CRANFIELD UNIVERSITY

DAVID TORRES OCAÑA

ADAPTIVE AND FAULT-TOLERANT FLIGHT CONTROL SYSTEMS

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Autonomous Vehicle Dynamics And Control

MSc THESIS
Academic Year: 2013 - 2014

Supervisor: Hyo-Sang Shin and Antonios Tsourdos
August 2014

CRANFIELD UNIVERSITY

SCHOOL OF AEROSPACE, TRANSPORT AND MANUFACTURING
Autonomous Vehicle Dynamics And Control

MSc THESIS

Academic Year 2013 - 2014

DAVID TORRES OCAÑA

ADAPTIVE AND FAULT-TOLERANT FLIGHT CONTROL SYSTEMS

Supervisor: Hyo-Sang Shin and Antonios Tsourdos
August 2014

This thesis is submitted in partial fulfilment of the requirements for the degree of Master of Science

ABSTRACT

Adaptive and Fault-Tolerant control are powerful tools when considering Flight Control Systems design. This thesis proposes two different solutions in the form of two control systems in that field.

The framework considered is an Explicit Model Following Direct Adaptive Control scheme using Adaptive Neural Networks. Non-linear Dynamic Inversion, Control Allocation and Reference Model Following are some of the techniques implemented. As well, in order to perform Fault-Tolerant control, System Identification and Fault Detection and Isolation techniques are used, whereas Fault Detection and Isolation was not implemented but simulated.

The solutions were tested, developed and validated using modified models of F-16. First modification explores the control surfaces redundancies of the aircraft, enabling the control system to command each control surface individually. The other modification of the model enable the option of reconfiguring the model during simulation.

Validation, analysis and comparisons of the proposed solutions are carried out. Proposed solutions complied with applicable specifications to the aircraft model used, showing an adequate performance, and performed well under deviations, faults and damages providing the adaptation and/or reconfiguration expected, fulfilling the initial objectives.

The potential applications of the philosophies mentioned and solutions proposed are virtually unlimited. For instance, they can be applied to UAVs, manned military airplanes or airliners, providing enhance capabilities.

Benefits of the proposed architectures are their flexibility and ability to reduce costs of development and operation. As well, they provide the capabilities to accommodate changes, faults and damages on ground or in mid-air, letting the aircraft enough manoeuvrability as physically possible and, eventually, a safe landing.

Keywords:

Adaptive Neural Networks, Non-linear Dynamic Inversion, Control Allocation, System Identification,Explicit Model Following, Direct Adaptive Control, F-16 model

ACKNOWLEDGEMENTS

I would like to thank my family for supporting me for so many years and encourage me to pursue in my career in spite of all the time it took me away from them.

Thanks to my supervisors Hyo-Sang Shin and Antonios Tsourdos for helping me to carry out this work, and to ERASMUS program which allow me to study at Cranfield University.

Thanks to all researchers that worked on the matter of this Thesis, since their efforts and contributions have bring us to a point where amazing things can be done in this field and have enable me to accomplish this work.

Special thanks to Ola Härkegård, Guillaume J.J. Ducard and A. Calise for their contributions in Adaptive and Fault-Tolerant control and to R. S. Russell for his work on the model used.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

AC	Aerodynamic Centre
AC	Aerodynamic Centre
AFTI	Advanced Fighter Technology Integration
AoA	Angle of Attack
BAC	Body Axes centre
CA	Control Allocation
CG	Centre of Gravity
DI	Dynamic Inversion
EAS or V_{EAS}	Equivalent airspeed
EKF	Extended Kalman Filter
EMMAE	Extended Multi Model Adaptive Estimation
FCS	Flight Control System
FDI	Fault Detection And Isolation
FDIR	Fault Detection Isolation And Recovery
GD	Gradient Descendent
KF	Kalman Filter
LEF	Leading Edge Flap
LQG	Linear-Quadratic-Gaussian
LQR	Linear Quadratic Regulator
MIAC	Model Identification Adaptive Control
MMAE	Multi Model Adaptive Estimation
MMST	Multiple Model Switching And Tuning
MPC	Model Predictive Controller
MRAC	Model Reference Adaptive Control
NDI	Non-Linear Dynamic Inversion
NMPC	Non-Linear Model Predictive Controller
NN	Neural Networks
PD	Proportional-Derivative Controller
PI	Proportional-Integral Controller
PID	Proportional-Integral-Derivative Controller
PSO	Particle Swarm Optimization
SI	System Identification
SLS	Sequential Least Squares

SMC	Sliding Mode Control
SS	Sideslip
TAS or V_{TAS}	True airspeed
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
UIDFO	Unknown Input Decoupled Functional Observer
UKF	Uncested Kalman Filter
VTOL	Vertical Take-Off And Landing
WLS	Weighted Least Squares

1 Introduction

Flight Control Systems are essential in most of the current aircrafts that allows controlling the vehicle in a simplified and safe manner.

Since the beginning of aviation, Flight Control Systems have evolved from simple human-powered control surfaces to the complex systems they currently are, enabling the pilot to perform complicate manoeuvres, increase the capabilities of the aircraft and achieve a safe flight.

Many challenges in this field still have to be achieved, even more taking into account the lately rise of Autonomous Systems such as Autonomous Unmanned Vehicles (UAVs) which are one of the most important potential applications of the work presented here.

1.1 Background and motivation

Flight Control Systems (FCS) are very expensive to develop, implement and certificate taking into account the current technology being used.

Most of these systems today are implemented using classical control techniques that have been used for many years proving their effectiveness. But developers of these systems usually face a great amount of work when developing a new aircraft regarding FCS alone.

Development of these systems implicates nowadays firstly, a good modelling of the plant or aircraft to control, including as many details and accuracy in modelling as possible. This means a lot of time and effort performing wind tunnel test, aerodynamics analysis, mass properties analysis, actual flight tests, etc. This implicates that developers of FCS for new aircrafts have to face an incredible amount of work when modelling.

Secondly, developers implementing currently used techniques in control, such as classical control approaches, have to rely in believe in the validity of the model to develop the control algorithms. These current techniques usually requires a big amount of design effort to build the architecture and tune its parameters, which is a tough task taking into account the amount of parameters implicated and that their absence of physical meaning. But, due to the complexity of the models developed, since aircrafts are asked to perform many different task given their cost, the developer has to develop the control system to cover all the peculiarities and variations of the model and, even more difficult, demonstrate stability for all these cases.

An example of variations in a model is an effect, very known by developers, such as the change of the dynamics properties of the aircraft depending on speed and altitude. This is usually represented by an aerodynamic model that gives the dynamics properties of the aircraft in all point inside the flight envelope, that is, variations of the properties with Mach number and altitude. Hence, developers usually have to design and tune the control system for all flight conditions inside flight envelope, something that can be an endless task.

And, to make the matter worse, the entire design of the system is carried out relying solely on the validity of the model that may not represent the reality in an accurate manner. Hence, developers have achieved a design that maximizes the stability margin in all flight conditions (Robustness), that is, maximize the deviation of the model form reality that the control system is able to overcome. These stability margins are complicated to achieve and complicated to delimitate so that, at the end, there exists a certain level of uncertainty about what the control system is capable of overcoming.

Finally, the entire design and aircraft have to be certified so that they comply with the specifications they are asked to comply and can be operated by the user. This also means that developers have to demonstrate the capabilities of the control system and aircraft, and also demonstrate a certain level of robustness.

As well, there exist cases in which already existing aircrafts are needed to implement new configurations that may alter their dynamics. Such changes may be represented by an incorporation of new fuel tanks or implementation of new winglets. Such changes implicate that the new configuration has to be analysed and modelled again. As well, after the re-modelling stage, developers have to redesign and tune the control system, so that the new design provides, at least, the same capabilities with the new aircraft as it did before. And, finally, the new design and new configuration have to be certified again. All that means lots of effort, time and money for the developers and, even worse, considering it is not the first time all those tasks had to be done.

A second problem is the case in which the aircraft receives damage or a fault in mid of the air. Historically mid-air collisions and/or control surfaces faults meant an almost sure failure of the system and a consequent accident. As new technology evolved, aircrafts had become more and more redundant in order to overcome control surfaces problems but there still a void in the used technology that controls them. So that, there exist needs of systems that command the aircraft effectively, in consequence of the aircraft health,

state and control commands and taking advantage of the redundancies of control surfaces.

Popular cases of mid-air collisions and successful landings are not unusual. For instance, Figure 1-1 shows the case of an F-15D that collided with an A-4 Skyhawk in mid-air, losing its right wing completely.

Figure 1-1 Israeli F-15D mid-air collisions and landing

The most impressive part of this case is that, even given the severity of the damage, the pilot was able to land the airplane safely. Accordingly with reports of the accident and [1] the aircraft initially started spinning in an apparent uncontrollable way. Then the pilot increased speed enabling him to recover authority and stabilizing the aircraft. The aircraft was able to maintain a sustain flight and safe landing due to its big elevators surfaces and remaining lifting surfaces. This case evocates to idea that, even under big damages and faults, something else can be done and a safe landing can be achieved. So, such capabilities should be developed in FCS, since most of the current systems are unable to overcome this kind of problems.

1.2 Aims and objectives

Given the presented background and capabilities and characteristics of current FCS, the aim of this work can be summarized as: Design adaptive and fault-tolerant control systems that allow changes in the aircraft configuration, reduce development effort of new systems and be tolerant to faults.

Hence:

- The first objective is to investigate about the current state of art about adaptive and fault-tolerant control systems, choose those philosophies that present better capabilities and develop and implement new algorithms and techniques if needed.
- The second main objective is to develop an adaptive flight control system that, using an adaptation feature, allows changes in configuration of the aircraft, reduces the development effort and overcome in-flight deviations and faults.
- Finally, the third objective is to design a fault-tolerant control system that, using a reconfiguration feature in addition to the adaptive one, allows the aircraft to be recovered from faults and damages such as foreign objects impacts and letting the mission to be accomplish or the pilot to land the aircraft safely.

1.3 Work presented

Work presented in this Thesis is focused in solving the main two challenges the section 1.2 states. That is, two architectures are presented both with a different level of complexity, capabilities and performances. These approaches are presented in next section.

These proposed architectures are composed by many algorithms and mathematical methods that have to be developed beforehand and once the main components are completed, they are assembled so that they perform in an overall solution.

These methods and techniques are selected mainly through the study of the state of art that, given its relative maturity, allows an easy understanding of the problem and the capabilities of each method related with the overall problem. As well, by the experience that simulations of the implemented controllers brought, some other techniques were found necessary and some other algorithms had to be developed in order to solve the found problems.

1.3.1 Approaches

This work presents two concepts that have different level of complexity and have different capabilities.

The first approach is called Basic Concept, which is based in an Adaptive strategy that allows achieving adaptation and the goals of the second objective.

This Basic or Adaptive concept, showed in Figure 1-2, is based mainly in something called Explicit model following in which the control system is implemented through a desired dynamics or reference model with a simple controller and, using Non-linear Dynamic Inversion (NDI) and Control Allocation (CA), the control commands are generated, commanding each control surface individually and taking advantages of the aircraft redundancies in terms of control surfaces.

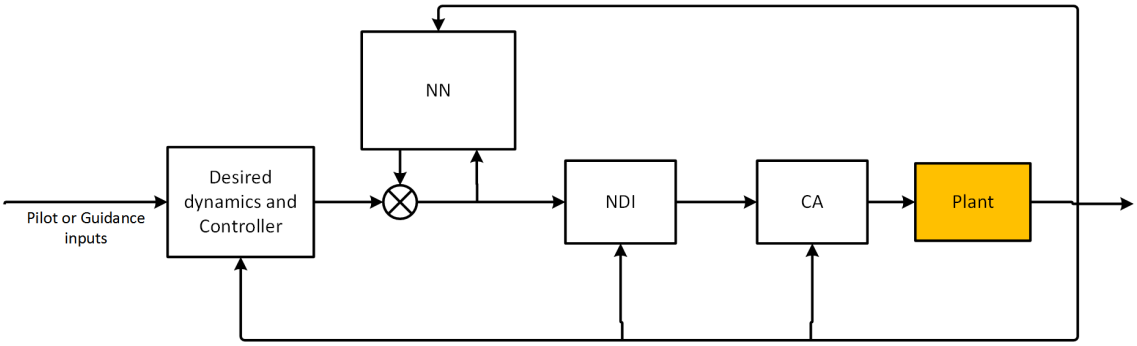


Figure 1-2 Basic Concept

Adaptation is achieved by implementing Adaptive Neural Networks (NN in Figure 1-2) that affectively provides a stabilization framework to the entire system, adapting changes, deviations and faults.

