_\theta \tag{B.63}$$

$$\psi_m = \psi + v_\psi \tag{B.64}$$

C

BASICS OF NONLINEAR DYNAMIC INVERSION AND INCREMENTAL NONLINEAR DYNAMIC INVERSION

In this appendix, the basics of NDI [179] and INDI, which were mentioned in Chapter 8, are introduced.

C.1. BASICS OF NDI

Consider the following nonlinear system:

$$\dot{\boldsymbol{x}} = \bar{\boldsymbol{f}} + \mathbf{G}\boldsymbol{u} \tag{C.1}$$

$$\boldsymbol{y} = \boldsymbol{x} \tag{C.2}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^n$ are the state, input and measurement vectors respectively. Note that in this thesis, we consider full state feedback. The function $\bar{\boldsymbol{f}}$ is a nonlinear function and $\mathbf{G}$ is assumed to be an invertible matrix. By inverting the system dynamics, we can obtain the following equation:

$$\boldsymbol{u} = \boldsymbol{G}^{-1}(\dot{\boldsymbol{x}} - \bar{\boldsymbol{f}}) \tag{C.3}$$

If we design the control input as follows:

$$\boldsymbol{u} = \boldsymbol{G}^{-1}(\boldsymbol{\nu} - \bar{\boldsymbol{f}}) \tag{C.4}$$

Then the dynamics of the system are:

$$\dot{\boldsymbol{x}} = \boldsymbol{\nu} \tag{C.5}$$

As can be seen, the system behaves like a pure integrator. Therefore, the dynamics of $\boldsymbol{x}$ can be controlled by a suitable choice of $\boldsymbol{\nu}$ which is usually designed using a linear

controller [177], based on the error $\boldsymbol{e}$ defined by

$$\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{x}_d \tag{C.6}$$

where $\boldsymbol{x}_d$ is the vector of desired states.

C.2. BASICS OF INDI

Assuming that $\boldsymbol{u}$ can change much faster than $\boldsymbol{x}$ [176], the dynamics of system Equation (C.1) can be rewritten into the following incremental form:

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_0 + \mathbf{G}\Delta\boldsymbol{u} \tag{C.7}$$

with

$$\dot{\boldsymbol{x}}_0 = \bar{\boldsymbol{f}}_0 + \mathbf{G}\boldsymbol{u}_0 \tag{C.8}$$

$$\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_0 \tag{C.9}$$

where $\dot{\boldsymbol{x}}_0$, $\boldsymbol{u}_0$, $\bar{\boldsymbol{f}}_0$ denote $\dot{\boldsymbol{x}}$, $\boldsymbol{u}$, $\bar{\boldsymbol{f}}$ in the previous time step respectively. $\Delta\boldsymbol{u}$ is the incremental input.

Rewrite Equation (C.7), we can obtain the following equation:

$$\Delta\boldsymbol{u} = \boldsymbol{G}^{-1}(\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_0) \tag{C.10}$$

Therefore, we can design the incremental control input as follows:

$$\Delta\boldsymbol{u} = \boldsymbol{G}^{-1}(\boldsymbol{\nu} - \dot{\boldsymbol{x}}_0) \tag{C.11}$$

Then dynamics of $\boldsymbol{x}$ can be controlled by a suitable choice of $\boldsymbol{\nu}$, which is mentioned in the Section C.1. The final control input is:

$$\boldsymbol{u} = \boldsymbol{u}_0 + \Delta\boldsymbol{u} \tag{C.12}$$

ACKNOWLEDGEMENTS

This dissertation is the result of four years of research performed within the Aerospace Engineering faculty at Delft University of Technology. This research was supposed to be a challenging journey. However, the following people, to who I am thankful, have made this journey much more smooth and enjoyable.

First of all, I would like to thank my promoter Prof. Max Mulder for his strategic guidance and all other support. Your support ensures that I am able to focus on my research. You put a lot of efforts in this thesis, which I really appreciate. Your rigorousness inspires me a lot. Also, thanks for giving me the opportunity to pursue my PhD and do research in this group.

I would like to thank my co-promoter Dr. Qiping Chu for his continuous guidance which leads to innovations during my research. I appreciate your insights which help me to look for the solutions. Your enthusiasm really inspires us to do creative research. Without your guidance and supervision, finishing this research project would never have been possible.

Dr. Erik-Jan van Kampen, my daily supervisor, provides a lot of support during my research project. I would like to thank you for your persistent guidance and supervision. You give me complete freedom to perform research, which I believe is important to enhance my ability of doing research independently. I really appreciate you for providing me such an easy and relaxing research experience. I am also thankful to Dr. Cornelis de Visser for his scientific support and discussions. Your rigorousness gives me more confidence in my work. I am thankful for your contributions to my journal publications.

My gratitude also goes out to Prof. Bob Mulder for recruiting me into this department. I clearly remember the interview when I was applying to do my research in this group.

Also, I would like to thank Laurens Van Eykeren for his scientific guidance and help. I really enjoy our discussions which give me a lot of thoughts. I also really appreciate that you offered me an exciting and memorable flying trip over Belgium in a small airplane. Also thank you for the home-made waffle.