The second approach is called Advanced concept and it is intended to fulfil a more complex task than the previous controller does. It is oriented to perform Fault-Tolerant control and achieve the goals of the third objective stated in previous section.

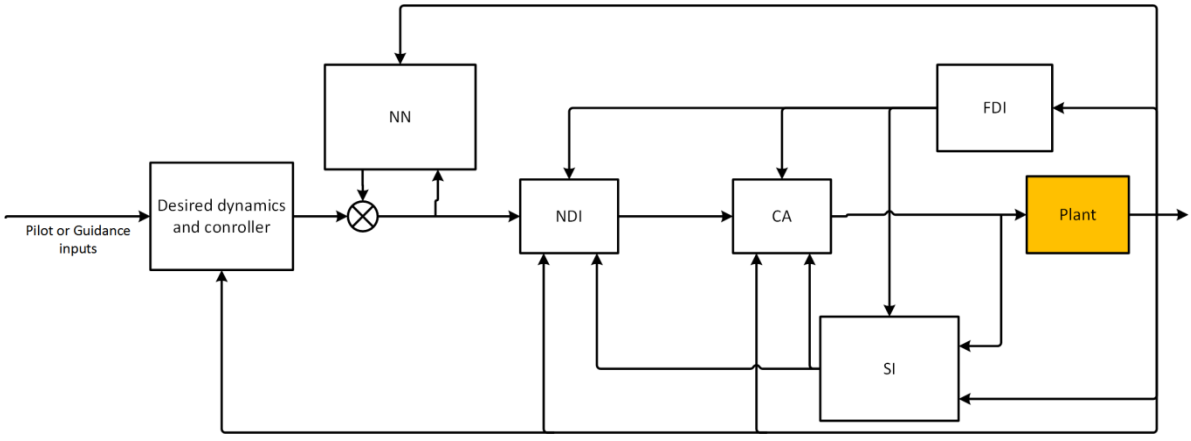


Figure 1-3 Advanced Concept

This Advanced Concept, in Figure 1-3, is based in the Basic one sine it implements the same controller and adaptation framework, but it also implement a Fault Detection and Isolation system (FDI) and a System Identifications module (SI).

This upgraded concept based its reconfiguration capabilities in the information provided by the FDI and SI module that allows the controller not only to deal with overactuated systems but to effectively take advantage of them performing reallocation of control effort. As well, SI module identifies key parameters of the aircraft dynamics which allows obtaining as much manoeuvrability as physically available in a damages receiving case.

1.3.2 Algorithms and techniques developed

1.3.2.1 Aircraft models and modifications

In order to test and validate the solutions proposed, a model of the fighter aircraft F-16 is implemented and used.

In order to get all capabilities of the aircraft and obtain an overactuated system, modifications were made so that the aircraft can be commanded using 7 control surfaces: Left/right elevator, left/right aileron, rudder and left/right LEF, where LEF means Leading Edge Flap.

But, due to the purpose of the work, further modifications were made to the model so that deviation, faults and damages can be implemented. This reconfigurable model includes enough parameters for representing the deviation, faults and damages stated in section 1.4.

1.3.2.2 Non-linear Dynamic Inversion and Control Allocation

Non-linear Dynamic Inversion (NDI) method was used to produce a module that performs this technique using a representation of the model previously described. Similarly Control Allocation (CA) technique is implemented to perform a final inversion and generate control surface deflection, taking advantage of the overactuated aircraft.

Depending on whether or not Fault-tolerant control is performed, reconfigurable model in NDI and CA is used or not.

1.3.2.3 Explicit Model Following

The framework in which control indeed is performed is something called Explicit Model Following or Reference Model Following. It is based on the implementation of a desired dynamics that the developer designs accordingly to his needs or design specifications.

In fact, this reference model can be understood as a shaping filter that shapes the inputs to the control system.

The key advantage of this technique is that the design is simplified in a notorious way if compared with classical control approach, since it uses a minimal number of parameters which have a physical meaning, so that the designer finds the tuning more intuitive and easy.

Also, another big advantage is that these parameters are not needed to change depending on the flight condition or flight phase, since these changes are taken into account by the NDI and CA algorithms.

1.3.2.4 Adaptive Neural Networks

Neural Networks have been used in many field of technology, including control theory. The method used here is an Adaptive Neural Network that performs learning or adaptation as the flight takes place. This allows the system to perform with an adaptive capability that allows accommodation of deviations, faults and damages on the aircraft.

The main advantages of these Adaptive Neural Networks for control are that they provides stabilization when deviations, faults and damages occur (the event) so that the need of System Identification or a fast identification after the event is no more an issue. Similarly, the need of information of the health of the control surfaces right after the event is also reduced. That allows to FDI and SI, in case of performing them, not to be crucial for survivability after such events.

Another important characteristic of this technique is that the need for an accurate model of the aircraft, which usually requires an off-line analysis and system identification, for designing the flight control system is reduced significantly.

Also, for the development of the control system stage, the need of in-flight tuning, validation of gain schedules, implementation of multiple controllers and switchers and their logic, if faults are taken into account, are greatly reduced.

So, this technique is found to be the right tool to solve most of the problems stated in section 1.1 and also achieve the second objective of this work: An adaptive controller.

1.3.2.5 System Identification and Fault Detection and Isolation

In order to overcome faults and damages and not being fundamental for stability but for manoeuvrability, System Identification and Fault Detection and Isolation techniques are

used. These two algorithms provide to the Advanced concept superior capabilities in comparison with the Basic concept.

System Identification can be performed using many techniques but in this work it was implemented using an Uncested Kalman Filter, which allows not only to perform identification but also to assess the quality of these estimations.

Due to its complexity, large development time and computational cost, Fault Detection and Isolation was not implemented in this work. Instead, responses and performances of typical methods presented in literature were simulated so that a realistic behaviour is achieved.

1.3.2.6 Other implemented algorithms

In order to make easier the development process and the usage of the simulation routines developed, secondary algorithms were developed. Note that the final product of this work is a simulation environment where models, controllers proposed and secondary algorithms are implemented.

The features included in the development of this work are:

- Piloting with a joystick and representation in real time using Flight Gear software
- Speed controller
- Auto-trim and auto-sensibility systems
- Stall protection and warning system
- Switching system between control in load factor for high speeds and angle of Attack (AoA) at low speeds.
- Replication of typical sensors in aircrafts using actual specifications. Sensors assumed that are implemented are: GPS, INS, AoA and Sideslip vanes, Pitot tube and static pressure ports

As well, a Simulink® library was developed including most of the most important algorithms implemented, so that future developers can make usage of it.

Hence, the purpose of all these implementations is, not only to provide a realistic simulation, but to allow future users to better understand the capabilities of the solutions proposed and to provide future developers a framework where to further develop the solutions proposed or even develop new ones.

1.3.3 Validation, Analysis and Comparisons

In order to verify the implemented controllers a specification compliance assessment was carried out using MIL F-8587C.

As well, in order to understand the working principle of many algorithms implemented, several test are carried out, including deviations, faults and damages injections.

In order to make evident the different capabilities of solutions proposed, comparisons with a classical controller is made for different cases so that, individual contributions of the algorithms are outlined.

Finally, in order to show the performances of the Basic and Advanced concept and to delimitate the capabilities of the Basic one, a comparison between these two is carried out.

1.4 Deviations, faults and damages

In this section, the meanings of deviations, faults and damages used in this work are presented.

“Deviation” is understood as an allowed change in a parameter of the aircraft that does not make a significant modification in the dynamics of the aircraft, for instance, changes in mass and moments of inertia.

“Fault” is usually defined as: “An unpermitted deviation of at least one characteristic property (feature) of the system from the acceptable, usual or standard condition”. For instance, a loss of a control surface.

“Damage receiving” is understood as an extreme deviation that can be considered as fault, for instance, a significant loss lift or drag. It is classify inside fault.

“Failure” is understood as: “A permanent interruption of a system’s ability to perform a required function under specifies operating conditions”. So, it is understood that failure occurs when one or more faults and/or a damage receiving event results in a not feasible operation of the aircraft. For instance, a crash of the airplane due to the loss of control.

In this work different kind of fault, deviations and damages (issues) are taken into account (see section 3.3.2 for interpretation):

- Control surfaces:
    - Loss part of surface
    - Loss of effectiveness due to external causes
    - Aerodynamics interference
    - Icing
    - Loss of authority: Blockages where hard-overs are understood as blockages as well
    - Loss of authority: Floating or total loss
- Airframe or structural
    - Loss of lifting surface
    - Change in shape: Change dragging properties
- Globally
    - Changes in aerodynamics characteristics
    - Mass properties variations

An useful classification of these issues can be:

Faults	**Deviations**
• Control surfaces: o Loss of authority: blockages o Loss of authority: Floating or loss • Structure: o Loss of lifting surface	• Control surfaces: o Loss part of control surface o Loss of effectiveness due to external causes o Aerodynamics interference o Icing • Structure: o Loss of part lifting surface o Change in shape: Change dragging properties • Globally: o Changes in aerodynamics characteristics o Mass properties variation
Need reconfiguration	Needs adaptation

Figure 1-4 Fault and deviations classification

2 Literature review

In this section an examination of the actual state of art and technology is done, incurring on those technologies that are more similar to the methodology used in this thesis.

2.1 Introduction

Reconfigurable control is a matter widely studied since the 70s and there exist and existed many publications, literature, technological demonstrators and programs regarding reconfigurable or adaptive philosophies.

It should be said about literature that terminology varies depending on the time the publication was done and even different authors give different names to the same things. Although an extensive reading along resources was carried out, it was impossible to agree terminology used by the different authors. Nevertheless, here the most common and successful techniques are presented.

2.2 Classification

Many ways to classify Reconfigurable control exist today, but the first and most simple one is shown in Figure 2-1.

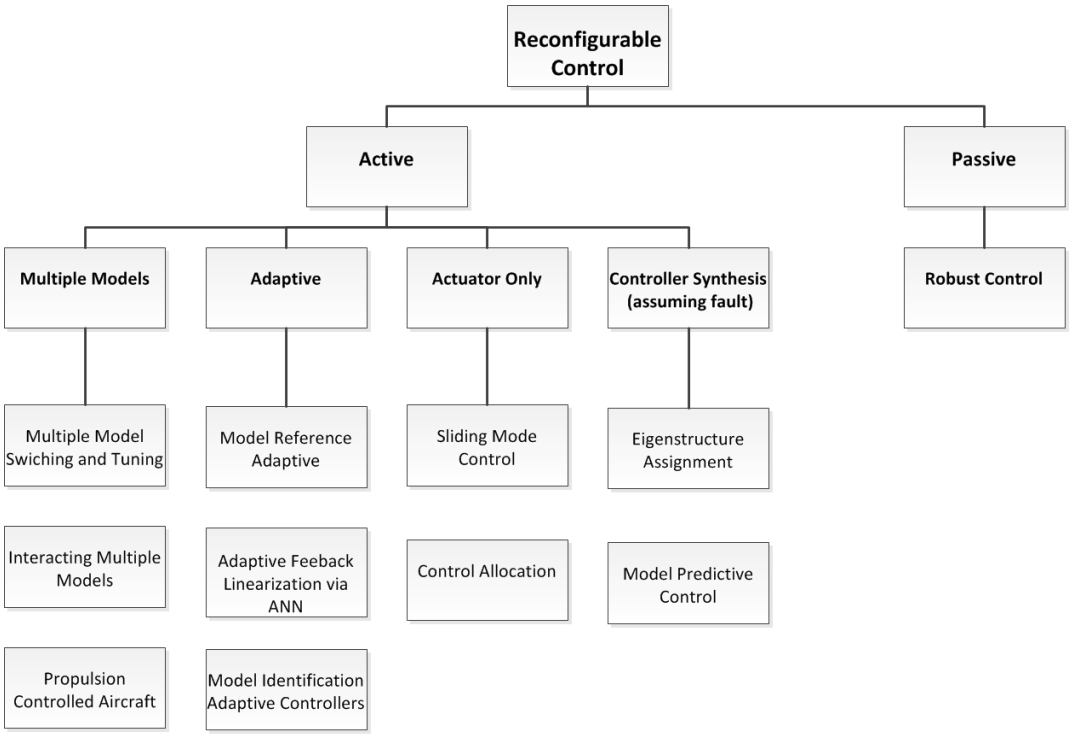


Figure 2-1 Reconfigurable Control Classification

The important part is the Active Reconfigurable Control where many efforts actually aim. Multiple models approach, like Multiple Model Switching and Tuning (MMST), or Adaptive ones have been used in several times as said bellow.

Regarding Adaptive methods in [2] several classifications are done. This field has been without doubt the main core of the efforts since the beginning of Control theory and Adaptive control is still used meaning reconfigurable.

As said in [2], in general, it should be distinguished between Direct methods and Indirect methods.

Direct methods use the estimated parameters directly in the adaptive controller. In contrast, indirect methods use estimated parameters to calculate required controller parameters.

A better definition, but slightly different, of these two is given in [3]:

> *"Indirect control: The basic philosophy of this approach is to estimate the parameters of the unknown plant from input output data and, in turn, use these estimates to adjust the parameters for a controller so that the transfer function of the controlled plant evolves to that of the model.*
>
> *Direct control: Unlike the previous case, no explicit plant identification is involved here. However, explicit use of a model described by equation (3) is used and the generated error e(t) is directly used in the adaptive control laws."*

Other classification, also done in [2], is:

- Dual Adaptive Controllers [based on Dual control theory]
  - Optimal Dual Controllers [difficult to design]
  - Suboptimal Dual Controllers
- Non-dual Adaptive Controllers
  - Adaptive Pole Placement
  - Extremum Seeking Controllers
  - Iterative learning control
  - Gain scheduling
  - Model Reference Adaptive Controllers (MRACs)
    - Gradient Optimization MRACs
    - Stability Optimized MRACs
  - Model Identification Adaptive Controllers (MIACs)
    - Cautious Adaptive Controllers (use current SI to modify control law, allowing for SI uncertainty)

- Certainty Equivalent Adaptive Controllers (take current SI to be the true system, assume no uncertainty)
    - Nonparametric Adaptive Controllers
    - Parametric Adaptive Controllers
        - Explicit Parameter Adaptive Controllers
        - Implicit Parameter Adaptive Controllers

The most widely used and discussed techniques from the previous classifications may be the MRACs and MIACs methods, where big efforts and researching have been done.

In Figure 2-2 MRAC method is shown. This method is based in a Reference model or desired behaviour and uses the information from the actual plant and the desired response to adjust the control law. Generally this adjustment varies depending on how it is implemented. It depends on if direct or indirect approach is used or if explicit or implicit model is used.

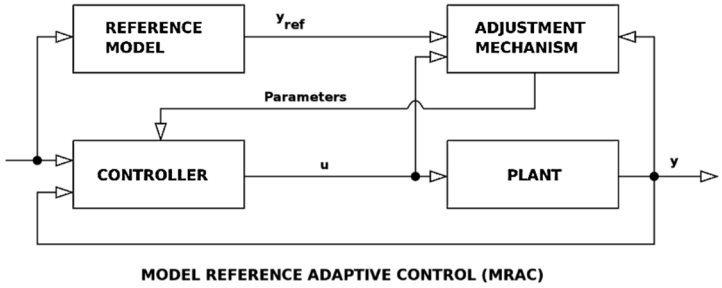


Figure 2-2 MRAC method. Source [1]

The other approach is MIACs method, in Figure 2-3 , where an online SI is performed in real time and the parameters used to adjust the controller. Here, again, direct or indirect methods can be used, but generally direct approaches are more suitable.

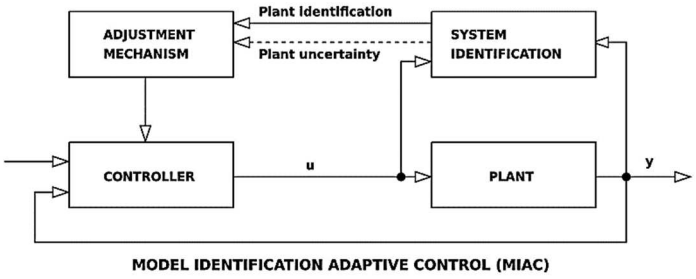


Figure 2-3 MIAC method. Source [1]

Regarding the problem of overcome failures or faults, Fault Detection Isolation and Recovery (FDIR) refers to all the stages of the solution, that is, it refers to Fault detection in first instance and Isolation and recovery as solution to the issue. So, FDIR algorithms are oriented to fault-tolerant control philosophies.

A complete picture of this can be found in [4] where FDI and SI techniques are explained and compared. As well, reconfiguration approaches are treated. Multiple Models approach is exposed as including different techniques like Multiple Model Switching and

Tuning or based in Multiple Models Adaptation Estimation. Interactive Multiple Models are also mentioned as an approach well considered in literature.

In the other hand, [4] also considers a solely adaptive approach in its two variants: indirect and direct. The author explains the characteristics of each method and focus more in some studies about direct adaptive studies like [5].

2.3 Brief overview of history

Marc Steinberg in [6] does an outline of the entire history of Reconfigurable control systems, where several programs and their results and legacies are explained. This publication has been extremely useful due to the condensation of data and information. The author also uses a common terminology and makes simpler the understanding of the concepts. Here just simple mentions of the most important milestones and most interesting concepts are done.

2.3.1 Early stages

In 1982, NASA sponsored a workshop in what was called “restructurable controls” [7]. The definition of restructurable control includes these aspects, as said in [6]:

- A method to measure the effectiveness of the current control mode
- A technique to identify the control(s) which have been lost when the above measure exceeds a threshold
- A manner to determine the characteristics of the remaining controls
- A routine that can redesign control laws for the remaining flight controls without the intervention of a controls engineer

Reconfigurable control was then defined as an alternative approach that complies with the first two points.

At 1980 the USAF with GE start working on a concept for a reconfigurable control system to overcome problems in actuators and damage in lifting surfaces. The concept developed uses a pseudo-inverse of a linear model of aircraft using a Kalman filter for FDI.

In 1984 the USAF start the program SRFCS. The efforts of the program were focus on F-16 and a fictitious aircraft called CRCA that would be lately build. These two were done by two different companies like GE and Grumman. Once again, in both cases pseudo-inverse was used as main approach for controller.

At the end of the 80's other publications like [8] used an implicit model following where it was not used directly a model of plant but a desired dynamics to follow.

Some flight test and simulated flights were done at the end of the 80's as well, carried out by NASA in the program SRFCS using F-15 as platform for flights. Only limited results and experiments were performed, although it proves that approaches like pseudo-inverse could be good solutions when reconfiguration is performed but only if FDI provides good performance.

Other programme at Dryden for only propulsion control of aircraft used switching approach with conventional control techniques. FDI was proposed and used in real flight with F-15 to be done by the pilot.

At the beginning of the 90's computational power increased and new non-conventional methods rose.

2.3.2 Adaptive and intelligent control approaches

As said, since computational power increased on 90's, new more complex and computationally expensive techniques appeared.

At latest years of 80's machine learning becomes to gain interest among the community, since it can be used, and was used, in control and FDI areas.

At [9] MRAC approach is explored and considered and three approaches of MRAC are explored and compared at [5]. At [10] Multiple Model Adaptive control is explored and LQR architecture indirect adaptive control is looked at [11] which is bases on a plant of F-16.

Simple direct explicit model following using linear model is explored at [12] where simple feedback using MRAC philosophy is used to perform model following.

Receding Horizon Optimal Control Law first looked at on-line parameters identification whose worse point is obviously the need of a fast identification. Several issues in system ID were addressed in the 90's like parameters overestimation or divergence. Publications like [13], [14] , [15] and [16] addressed different techniques in system ID and solutions for the stated problems.

Architectures based on NN like [17] used two different NN to invert nonlinearities using a dynamic inversion, that is, use the NN to correct the errors of modelling of a linear

dynamic inversion. This methodology was later used by the same author in different applications like missiles [18], tilt-rotor aircrafts [19], etc.

An important program called RESTORE, made significant advances in practical application of adaptive control in a tailless aircraft at [20]. Later, neuro-adaptive approach with an X-36 model was demonstrated in [21]. Then, the general structure of the controller was modified and the architecture becomes an explicit model following with ANN to correct deviations of a NDI in [22] , whose general scheme is shown in Figure 2-4.

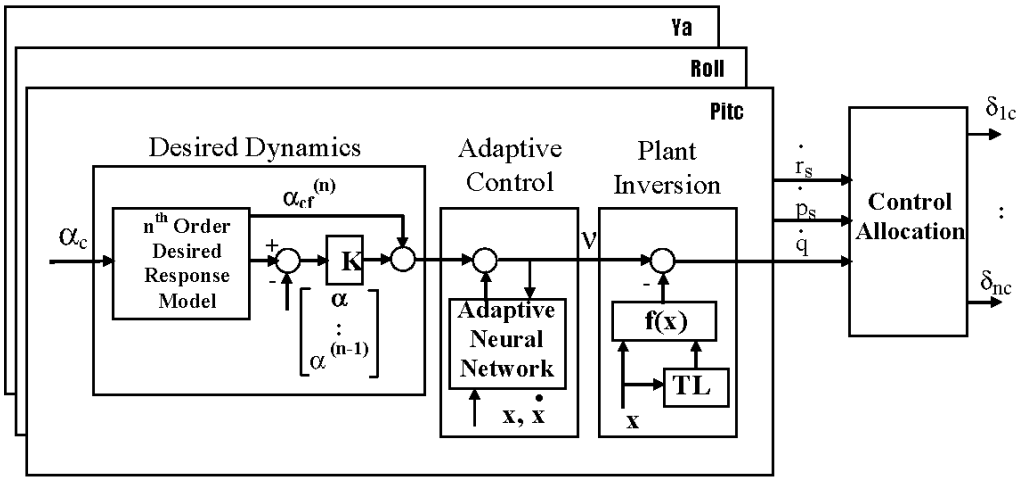


Figure 2-4 Direct adaptive explicit model following. Source [22]

It is worthy to mention the success of this program regardless of the short flight test period.

Similarly, from same authors, in [23] same control architecture is used but using on-line SI in order to change the model of control allocator, plant of the dynamic inversion and the reference model.

An independent approach using MMST is doing at [24] for the same aircraft X-36.

Other flight tests and approaches like the Intelligent Flight Control by NASA in 1994, explored a more sophisticated used of NN and the design evolved in something similar to an indirect adaptive control law using and LQR tuned on-line using parameters from a NN and on-line SI coupled with a NN to correct errors of the identification. This [25] approach was flight tested in an F-15 and Figure 2-5 shows the main architecture.

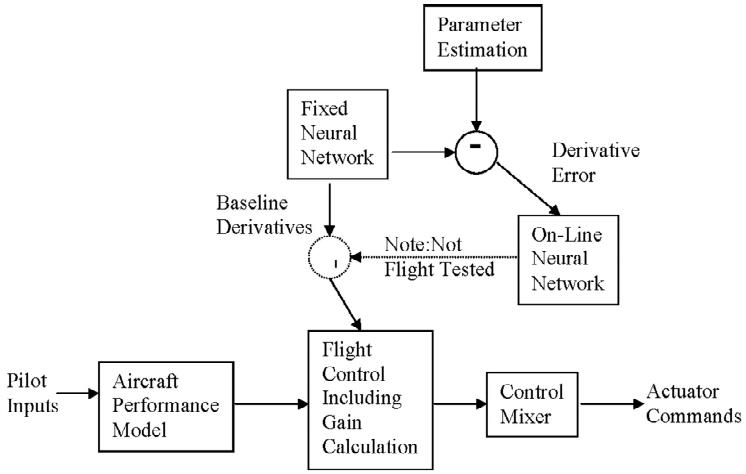


Figure 2-5 Indirect adaptive LQR through system ID and NN. Source [25]

Regardless the exit of this flight test, NASA move towards [23] concept.