My thanks also goes out to the people in the micro aerial vehicle lab. I thank Guido de Croon, Christophe De Wagter, Bart Remes, Erik van der Horst, Ewoud Smeur, Lodewijk Sikkel, Sjoerd Tijmons, for their scientific help.

My gratitude goes out to all the colleagues at the division of control and simulation: Bertine Markus, Jacco Hoekstra, René van Paassen, Daan Pool, Clark Borst, Joost Ellerbroek, Olaf Stroosma, Marilena Pavel, Alexander in 't Veld, Hans Mulder, Ferdinand Postema, Andries Muis, Harold Thung, Alwin Damman, Gijsbert den Toom, Herman Koolstra, Maarten Tielrooij, Rolf Klomp, Gustavo Mercado Velasco, Jan Comans, Deniz Yilmaz, Rita Valente Pais, Herman Damveld, Yazdi Jenie, João Caetano, Jan Smisek, Dyah Jatiningrum, Tommaso Mannucci, Jaime Junell, Sophie Armanini, Henry Tol, Tim Visser, Ivan Miletovic, Kirk Scheper, Emmanuel Sunil, Kasper van der El, Annemarie Landman,

Isabel Metz, Julia Rudnyk, Kimberly Mcguire, Dirk Van Baelen, Matěj Karásek, Kevin van Hecke, Lodewijk Sikkel, Andries Koopmans, Roland Meertens, Jerom Maas. Thank you all for the accompany during my stay and during the conferences. Haibo Zhang, Shaojie Zhang, Jia Wan, Liguo Sun, Tao Lu, Ye Zhou, Hann Woei Ho, Ye Zhang, Yingzhi Huang, Wei Fu, Junzi Sun, Sherry Wang, Shuo Li. We also had many enjoyable dinners and barbecues.

I would like to thank my friends Xuezhou Wang, Wandong Wang, Ye Zhang, Yuewei Ma, Hongxia Zhou, Qi Wang, Yannian Yang, Peijian Lv, Wei Yu, Zheng Zhou, Zi Wang, Qiuhua Chen, Dejian Kong. We had so many interesting parties and journeys. It is very nice to have you guys especially when we are abroad. I would also like to thank the buddies, who study in this university, for the enjoyable get-togethers. Zhiyong Wang, Huayang Cai, Peng Xu, Guangming Chen, Qingqing Ye, Shuanghou Deng, Bo Qu, Wuyuan Zhang, Jinhu Wang, Shijie Li, Jia Xu, Qujiang Lei, Jialun Liu, Tao Zou, Tao Lv, Huarong Zheng, Feifei Wang, Weiwei Wang, Zilong Wei, Yunlong Gong, Yunlong Yu, Ming Li, Kai Yuan, Kai Zhang, Likun Ma, Jicheng Feng, Xiangwei Liu, Yu Sun, Changgong Zhang, Tiantian Yao, Zhenpei Wang, Xi Zhang, Liang Zhang. I would also like to thank Qingbao Guan, Le Li, Jing Chu for their advice on living.

I also want to thank my previous supervisor during my Master study, Prof. Weiguo Zhang for his support. I also thank Xiaoxiong Liu and Guangwen Li for their support on me.

I am grateful to China Scholarship Council which provides me a lot of support during the PhD research.

Last but not least, I want to give my biggest thanks to my family for their persistent support. I thank my parents, Zelin Lu and Deyun Xu, for providing me with unconditional love. Without your support, I would not have been able to finish this thesis. Also many thanks to my brother and sister-in-law. I would like to thank my beloved wife Xiang Li for her accompany and support. Your unconditional love makes our lives better. Together with you we can create a wonderful future.

Peng Lu
Delft, March 2016

CURRICULUM VITÆ

Peng LU

14-01-1987 Born in Anhui, China.

EDUCATION

2005.9-2010.6 BSc in Automation

Northwestern Polytechnical University, China

Thesis: Research on Trajectory Optimization Methods for Large Commercial Aircraft

2010.9–2013.3 MSc in Modern Flight Control

Northwestern Polytechnical University, China

Thesis: Research on the Optimal Allocation Method of Control Systems and its Monte Carlo Assessment

2012.9 PhD in Aerospace Engineering

Delft University of Technology, the Netherlands

Research: Fault-Tolerant Control, Fault Detection and Diagnosis, state estimation, unmanned aerial vehicles, model identification

AWARDS

2010 Distinguished undergraduate student

2013 Distinguished Master degree dissertation

2014 Best student paper finalist

PROFESSIONAL ACTIVITIES

Session co-chair in the 2015 AIAA SciTech conference on Guidance, Navigation and Control.

Reviewer for Mechanical Systems And Signal Processing

Reviewer for European Journal of Control

Reviewer for Journal of Robotics and Mechatronics

Reviewer for International Conference on Robotics and Automation (ICRA)

Reviewer for AIAA SciTech conference on Guidance, Navigation and Control.