Regarding CA at the end of the 90's, in [26] several techniques to perform CA and their integration with an adaptive scheme are examined.

2.4 F-16 examples in adaptive control

Many studies have relied their simulations and concepts tests in an F-16 model or even the real aircraft.

For example the programme SRFCS initially focused in F-16 for the designs of fault-tolerant autopilot.

In the 80's GE used a model of AFTI F-16 at [27] for simulation and testing of concepts. These concepts were based in dynamic inversion.

In 1991 an LQR architecture indirect adaptive control is looked at [11], which is based on a plant of F-16 and reached successful simulations without any FDI or fault identification.

At [28] the author explores a two-stage Kalman filter to deal with FDI and for reconfigurable control the approach is a direct adaptive approach using an on-line tuning of LQR controller taking into account the FDI. The simulations and testing of performance is carried out with a model of AFTI/F-16.

At [29] reinforcement learning (Adaptive Critic Design) is used along with an approximated plant dynamics of an F-16, showing how the system can learn from environment and adapt to it.

Different from control but still adaptive philosophy, a trajectory control based on NN and applied to a model of F-16 is done at [30]. Also an adaptive autopilot based on Lyapunov functions and different loops with a philosophy of direct adaptive MIAC. B-spline NN are used to constrain the control law.

2.5 Attractive concepts

The most interesting concepts, regarding the different faults there should be faced in this thesis, are the concepts based in [22] and [31].

These concepts present the following features that make them attractive:

- They have an actual reconfigurator or allocator that can achieve extreme control configurations and extreme damage or faults
- They used an reference model as desired dynamics to follow and this presents the advantage that once can design the performance of the controller
- They used of an on-line adaptation like NN, presents the advantage that there is no need to rely solely on an FDI system
- Most of them present an online SI that can alleviate big deviations

Among the techniques used on this publications and books, dynamic inversion and/or any design of reference model are used mostly to generate semi-control commands. Control allocator seems to be a fundamental piece of the concepts and on-line SI sets exact dynamics onto the previous mentioned to alleviate deviations.

Also NN are used to correct or follow a model either, explicit or implicit.

FDI seems not to appear like needed but there should be a fast FDI to further accommodate major changes of the plant into the dynamic inverter and/or control allocator, as well as the reference model, which should relax the commands for a faulty plant.

The most agreed and proven approach is presented in the book [31] where approaches, methods and techniques, similar to the presented at [22], are explained. That is, direct explicit or implicit model following.

2.5.1 Nonlinear Dynamic Inversion

Dynamic inversion control has been used extensively in the past years in flight control systems and more recently, nonlinear dynamic inversion has been explored in [31] , [32], [33] and [34] for example.

In [32] NDI is fully covered along with other similar techniques like Sliding Mode Control and Adaptive Robust Control, for instance. There the author explain the basics of the method and demonstrate its stability and stability conditions. Note, that in this publication the author performs pure NDI without the use of a control allocator.

The use of CA in conjunction with NDI simplifies substantially the problem. This approach is used in [31] where CA is used to perform a final inversion of the control deflection contributions on the rotational dynamics and obtain control deflections. This approach present the benefits of not rely solely on NDI and the allocation of control effort of CA.

A more advanced approach is used in [33] which performs a Polynomial Eigenstructure Assignment as approach to dynamic inversion applying it to a nonlinear model of missile and proving stability over its entire flight envelope.

NDI is also used for UAVs in [34] where a full autopilot is develop using Dynamic inversion as method to compute control deflections and MRAC to compute the pseudo control signals used by the DI. Note that the author used an Adaptive Neural Network to perform adaptation and a reference model to generate the adaptation error. This approach is less accurate than [31] but its performance was demonstrate in actual flight.

Using the same method than in [31] of doing NDI, that is, using a CA to perform a final inversion, the authors in [35] propose an Incremental Nonlinear Dynamic Inversion to simplify the inversion and get rid of the main term on it, so any effect of uncertainties on this term are removed. This proposal assume a discrete process and, using information from the previous step, estimate the increment on the virtual control commands that are need to achieve the pseudo control signals (angular accelerations). This increment can be passed then to a Control allocator to compute the control deflections or simply use a model to perform the inversion as stated in classical NDI, which is actually what they propose.

2.5.2 Control Allocation

For a long time many authors, like Anthony Calise in [22], used the idea of Control effort manager as the way to cope with redundancies and faults. These advantages have made CA to become a part almost essential in many Fault-Tolerant FCS.

CA is studied in [36], [31], [37], [38] , [39] and [40] where a set of different methods like pseudo inverse or dynamic CA among others are explained.

The most important publication taken into account is [39] where the author uses mainly dynamic programing (quadratic programing) to generate control deflections. Using this technique, the author defines some different algorithms like Sequential Least Squares (SLS), Weighted Least Squares (WLS), etc.

2.5.3 System Identification and Fault Detection and Isolation

Some of the attractive concepts like [22], in addition to perform adaptive control, also perform SI. For instance, in [22] a real time estimation of aerodynamic coefficients and propulsion forces is proposed using a least square algorithm.

In the Control Handbook [41] Lennart Ljung explains different techniques and methods to perform SI, but most of them are not based on a specific model as in [22], but in a general models like ARX, ARMAX or NN. He proposed as well, a least square minimization technique using different minimization methods like Gradient descent, Gauss-Newton, Levenbergh-Marquard or Conjugate Gradient. He also proposes on-line algorithms which generally represent the “optimal” algorithm: Kalman Filter Based SI.

More advanced SI methods can be found in [42] and [43] for instance. In [43] Dynamic NN are used for identification of nonlinear systems and several learning methods are proposed for online learning, that is, adaptation laws.

Regarding FDI, Ducard in [31] proposes a Multi Model Adaptive Estimation using Extended Kalman Filters, that is EMMAE, to perform a FDI using nonlinear models and avoiding an online linearization of the model. This innovative FDI method accurately predicts deflection angles and fault modes, but still rely on fault modes to design each individual EKF, so, computational power is quite big.

Other approach would be the proposed by authors in [44] and [45] where, they describe the called Unknown Input Decoupled Functional Observer (UIDFO) which uses linear

models of the plant. In [45] the authors are able to estimate stuck control surfaces positions with a good performance.

2.5.4 Neuro-adaptive control

There are many publications referring to adaptive and neuro-adaptive approaches like the already mentioned [22], [23] , [18] , [34] or [25] but later publications like [46] explore deeply the approach implementing it in a dual-propeller VTOL vehicle controller using explicit model following and dynamic inversion. Similarly, in [47] the same concept is used for missiles like in [18] . A similar approach is used in [48] in an automatic landing manoeuvre controller as in [49].

A discrete analysis of the neuro-adaptive method is performed in [50] where stability is demonstrated for a discrete approach, demonstrating the improved performance and stability of a switching scheme. In [51] an LMI based linear controllers like LQG or H_∞ are implemented along with an adaptive neural network using two approaches, bot different to the generally used in [22] or [23].

A good review of efforts on this field is done in [52] where different approaches are discussed, as well as in [53], which discuss all programs and efforts in the matter.

More advanced approaches are explained at [54], [55], [56], [29], [57], [58], [59], [60] and [61]. For instance, in [54] a discrete adaptive control for switched nonlinear systems is developed where a similar adaptation law as in [22] is used but demonstrating stability in the discrete domain. And, in [61] an adaptive recurrent neuro-fuzzy control, performing direct adaptive control with a PD controller, is applied to an AUV.

All these publications tackle the problem in different ways and perform the learning process using different techniques, but most of them use back-propagation of the error at the output and use tracking error, for instance, as adaptation error. Note that most of the adaptation laws are obtained from a Lyapunov stability analysis, so that the entire adaptation process is convergent. However, none of them except [50] and [54], demonstrate stability of the discrete learning process.

As a final comment, the authors on the Neural Control section of [41] explain a general procedure to design Adaptive NN in the context of control systems and describe some of the properties of them. As well, they design an adaptation law for a single layer NN.

2.6 Other approaches

Although some of the previous concepts are stills explored and improved today, some other new approaches and more expensive computationally have appeared lately.

In [32] NDI is used together with an adaptive robust control, which is described in [62]. Adaptive robust control is described, as said, at [62] and it combines techniques of adaptive control and deterministic robust control.

Other approach is the one using model Predictive Control or even Non-linear Model Predictive Control (NMPC). Reconfigurable fault-tolerant control using NMPC is used in [63] and [64] where the controller takes the faults and accommodates them. Although the results were promising, the still inconvenience of the needed of a perfect FDI and its need makes it unreliable in some way. And the expensive computational power makes it at least not suitable for today's computers.

Other case is the Rockwell Collins adaptive flight controller [65] in which a simple direct adaptive MRAC is implemented along with an adaptive guidance algorithm. The works successfully demonstrated effectiveness of the method by flight test of a model of F-18 where damage up to 60% wing area loss was performed successfully.

Other approaches use Sliding Mode Control concepts to perform an adaptive or fault-tolerant controller. At [66] Sliding Mode is used to accommodate faults in aircraft using a model of Boeing 747. In this approach a direct adaptive philosophy is used in conjunction with a Sliding Mode algorithm to generate control signals, which redistribute control signals to overcome faults.

Other approach using SMC is [67] which uses a SMC and control allocator in an inner loop to reach the commanded inputs. The control allocator uses a nominal model of plant, so no adaptation is included there, but the adaptation is on the SMC part that does not even use any FDI or parameter identification. Instead, SMC uses a sliding surface boundary reconfiguration (direct adaptation) to overcome effectors fault or deviations.

A more complete approach that also considers guidance for UAVs is [34] where a direct explicit model following approach is used. The method for the inner control is based in a reference model or desired dynamics, a NDI (called Approximate Model Inversion MRAC) an adaptation through a neural network with a PD controller. Also, the external module of guidance takes into account the faulty behaviour to reconfigure its commands.

Other approach like [68] uses Linear Matrix Inequality (LMI) and adaptive methods for the problem of tracking under flight faults. Eventual faults estimations and addition of a new controller in that case, fulfils the objective without the used if FDI, That is, it uses a direct adaptive approach but without any reference model, just fault estimation. Also an F-16 simulation of the controller is performed in this work.

2.7 F-16 models literature

As said, the theory and algorithms developed in this thesis were tested and proven with an F-16 model. In fact, some of the algorithms require certain information about the model to be controlled as it is the NDI. So, an F-16 model had to be developed.

The basic model was found to be developed by Richard Russell in [69] who present sand implements a model of F-16 in C code from official reports of NASA of wind tunnel test at NASA Ames and Langley Research centres. This implementation in C is also used in Matlab/Simulink crating a complete simulation framework. Sonneeldt explain in a better way the same model in [70], where the model is detailed explained.

The aerodynamic model used in both publications is implemented in many lookup tables from data of official NASA technical reports, as said. The same aerodynamic data is implemented in an analytical manner by Morelli at NASA Langley Research centre in [71] using multi polynomial fittings of the aerodynamic coefficients and a least square method.

Data of F-16 design were found in the NASA technical report [72], which was useful when modifying the model.

Modification of the already mentioned 3dof models was necessary to achieve a 7dof model. This modification was mainly derived in this work.

A similar modification can be found in [73] where a modification of a simplified 3dof model is performed to get a 5dof model: left/right aileron, left/right elevator and rudder. This publication was found to be very useful when performing the modification in this work, although sign and unit inconsistencies (some introduced terms have units in dimensionless expressions) were found in the modification of the publication mentioned. So, the careful reading and understanding of the mentioned publication is encouraged.

3 Non-linear Aircraft model

The control systems proposed in this work are developed using an F-16 model and, therefore, tested using the same model. In this section a nonlinear model of F-16 is developed configuring a base model. This model is mainly based on [70] and [69], since the aerodynamic properties are obtained from these publications. Also, the book [74] was useful when deriving some equations.

Further modifications of the base model are performed as well to achieve a 7 degrees-of-freedom model and a reconfigurable model. This reconfigurable model includes enough parameters for representing the failures and deviations stated in section 1.

3.1 3dof baseline description

The first model to describe is a full 6dof model with 3dof of control. Note that the basic classic mechanics derivation is valid for the rest of the models developed in this section, since the changes are performed in the aerodynamic model.

3.1.1 Hypotheses and definitions

The first step is to define the assumptions made and the reference frames used.

Firstly, the assumptions assumed for this model are:

- The aircraft is a rigid body.
- The inertial reference assumed is a local frame on the surface of the Earth, hence the earth curvature is neglected as well as its rotational effects.
- Mass variations are not considered as part of the dynamic model
- Mass distribution of the aircraft is symmetric about the plane $X_b O Z_b$
- Steady atmosphere: Non wind disturbances
- International Standard Atmosphere is used

Secondly, the main frames are defined (see Figure 3-1):

- Inertial reference F_e with the origin at the initial nadir of the aircraft with the x-axis pointing to the centre of the earth and the x-axis to the North
- F_h is the reference frame with origin at CG, z-axis vertical with the earth (Radial to it), and x-axis pointing to the North. That is, parallel to F_e but with origin at CG of the aircraft
- F_b is the reference frame with origin at CG z-axis on the symmetry plane of the aircraft and x-axis pointing in the direction that makes $I_{xy} = 0$ or, normally, with the datum of the airplane.

- F_w is the reference frame with origin at CG, x-axis pointing in the opposite direction of $\vec{V}_{wind}$ and z-axis on the symmetry plane of the aircraft.

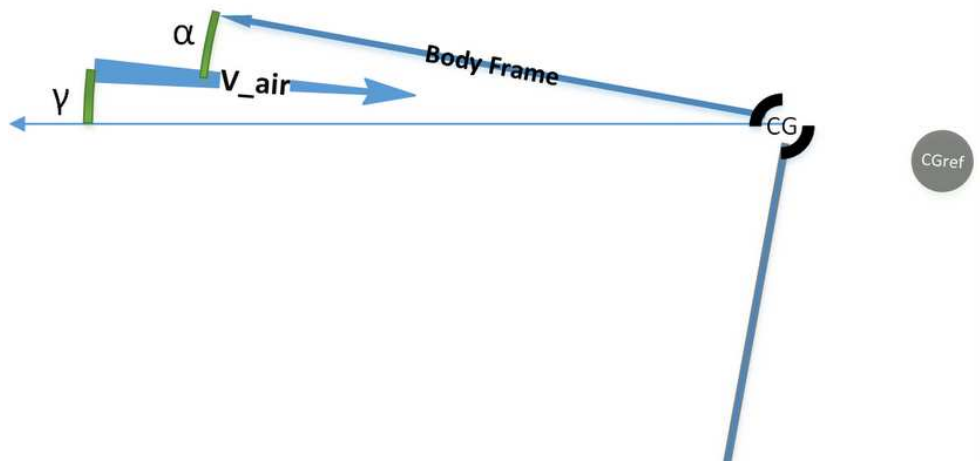


Figure 3-1 CGref- CG and Frames arrangement

In Figure 3-1 it can be seen how the body axes are placed at the CG and that there exist a reference CG (CG_{ref}). This reference CG is where aerodynamic centre is placed, that is, where the aerodynamic forces are applied.

Mass properties and geometric data of the baseline model can be found at Table 10-1 at Appendix A.1.

With these definitions of reference frames, the following transformations can be defined.

The first rotation is body axes-wind axes. This rotation is defined by the incidence angles, which are the angles which the air incises to the aircraft with:

$$F_b \underset{\alpha,\beta}{\longleftrightarrow} F_w$$

$$T_{F_b \to F_w} = \begin{pmatrix} c\alpha c\beta & s\beta & s\alpha c\beta \\ -s\beta c\alpha & c\beta & -s\alpha s\beta \\ -s\alpha & 0 & c\alpha \end{pmatrix}$$

$$\vec{V}_{F_h} = T_{F_b \to F_h} \cdot \vec{V}_{F_b}$$

Where c and s represent $cosine$ and $sine$ respectively and $T_{F_b \to F_w} = \left(T_{F_w \to F_b}\right)^T$. Note that $\vec{V}_{F_w} = \begin{pmatrix} -V_{TAS} \\ 0 \\ 0 \end{pmatrix}_{F_w}$ if no wind is considered.

The next rotation is between local horizon and body axes:

$$F_b \xleftrightarrow[\varphi,\theta,\psi]{} F_h \ \ (Euler\ angles)$$

This last rotation is defined by Euler angles, and in this case the order of rotation is Yaw-Pitch-Roll (or z, y, x or 3,2,1) with the rotational matrix:

$$T_{F_h \to F_b} = \begin{pmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\varphi c\theta c\psi - c\varphi s\psi & s\varphi s\theta s\psi + c\varphi c\psi & s\varphi c\theta \\ c\varphi s\theta c\psi + s\varphi s\psi & c\varphi s\theta s\psi - s\varphi c\psi & c\varphi c\theta \end{pmatrix}$$

Where $T_{F_b \to F_h} = \left(T_{F_h \to F_b}\right)^T$ and $\vec{V}_{F_h} = T_{F_b \to F_h} \cdot \vec{V}_{F_b}$. Note that $\vec{g}_{F_h} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}_{F_h}$

An alternative representation of this rotation can be done with quaternions defining the quaternion with the Euler angles:

$$q = \begin{pmatrix} cos\left(\frac{\psi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\phi}{2}\right) \\ \cos\left(\frac{\psi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\phi}{2}\right) - \sin\left(\frac{\psi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\phi}{2}\right) \\ \cos\left(\frac{\psi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\phi}{2}\right) \\ \sin\left(\frac{\psi}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\phi}{2}\right) - \cos\left(\frac{\psi}{2}\right) \cdot \sin\left(\frac{\theta}{2}\right) \cdot \sin\left(\frac{\phi}{2}\right) \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix}$$

Or defining the Euler angles with the quaternion components:

$$\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} = \begin{pmatrix} atan2(2 \cdot q_3 \cdot q_4 + 2 \cdot q_1 \cdot q_2, 2 \cdot q_1^2 + 2 \cdot q_4^2 - 1) \\ -\sin^{-1}(2 \cdot q_2 \cdot q_4 - 2 \cdot q_1 \cdot q_3) \\ atan2(2 \cdot q_2 \cdot q_3 + 2 \cdot q_1 \cdot q_4, 2 \cdot q_1^2 + 2 \cdot q_2^2 - 1) \end{pmatrix}$$

Achieving the axes transformation or rotation with:

$$\begin{pmatrix} 0 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix}_{F_h} = q \otimes \begin{pmatrix} 0 \\ v_1 \\ v_2 \\ v_3 \end{pmatrix}_{F_b} \otimes q^*$$

Where $\otimes$ is the Hamiltonian product (see [75]) and $q = {}^{F_b}_{F_h}\hat{q}$ $y\ q^* = {}^{F_h}_{F_b}\hat{q}$ as defined in [75]. Note that the conjugate quaternion $q^* = \begin{pmatrix} q_1 \\ -q_2 \\ -q_3 \\ -q_4 \end{pmatrix}$.

Once reference frames are defined, the kinematics is described next. F_b is chosen for working, so, as far as possible, all the variables are referenced to that frame and to CG. The velocity of CG:

$$\vec{V} = \vec{V}_{CG} = \begin{pmatrix} u \\ v \\ w \end{pmatrix}_{F_b}$$

So, this velocity is the velocity respecting the inertial reference. The air velocity:

$$\vec{V}_{air} = \vec{V}_{ground} + \vec{V}_{wind}$$

Where $\vec{V} = \vec{V}_{ground} = Inertial\ velocity$

Since, as said, in this models wind is not considered: $\vec{V}_{air} = \vec{V}_{ground}$ and $V_{TAS} = \|\vec{V}_{air}\| = \|\vec{V}\|$.

With these assumptions, wind incidence angles can be computed as function of velocity components in body axes:

$$V_{TAS} = \|\vec{V}\| = \sqrt{u^2 + v^2 + w^2}$$

$$\alpha = \arctan\left(\frac{w}{u}\right)$$

$$\beta = \arcsin\left(\frac{v}{V_{TAS}}\right)$$

Note also that the dynamic pressure is: $\bar{q} = \frac{1}{2} \cdot \rho \cdot V_{TAS}^2 = \frac{1}{2} \cdot \rho_0 \cdot V_{EAS}^2$

And, defining the path angle γ for the case $\beta = \phi = 0$ as: $\gamma = \theta - \alpha$

And the angular rate of the body $\vec{\omega}_b = \begin{pmatrix} p \\ q \\ r \end{pmatrix}_{F_b}$, which is the angular rate of the body axes respecting F_h. So, the relationship between body angular rates and Euler angles rates:

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix}_{F_b} = \begin{pmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi \end{pmatrix} \cdot \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}$$

3.1.2 Translational dynamics

Using the second law of Newton with the notation and definitions already done:

$$\sum \vec{F}_{ext} = \frac{d}{dt}(m \cdot \vec{V}_{CG}) \tag{3-1}$$

Being $\vec{F}$ the total forces applied to the body and $m \cdot \vec{V}_{CG}$ the momentum of the body represented by the total mass and the velocity of the CG.

The second terms of the equation (3-1) is obtained as follows:

$$\frac{d}{dt}(m \cdot \vec{V}_{CG}) = \dot{m} \cdot \vec{V}_{CG} + m \cdot \frac{d}{dt}(\vec{V}_{CG}) \quad with\, \dot{m} = 0$$

$$\frac{d}{dt}(\vec{V}_{CG}) = \left(\frac{d}{dt}(\vec{V})\right)\Big|_{F_e} = \vec{\dot{V}}\Big|_{F_b} + \vec{\omega}_b \times \vec{V}$$

Being $\vec{\dot{V}} = \begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix}_{F_b}$. With that the translational equations is written:

$$\sum \vec{F}_{ext} = m \cdot \left(\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix}_{F_b} + \vec{\omega}_b \times \vec{V} \right)$$

Or, alternatively:

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix}_{F_b} = \frac{\sum \vec{F}_{ext}}{m} - \vec{\omega}_b \times \vec{V} \tag{3-2}$$

Discussing now the first term of equation (3-1), the external forces have different contributions:

$$\sum \vec{F}_{ext} = \vec{F}_{Gravity} + \vec{F}_{Aerodynamics} + \vec{F}_{Ground} + \vec{F}_{Propulsion} = \begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix}_{F_b}$$

For this case, Ground forces are neglected since only flight case is considered.

Defining now aerodynamic forces as $\vec{F}_{Aerodynamics} = \begin{pmatrix} \bar{X} \\ \bar{Y} \\ \bar{Z} \end{pmatrix}_{F_b}$, propulsion forces as $\vec{F}_{Propulsion} = \begin{pmatrix} F_T \\ 0 \\ 0 \end{pmatrix}_{F_b}$ where F_T is the thrust force produced by the engine.

Gravity force, or weight: $\vec{F}_{Gravity} = \begin{pmatrix} W_x \\ \mathrm{W_y} \\ \mathrm{W_z} \end{pmatrix}_{F_b} = m \cdot T_{F_h \to F_b} \cdot \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}_{F_h}$.

The aerodynamics forces are defined using an aerodynamic model, expressed in body axes. With that the aerodynamic forces in body axes are obtained:

$$\begin{pmatrix} \bar{X} \\ \bar{Y} \\ \bar{Z} \end{pmatrix}_{F_b} = \bar{q} \cdot S \cdot \begin{pmatrix} \mathrm{C_{X_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \\ \mathrm{C_{Y_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \\ \mathrm{C_{Z_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \end{pmatrix} \quad \textbf{(3-3)}$$

Where δ_i are the deflections of elevators, ailerons and rudder.