Reviewer for International Conference on Control and Fault-Tolerant Systems

LIST OF PUBLICATIONS

JOURNAL PAPERS

1. **P. Lu**, E. van Kampen, C. de Visser, Q. Chu. Nonlinear Aircraft Fault-Tolerant Trajectory Control Using Incremental Nonlinear Dynamic Inversion. *Control Engineering Practice*, 2016 (under review).

2. **P. Lu**, E. van Kampen, C. de Visser, Q. Chu. Nonlinear Simultaneous Aircraft Sensor and Actuator Fault-Tolerant Flight Control. *Journal of Guidance, Control and Dynamics*, 2016 (under review).

3. **P. Lu**, E. van Kampen, C. de Visser, Q. Chu. Novel framework to state and unknown input estimation for time-varying systems. *Automatica*, 2016, http://dx.doi.org/10.1016/j.automatica.2016.07.009.

4. **P. Lu**, L. Van Eykeren, E. van Kampen, C. de Visser, Q. Chu. Double-model adaptive fault detection and diagnosis applied to real flight data. *Control Engineering Practice*, 2015(36), 39-57.

5. **P. Lu**, E. van Kampen, C. de Visser, Q. Chu. Nonlinear aircraft sensor fault reconstruction in the presence of disturbances validated by real flight data. *Control Engineering Practice*, 2016(49), 112-128.

6. **P. Lu**, L. Van Eykeren, E. van Kampen, C. de Visser, Q. Chu. Adaptive Three-Step Kalman Filter for Air Data Sensor Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics*, 2016, 39(3):590-604

7. **P. Lu**, L. Van Eykeren, E. van Kampen, C. de Visser, Q. Chu. Aircraft Inertial Measurement Unit Fault Identification with Application to Real Flight Data. *Journal of Guidance, Control and Dynamics*. 2015, 38(12), 2467-2475.

8. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu. Selective-Reinitialization Multiple Model Adaptive Estimation for Fault Detection and Diagnosis. *Journal of Guidance, Control and Dynamics*, 2015, 38(8), 1409-1424.

BOOK CHAPTERS:

1. **P. Lu**, E. van Kampen, Q. Chu. Nonlinear Quadrotor Control with Online Model Identification. *Advances in Aerospace Guidance, Navigation and Control*, 81-98.

2. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu. Sensor Fault Detection and Estimation for Quadrotors Using Kinematic Equations. *Advances in Aerospace Guidance, Navigation and Control*, 363-379.

PEER-REVIEWED CONFERENCE PUBLICATIONS:

1. **P. Lu**, E. van Kampen. Active fault-tolerant control for quadrotors subjected to a complete rotor failure. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4698 - 4703.

2. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu. Active Fault-Tolerant Control System using Incremental Backstepping Approach. *AIAA Guidance, Navigation, and Control Conference*, 2015, 1-17. **Nominated for Best Student Paper Award**

3. **P. Lu**, E. van Kampen, Q. Chu. Robustness and Tuning of Incremental Backstepping Approach. *AIAA Guidance, Navigation, and Control Conference*, 2015, 1-15

4. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu. Air Data Sensor Fault Detection and Diagnosis with Application to Real Flight Data. *AIAA Guidance, Navigation, and Control Conference*, 2015, 1-18.

5. **P. Lu**, E. van Kampen. Selective-Reinitialization Multiple Model Adaptive Estimation for Fault Detection and Diagnosis. *AIAA Guidance, Navigation, and Control Conference*, 2014, 1-24. **Nominated for Best Student Paper Award**

6. **P. Lu**, E. van Kampen. Aircraft Inertial Measurement Unit Fault Identification with Application to Real Flight Data. *AIAA Guidance, Navigation, and Control Conference*, 2015, 1-20

7. **P. Lu**, E. van Kampen, Q. Chu. Nonlinear Quadrotor Control with Online Model Identification. *Proceedings of the 3rd CEAS Specialist Conference on Guidance, Navigation and Control*, 2015, 1-19.

8. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu. Sensor Fault Detection and Estimation for Quadrotors Using Kinematic Equations. *Proceedings of the 3rd CEAS Specialist Conference on Guidance, Navigation and Control*, 2015, 1-17.

9. **P. Lu**, L. Van Eykeren, E. van Kampen, Q. Chu, B Yu. Adaptive Hybrid Unscented Kalman Filter for Aircraft Sensor Fault Detection, Isolation and Reconstruction, *AIAA Guidance, Navigation, and Control Conference*, 2014, 1-18.

10. **P. Lu**, E. Van Kampen, B Yu. Actuator Fault Detection and Diagnosis for Quadrotors. *IMAV 2014: International Micro Air Vehicle Conference and Competition*, 2014.

11. D. Jatiningrum, **P. Lu**, C. De Visser, R. van Paassen, M. Mulder. Development of a New Method for Calibrating an Angular Accelerometer Using a Calibration Table, *AIAA Guidance, Navigation, and Control Conference*, 2015, 1-13.

12. B. Yu, Y. Zhang, Y. Yi, Y. Qu, **P. Lu**. Fault detection for partial loss of effectiveness faults of actuators in a quadrotor unmanned helicopter. *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, 2014, 3204-3209.

ISBN: 978-94-6186-701-8