Force coefficients$\mathrm{C_{X_T}}$, $\mathrm{C_{Y_T}}$ and $\mathrm{C_{Z_T}}$ are given by the aerodynamic model in section 3.1.7.

With that, using equations (3-3), the final formulation of the translational dynamics is:

$$\begin{aligned} \bar{X} + F_T - m \cdot g \cdot \sin(\theta) &= m \cdot (\dot{u} + q \cdot w - r \cdot v) \\ \bar{Y} + m \cdot g \cdot \sin(\phi) \cdot \cos(\theta) &= m \cdot (\dot{v} + r \cdot u - p \cdot w) \\ \bar{Z} + m \cdot g \cdot \cos(\phi) \cdot \sin(\theta) &= m \cdot (\dot{w} + p \cdot v - q \cdot u) \end{aligned} \quad \textbf{(3-4)}$$

3.1.3 Rotational dynamics

For the rotational dynamics, angular momentum has to be defined using the inertia matrix in body axes and, since the CG is the body origin, about the CG.

$$\vec{\mathcal{L}} = I_{CG} \cdot \vec{\omega}_b + \vec{\mathcal{L}}_{eng}$$

Where $\vec{\mathcal{L}}_{eng} = \begin{pmatrix} h_{eng} \\ 0 \\ 0 \end{pmatrix}_{F_B}$ is a constant approximation of the angular momentum of the engine. See Table 10-1.

It is assumed that I_{CG} is known or can be obtained and is expressed in F_b and about the CG. So, the angular momentum $\vec{\mathcal{L}}$ is generally expressed in F_b. In fact, Table 10-1 presents the moments of inertia of the baseline model.

$$I_{CG} = \begin{pmatrix} I_x & I_{xy} = 0 & I_{xz} \\ I_{xy} = 0 & I_y & I_{yz} = 0 \\ I_{xz} & I_{yz} = 0 & I_z \end{pmatrix}$$

Applying the same Newton's law but to the rotational dynamics:

$$\sum \vec{M}_{ext} = \frac{d}{dt}\left(\vec{\mathcal{L}} + \vec{\mathcal{L}}_{eng}\right) \quad \textbf{(3-5)}$$

Where the second term is the derivative of the angular momentum respecting the inertial frame:

$$\frac{d}{dt}\left(\vec{\mathcal{L}}\right)\Big|_{F_e} = \frac{d}{dt}\left(\vec{\mathcal{L}}\right)\Big|_{F_b} + \vec{\omega}_b \times \vec{\mathcal{L}} + \vec{\omega}_b \times \vec{\mathcal{L}}_{eng} = I_{CG} \cdot \dot{\vec{\omega}}_b + \vec{\omega}_b \times (I_{CG} \cdot \vec{\omega}_b) + \vec{\omega}_b \times \vec{\mathcal{L}}_{eng}$$

So:

$$\sum \vec{M}_{ext} = I_{CG} \cdot \dot{\vec{\omega}}_b + \vec{\omega}_b \times (I_{CG} \cdot \vec{\omega}_b) + \vec{\omega}_b \times \vec{\mathcal{L}}_{eng}$$

Where $\dot{\vec{\omega}}_b = \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{F_b}$

Clearing the angular accelerations:

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{F_b} = (I_{CG})^{-1} \cdot \left(\sum \vec{M}_{ext} - \vec{\omega}_b \times (I_{CG} \cdot \vec{\omega}_b) - \vec{\omega}_b \times \vec{\mathcal{L}}_{eng}\right) \quad \textbf{(3-6)}$$

Defining now the first term of equation (3-5) :

$$\sum \vec{M}_{ext} = \vec{M}_{Aero} + \vec{M}_{Prop} + \vec{M}_{GroundForces} = \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix}_{F_b}$$

Where $\vec{M}_{GroundForces}$ are neglected since only flight conditions are considered.

$\vec{M}_{Aero}$ is given by the aerodynamic model in section through moments coefficients 3.1.7. This aerodynamic model already takes into account moments of forces due to the fact

that forces are applied at the Aerodynamics Centre (CG_{ref}) and the rotational equations are formulated at CG.

$$\vec{M}_{Aero} = \begin{pmatrix} \bar{L} \\ \bar{M} \\ \bar{N} \end{pmatrix}_{F_b} = \bar{q} \cdot S \cdot \begin{pmatrix} b \cdot \mathrm{C}_{\mathrm{l_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \\ \bar{\mathrm{c}} \cdot \mathrm{C}_{\mathrm{m_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \\ b \cdot \mathrm{C}_{\mathrm{n_T}}(\alpha, \beta, p, q, r, \delta_i, \dots) \end{pmatrix} \quad \textbf{(3-7)}$$

Due to the engine position, the moments of the thrust are:

$$\vec{M}_{Prop} = \begin{pmatrix} L_T \\ M_T \\ N_T \end{pmatrix}_{F_b} = \begin{pmatrix} 0 \\ F_T \cdot z_t \\ 0 \end{pmatrix}$$

3.1.4 Final formulation of motion

Once the differential equations are defined as functions of the dynamics and forces and moments, the entire dynamics is described here.

For the translational dynamics:

$$\vec{V} = \vec{V}_b = \int_{t_0}^{t} \begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix}_{F_b} \cdot dt + \vec{V}_0$$

Where $\vec{V}_b$ specifies the velocity in body axes F_b.

$$\begin{pmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{pmatrix}_{F_b} = \begin{pmatrix} r \cdot v - q \cdot w - g \cdot \sin(\theta) + \frac{1}{m} \cdot (\bar{X} + F_T) \\ p \cdot w - r \cdot u + g \cdot \sin(\theta) \cdot \cos(\theta) + \frac{1}{m} \cdot \bar{Y} \\ q \cdot u - p \cdot v + g \cdot \cos(\phi) \cdot \cos(\theta) + \frac{1}{m} \cdot \bar{Z} \end{pmatrix} \quad \textbf{(3-8)}$$

But for obtaining the position in the inertial frame $\vec{X}_e$, the velocity $\vec{V}_e$ is obtained:

$$\vec{V}_e = T_{F_b \to F_h} \cdot \vec{V}_b$$

So:

$$\vec{X}_e = \int_{t_0}^{t} \vec{V}_e \cdot dt + \vec{X}_0$$

And with that, all the translational dynamics is described. Now, the rotational dynamics is completed with the angular rates in body axes and rotation angles rates definition:

$$\vec{\omega}_b = \begin{pmatrix} p \\ q \\ r \end{pmatrix}_{F_b} = \int_{t_0}^{t} \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{F_b} \cdot dt + \vec{\omega}_{b\,0}$$

Where $\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{F_b}$ is given by equation (3-6). The Euler angles derivatives:

$$\begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \sin(\varphi) \cdot \tan(\theta) & \cos(\varphi) \cdot \tan(\theta) \\ 0 & \cos(\varphi) & -\sin(\varphi) \\ 0 & \dfrac{\sin(\varphi)}{\cos(\theta)} & \dfrac{\sin(\varphi)}{\cos(\theta)} \end{pmatrix} \cdot \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

And finally the attitude is given by:

$$\begin{pmatrix} \varphi \\ \theta \\ \psi \end{pmatrix} = \int_{t_0}^{t} \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \cdot dt + \begin{pmatrix} \varphi_0 \\ \theta_0 \\ \psi_0 \end{pmatrix}$$

That defines as well dynamically the rotation matrices $T_{F_b \to F_h}$ and $T_{F_h \to F_b}$

Alternatively, rotations can be performed with quaternions, where:

$$\begin{pmatrix} 0 \\ \vec{V}_e \end{pmatrix} = q \otimes \begin{pmatrix} 0 \\ \vec{V}_b \end{pmatrix} \otimes q^*$$

And the quaternion derivative:

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \frac{1}{2} \cdot q \otimes \begin{pmatrix} 0 \\ p \\ q \\ r \end{pmatrix}$$

If, wind-axes are wanted to be used as reference frame:

$$V_{TAS} = \sqrt{u^2 + v^2 + w^2}$$

$$\alpha = \arctan\left(\frac{w}{u}\right) \qquad \leftrightarrow \qquad \begin{aligned} u &= V_{TAS} \cdot \cos(\alpha) \cdot \cos(\beta) \\ v &= V_{TAS} \cdot \sin(\alpha) \\ w &= V_{TAS} \cdot \sin(\alpha) \cdot \cos(\beta) \end{aligned} \qquad \textbf{(3-9)}$$

$$\beta = \arcsin\left(\frac{v}{V_{TAS}}\right)$$

The derivatives are then:

$$\dot{V}_{TAS} = \frac{u \cdot \dot{u} + v \cdot \dot{v} + w \cdot \dot{w}}{V_{TAS}}$$

$$\dot{\alpha} = \frac{u \cdot \dot{w} - w \cdot \dot{u}}{u^2 + w^2} \quad and \quad \dot{\beta} = \frac{\dot{v} \cdot V_{TAS} - v \cdot \dot{V}_{TAS}}{V_{TAS}^2 \cdot \cos(\beta)}$$

Using this last equation and equation (3-8):

$$\dot{V}_{TAS} = \frac{1}{m} \cdot (-D + F_T \cdot \cos(\alpha) \cdot \cos(\beta) + m \cdot g_1)$$

$$\dot{\alpha} = q - (p \cdot \cos(\alpha) + r \cdot \sin(\alpha)) \cdot \tan(\beta) + \frac{1}{m \cdot V_{TAS} \cdot \cos(\beta)} \cdot (-L - F_T \cdot \sin(\alpha) + m \cdot g_3) \tag{3-10}$$

$$\dot{\beta} = \mathrm{p} \cdot \sin(\alpha) - \mathrm{r} \cdot \cos(\alpha) + \frac{1}{m \cdot V_{TAS}} \cdot (Y - F_T \cdot \cos(\alpha) \cdot \sin(\beta) + m \cdot g_2)$$

Where:

$$\begin{pmatrix} -D \\ Y \\ -L \end{pmatrix}_{F_W} = T_{F_b \to F_w} \cdot \begin{pmatrix} \overline{X} \\ \overline{Y} \\ \overline{Z} \end{pmatrix}_{F_B}$$

And: $\begin{pmatrix} g_1 \\ g_2 \\ g_3 \end{pmatrix}_{F_w} = T_{F_b \to F_w} \cdot \left(T_{F_h \to F_b} \cdot \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}_{F_h} \right)$

3.1.5 Control surfaces

This model has as control inputs the throttle (need to define engine model), elevator, ailerons and rudder. Note that deflection is defined as conventionally: positive elevator gives negative pitch rate, etc.

F-16 also has Leading Edge Flaps (LEFs) which are used to increase performance at high AoA. Deflection of LEF δ_{LEF} are given positively for a downwards deflection. The same aircraft controls this deflection automatically, which is given by the following equation in the Laplace space:

$$\delta_{LEF}(^\circ) = 1.38 \cdot \frac{2 \cdot s + 7.25}{s + 7.25} \cdot \alpha - 9.05 \cdot \frac{\bar{q}}{P} + 1.45 \quad with\ P = static\ Pressure \tag{3-11}$$

Note that later modifications of the model and usage of LEF as control surfaces assumes that the aircraft already deflect LEF in the described way in equation (3-11), so the control variables of LEF are added to the described δ_{LEF} in equation (3-11).

The behaviours of the control surfaces are assumed as 1st order systems and with the following characteristics:

Control surface	Units	Min	Max	Rate limit (deg/s)	Time constant (s)	Symbol
Elevators	deg	-25	25	± 60	0.0495	δ_e
Ailerons	deg	-21.5	21.5	± 80	0.0495	δ_a
Rudder	deg	-30	30	± 120	0.0495	δ_r
LEF	deg	0	25	± 25	0.136	δ_{LEF}

Table 3-1 Control surfaces characteristics

3.1.6 Engine model

F-16 is powered by a turbofan with afterburner engine. The model of engine used is explained at [70] and at appendix A.2.

The commanded variable is the throttle $\delta_{th} = [0 - 1]$. The steady level of power is P_c and is defined as:

$$P_c(\delta_{th}) = \begin{cases} 64.94 \cdot \delta_{th} & if \ \ \delta_{th} \leq 0.77 \\ 2017.38 \cdot \delta_{th} - 117.38 & if \ \ \delta_{th} > 0.77 \end{cases}$$

Note that 0.77 is the throttle deflection at which afterburner is activated.

For most of the models used in NDI, SI and CA, it is considered that for a given a throttle the engine is in a stable regime, so that the level of power is P_c and not P_a as define in the dynamic model in appendix A.2.

The thrust generated (in Newtons) is then given by the actual level of power P_a:

$$F_T = \begin{cases} T_{idle} + (T_{mil} - T_{idle}) \cdot \left(\frac{P_a}{50}\right) & if \ P_a < 50 \\ T_{mil} + (T_{max} - T_{mil}) \cdot \left(\frac{P_a}{50} - 1\right) & if \ P_a \geq 50 \end{cases}$$

3.1.7 Aerodynamic model

There exist two aerodynamic models presented by Russell at [69] and Sonneveldt [70]: a low-fidelity model and a high-fidelity model.

The high-fidelity model data has been obtained from wind tunnel tests at NASA Ames and Langley Research Centres. This model can be found in [76] and data is valid for the following flight envelope:

- $-20 \leq \alpha \leq 90\ degrees$
- $-30 \leq \alpha \leq 30\ degrees$
- $0.1 \leq M \leq 0.6$

Note how this model includes stall and deep stall data. Note also that, the model can be flight at a Mach number higher than 0.6, although its reality is unknown.

The low-fidelity model can be found at [77] and present the next flight envelope:

- $-10 \leq \alpha \leq 45 degrees$
- $-30 \leq \alpha \leq 30\ degrees$
- $0.1 \leq M \leq 0.6$

This low fidelity model does not use LEF neither contains some crossed aerodynamic effects.

Although both models can be selected on the files provided by Russell [69] and the software developed, high fidelity model has been always chosen to develop all algorithms and perform all simulations.

3.1.7.1 Aerodynamic coefficients

Forces and moment coefficients used in sections 3.1.2 and 3.1.3 to generate their respective forces and moments, are total coefficients, that is, they are the sum of several contributions. In fact, the aerodynamic model is very complex and has many cross terms that represents cross influences. For instance, influence of LEF deflection in aileron effectiveness.

Due to the complexity of the model, this is explained in appendix A.3 while here only a simplified version is exposed. It is possible, by simple comparison of these two, to identify the correspondent expression of every derivative here presented.

Note that the coefficients are generally function of:

$$\{C_{X_T}, C_{Y_T}, C_{Z_T}, C_{l_T}, C_{m_T}, C_{n_T}\} = f(\alpha, \beta, p, q, r, \delta_e, \delta_a, \delta_r, \delta_{LEF})$$

And even, most of the derivatives in the following equations are functions of $(\alpha, \beta, p, q, r, \delta_e, \delta_a, \delta_r, \delta_{LEF})$.

Note that here the deflections used are different and that there exist the following conversions between those used in appendix A.3 (actual deflections) and here:

$$\begin{array}{ccc} \delta_e & \Longrightarrow & \delta_e \\ \delta_a & \Longrightarrow & \delta'_a = \dfrac{\delta_a}{20} \\ \delta_{LEF} & \Longrightarrow & \delta'_{LEF} = \left(1 - \dfrac{\delta_{LEF}}{25}\right) \\ \delta_r & & \delta'_r = \dfrac{\delta_r}{30} \end{array} \tag{3-12}$$

So, the <u>simplified expressions</u> of total coefficients are:

$$C_{X_T} = C_X(\alpha, \beta, \delta_e) + C_{X_{LEF}} \cdot \delta'_{LEF} + C_{X_q} \cdot q$$

$$C_{Y_T} = C_Y(\alpha, \beta) + C_{Y_{LEF}} \cdot \delta'_{LEF} + C_{Y_{\delta_a}} \cdot \delta'_a + C_{Y_{\delta_r}} \cdot \delta'_r + C_{Y_r} \cdot r + C_{Y_p} \cdot p$$

$$C_{Z_T} = C_Z(\alpha, \beta, \delta_e) + C_{Z_{LEF}} \cdot \delta'_{LEF} + C_{Z_q} \cdot q$$

$$C_{l_T} = C_l(\alpha, \beta, \delta_e) + C_{l_{LEF}} \cdot \delta'_{LEF} + C_{l_{\delta_a}} \cdot \delta'_a + C_{l_{\delta_r}} \cdot \delta'_r + C_{l_r} \cdot r + C_{l_p} \cdot p + C_{l_\beta} \cdot \beta \tag{3-13}$$

$$C_{m_T} = C_m(\alpha, \beta, \delta_e) + C_{Z_T} \cdot \left(x_{cg_r} - x_{cg}\right) + C_{m_{LEF}} \cdot \delta'_{LEF} + C_{m_q} \cdot q + \Delta C_m(\alpha, \delta_e) + \Delta C_{m_{ds}}(\alpha, \delta_e)$$

$$C_{n_T} = C_n(\alpha, \beta, \delta_e) + C_{n_{LEF}} \cdot \delta'_{LEF} + C_{n_{\delta_a}} \cdot \delta'_a + C_{n_{\delta_r}} \cdot \delta'_r + C_{n_r} \cdot r + C_{n_p} \cdot p + \Delta C_{n_\beta}(\alpha) \cdot \beta - C_{Y_T} \cdot \left(x_{cg_r} - x_{cg}\right) \cdot \frac{\bar{c}}{b}$$

It is encouraged to identify by comparison the partial derivatives in equation (3-13) with the actual expressions in appendix A.3. With this, it can be noticed that this aerodynamic model is very non-lineal and contain many effects like deep-stall ($\Delta C_{m_{ds}}(\alpha, \delta_e)$).

Through the mentioned identification of partial derivatives, it can also be noticed that many of these partial derivatives are usually a function like $f(\alpha, \beta, p, q, r, \delta_i, \dots)$. This is very important to understand when facing next sections.

Note that for further modifications, notation and derivatives presented in this section is used.

An example of the data available on the model by [69] can be seen in Figure 3-2, where lift coefficient is presented against AoA and Sideslip:

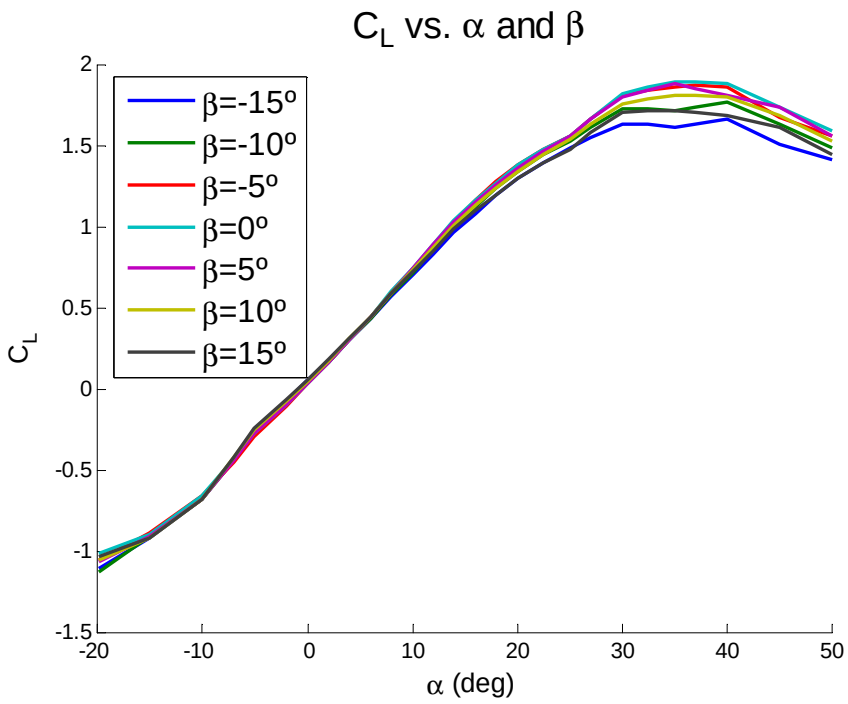


Figure 3-2 C_L curve Model

3.2 7dof modification

The objective of this modification is to obtain a model of the F-16 in all its 6 degrees-of-freedom functions of independent deflections of each of its control surface (7dof of control): left/tight elevator, left/right aileron, rudder, left/right LEF. Note that usually this model is called asymmetric model, due to the asymmetry of deflections.

The purpose of this modification is to provide to the control system and the aircraft with a higher redundancy and control capabilities. So, for example, the aircraft can fly in non-conventional ways. This is essential when actuators faults or structural damage occurs.

As mentioned before, since the purpose of the LEF is to improve performance at high AoA and this behaviour is internally computed, LEF deflections as control surfaces are considered as additional, they are added, to the internally computed deflections. Note that absolute limits of these actuators were taken into account.

Deflections taken into account now are:

Surface	Symbol	Positive deflection (its effect)
Left elevator	δ_{el}	Downwards (Negative Pitching-moment)
Right elevator	δ_{er}	Downwards (Negative Pitching-moment)
Left Aileron	δ_{al}	Upwards (Negative Rolling-moment)
Right Aileron	δ_{ar}	Upwards (Positive Rolling -moment)
Rudder	δ_{r}	Leftwards (Negative Yawing-moment)
Left LEF	δ_{LEF_l}	Downwards (Positive Rolling -moment)
Right LEF	δ_{LEF_r}	Downwards (Negative Rolling -moment)

Table 3-2 7dof model control surfaces

In order to design this modification the simplified model show in equations (3-13) is used and, hence, it's simplified notation and control variables. That is, here the conversions between control variables shown in equation (3-12) are also used, so that the control variables used here are:

$$Control\ variables = \{\delta_{el}, \delta_{er}, \delta'_{al}, \delta'_{ar}, \delta'_{r}, \delta'_{LEF_l}, \delta'_{LEF_r}\}$$

Once using these variables, the effects of these deflections can be classified into two categories: symmetric and asymmetric deflections. That is, a symmetric deflection of LEF is totally different to an asymmetric deflection of these surfaces. So, inside the aerodynamic model, symmetric and asymmetric deflections have different contributions and can be separated.

Due to this fact, four new variables are introduced in order to simplify the notation:

Symmetric deflections	Asymmetric deflections
$\delta_a^S = \frac{\delta'_{al} + \delta'_{ar}}{2}$	$\delta_a^A = \frac{\delta'_{al} - \delta'_{ar}}{2}$
$\delta_{LEF}^S = \frac{\delta'_{LEF_l} + \delta'_{LEF_r}}{2}$	$\delta_{LEF}^S = \frac{\delta'_{LEF_l} - \delta'_{LEF_r}}{2}$

Table 3-3 Symmetric/Asymmetric deflections

Note that due to the nature of the contributions to the total coefficients of the elevator deflections (non-linear), the conversion in Table 3-3 cannot be performed. In contrast, symmetric and asymmetric contributions are performed with the derivatives.

There are two main steps to carry out this modification:

- Identify and compute the correspondent symmetric or asymmetric contribution of each control surface already in the model and transform it into separate contributions proportional to each control surface area. For instance, elevators in C_Z: $C_Z(\alpha,\beta,\delta_e) \Rightarrow \frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{el}) + \frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{er})$
- Identify and compute crossed terms that appears due to non-conventional deflections of the control surfaces. For instance, rolling moment due to asymmetric elevators deflection: $C_{l_{cross}} = -\frac{l_e}{b}\cdot\left(\frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{el}) - \frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{er})\right)$

Due to the new terms, cross terms mainly, there are some dimensions to be introduced. These distances, in Table 10-1, are distances from symmetry plane to the correspondent Aerodynamic Centre of the control surface. Note that these distances where found in [73] and checked for geometric consistency: l_e, l_a, l_{LEF}

Finally, the asymmetric model is:

Force coefficient in x-axis:

$$\begin{aligned} C_{X_T} &= \frac{1}{2}\cdot C_X(\alpha,\beta,\delta_{el}) + \frac{1}{2}\cdot C_X(\alpha,\beta,\delta_{er}) + C_{X_{LEF}}\cdot\delta^S_{LEF} + C_{X_q}\left(\alpha,\beta,\delta^S_{LEF}\right)\cdot q \\ &\quad + C_{X_{cross}} \\ C_{X_{cross}} &= -\frac{b}{l_e}\cdot\left|\frac{1}{2}\cdot C_{n_{\delta_a}}\left(\alpha,\beta,\delta'_{LEF_l}\right)\cdot\delta'_{al} + \frac{1}{2}\cdot C_{n_{\delta_a}}\left(\alpha,\beta,\delta'_{LEF_r}\right)\cdot\delta'_{ar}\right| \approx \\ &= -\frac{b}{l_e}\cdot\left|C_{n_{\delta_a}}\left(\alpha,\beta,\delta^S_{LEF}\right)\cdot\delta^S_a\right| \end{aligned} \quad \textbf{(3-14)}$$

Force coefficient in z-axis:

$$\begin{aligned} C_{Z_T} &= \frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{el}) + \frac{1}{2}\cdot C_Z(\alpha,\beta,\delta_{er}) + C_{Z_{LEF}}\cdot\delta^S_{LEF} + C_{Z_q}\left(\alpha,\beta,\delta^S_{LEF}\right)\cdot q \\ &\quad + C_{Z_{cross}} \\ C_{Z_{cross}} &= -\frac{b}{l_a}\cdot\left(\frac{1}{2}\cdot C_{l_{\delta_a}}\left(\alpha,\beta,\delta'_{LEF_l}\right)\cdot\delta'_{al} + \frac{1}{2}\cdot C_{l_{\delta_a}}\left(\alpha,\beta,\delta'_{LEF_r}\right)\cdot\delta'_{ar}\right) \\ &\approx -\frac{b}{l_a}\cdot C_{l_{\delta_a}}\left(\alpha,\beta,\delta^S_{LEF}\right)\cdot\delta^S_a \end{aligned} \quad \textbf{(3-15)}$$

Pitching moment coefficient:

$$C_{m_T} = \frac{1}{2} \cdot C_m(\alpha, \beta, \delta_{el}) + \frac{1}{2} \cdot C_m(\alpha, \beta, \delta_{er}) + C_{Z_T} \cdot \left(x_{cg_r} - x_{cg}\right) + C_{m_{LEF}} \cdot \delta_{LEF}^S + C_{m_q}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot q + \frac{1}{2} \cdot \Delta C_m(\alpha, \delta_{el}) + \frac{1}{2} \cdot \Delta C_m(\alpha, \delta_{er}) + \frac{1}{2} \cdot \Delta C_{m_{ds}}(\alpha, \delta_{el}) + \frac{1}{2} \cdot \Delta C_{m_{ds}}(\alpha, \delta_{er}) \quad \textbf{(3-16)}$$

Force coefficient in y-axis:

$$C_{Y_T} = \frac{1}{2} \cdot C_Y(\alpha, \beta, \delta_{el}) + \frac{1}{2} \cdot C_Y(\alpha, \beta, \delta_{er}) + C_{Y_{LEF}} \cdot \delta_{LEF}^S + C_{Y_{ail}} + C_{Y_{\delta_r}} \cdot \delta_r' + C_{Y_r}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot r + C_{Y_p}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot p \quad \textbf{(3-17)}$$

$$C_{Y_{ail}} = \frac{1}{2} \cdot C_{Y_{\delta_a}}\left(\alpha, \beta, \delta_{LEF_l}'\right) \cdot \delta_{al}' - \frac{1}{2} \cdot C_{Y_{\delta_a}}\left(\alpha, \beta, \delta_{LEF_r}'\right) \cdot \delta_{ar}' \approx C_{Y_{\delta_a}}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot \delta_a^A$$

Yawing moment coefficient:

$$C_{n_T} = \frac{1}{2} \cdot C_n(\alpha, \beta, \delta_{el}) + \frac{1}{2} \cdot C_n(\alpha, \beta, \delta_{er}) + C_{n_{LEF}} \cdot \delta_{LEF}^S + -C_{Y_T} \cdot \left(x_{cg_r} - x_{cg}\right) \cdot \frac{\bar{c}}{b} + C_{n_{ail}} + C_{n_{\delta_r}} \cdot \delta_r' + C_{n_r}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot r + C_{n_p}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot p + \left(\frac{1}{2} \cdot \Delta C_{n_\beta}(\alpha, \delta_{el}) + \frac{1}{2} \cdot \Delta C_{n_\beta}(\alpha, \delta_{er})\right) \cdot \beta + C_{n_{cross}} \quad \textbf{(3-18)}$$

$$C_{n_{ail}} = \frac{1}{2} \cdot C_{n_{\delta_a}}\left(\alpha, \beta, \delta_{LEF_l}'\right) \cdot \delta_{al}' - \frac{1}{2} \cdot C_{n_{\delta_a}}\left(\alpha, \beta, \delta_{LEF_r}'\right) \cdot \delta_{ar}' \approx C_{n_{\delta_a}}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot \delta_a^A$$

$$C_{n_{cross}} = \frac{l_e}{b} \cdot \left(\frac{1}{2} \cdot C_X(\alpha, \beta, \delta_{el}) - \frac{1}{2} \cdot C_X(\alpha, \beta, \delta_{er})\right) + \frac{l_{LEF}}{b} \cdot C_{X_{LEF}} \cdot \delta_{LEF}^A$$

Rolling moment coefficient:

$$C_{l_T} = \frac{1}{2} \cdot C_l(\alpha, \beta, \delta_{el}) + \frac{1}{2} \cdot C_l(\alpha, \beta, \delta_{er}) + C_{l_{LEF}} \cdot \delta_{LEF}^S + C_{l_{ail}} + C_{l_{\delta_r}} \cdot \delta_r' + C_{l_r}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot r + C_{l_p}\left(\alpha, \beta, \delta_{LEF}^S\right) \cdot p + \left(\frac{1}{2} \cdot C_{l_\beta}(\alpha, \delta_{el}) + \frac{1}{2} \cdot C_{l_\beta}(\alpha, \delta_{er})\right) \cdot \beta \quad \textbf{(3-19)}$$

$$C_{l_{ail}} = \frac{1}{2} \cdot C_{l_{\delta_a}}(\alpha, \beta, \delta'_{LEF_l}) \cdot \delta'_{al} - \frac{1}{2} \cdot C_{l_{\delta_a}}(\alpha, \beta, \delta'_{LEF_r}) \cdot \delta'_{ar}$$
$$\approx C_{l_{\delta_a}}(\alpha, \beta, \delta^S_{LEF}) \cdot \delta^A_a$$

$$C_{l_{cross}} = -\frac{l_e}{b} \cdot \left(\frac{1}{2} \cdot C_Z(\alpha, \beta, \delta_{el}) - \frac{1}{2} \cdot C_Z(\alpha, \beta, \delta_{er}) \right) + \frac{l_{LEF}}{b} \cdot C_{Z_{LEF}} \cdot \delta^A_{LEF}$$

It is encouraged to perform a dimensionality analysis of added cross terms using equations (3-7) and (3-3) in order to understand better them and understand mistakes in [73].

Note as the added cross terms introduce non-conventional ways to control the aircraft like roll using elevators or pitch using ailerons.

With this aerodynamic model and using equations in section 3.1.4, fully asymmetrical or 7dof model is configured.

3.3 Reconfigurable model

In this section reconfiguration parameters that define faults and deviations discussed in section 1.3.3 are defined. These parameters and their combinations represent all the variety of possible faults and deviations mentioned in section 1.4.

Finally in this section, a modification of the aerodynamic model presented in section 3.2 is also performed, so that the final model is a fully asymmetric or 7dof including a full set of reconfigurable capabilities.

3.3.1 Reconfiguration parameters

In this section, all parameters used to achieve in the model all faults and deviations stated in section 1.4 are explained.

These parameters represent different natures of faults and deviations and different combinations of them can represent common actual problems in aircrafts. For example, an impact of a foreign object that destroys part of the right semi-wing can be though to be expressed as loss of a given % of wing surface and loss of aileron and leading edge flap. If mass properties also are considered to be changed, mass, CG position and moments of inertia can be also changed.

The first parameters explained are related with control surfaces in Table 3-4:

Parameter	Range	Meaning
$Blockage\_at$	$[Correspondent\ defl.range, NaN]$	$\begin{cases} no\ blockage\ occurs \quad if\ NaN \\ blockage\ at\ given\ deflection\ otherwise \end{cases}$
$Blockage\_now$	$[0\ or\ 1]: Boolean$	$\begin{cases} no\ blockage\ occurs \quad if\ 0 \\ blockage\ at\ current\ deflection\ if\ 1 \end{cases}$
Effectiveness (η_i)	$[0-1]$	Effectiveness in unitary representation: e.g. 0 is surface loss or floating

Table 3-4 Control surfaces reconfiguration parameters

There are two parameters that represent blockages of each surface and an effectiveness parameter. Meaning and applicability of these parameters are explained at section 3.3.2.

The next set of parameters is parameters related with aerodynamics, mass properties and surfaces, that is, with the airframe in general and not with control surfaces:

Parameter	Range	Meaning
$\Delta\ mass$	$[-1, \infty]$	Unitary increment/decrement of mass
$\Delta\ S_L$	$[0,1]$	Unitary decrement of left semi-wing surface
$\Delta\ S_R$	$[0,1]$	Unitary decrement of right semi-wing surface
$\Delta\ S_{fin}$	$[0,1]$	Unitary decrement of fin surface
x_{cg}	$[0,1]$	Default value is 0.3. Unitary CG position of $\bar{c}$
$\Delta\ I_i$	$[-1, \infty]$	Unitary increment/decrement of Moments of inertia $[I_y, I_{xz}, I_z, I_x]$
$\Delta C_D, \Delta C_L, \Delta C_m$	$[-\infty, \infty]$	Increment/decrement of correspondent coefficients Typically $[C_D, C_L, C_m] \sim [0.036, 0.2, -0.054]$

Table 3-5 Reconfiguration parameters of the Airframe

Given a decrement of wing or fin surfaces, it is possible to compute the actual surface area. In fact, the aerodynamic model uses the current surface area to compute the effect of their variations.

The method used to compute the current surfaces areas given a reduction of them, takes into account Figure 3-3 and the fact that almost 30% of the wing surface belong to the body. The main assumption for these estimations is that surfaces loss always occurs from tip to root. The method is presented in equation (3-20).

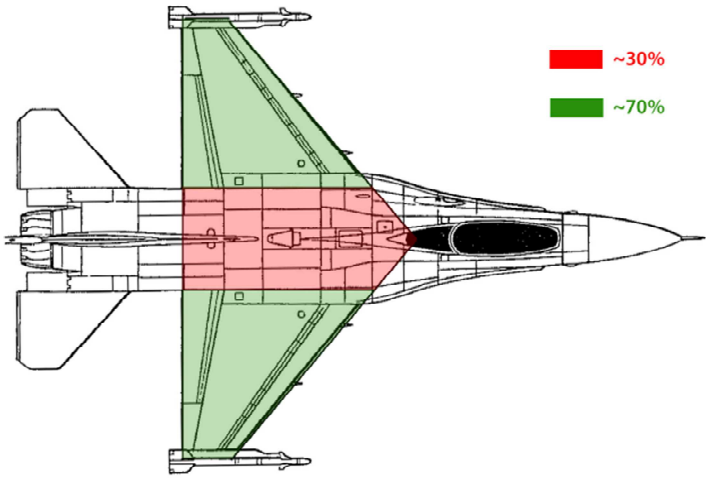


Figure 3-3 F-16 planform Wing areas

Note that $S_{L_{baseline}} = S_{R_{baseline}} = \frac{S}{2}$

$$S_{L_{current}} \approx S_{L_{baseline}} \cdot (0.3 + 0.7 \cdot (1 - \Delta S_L)\,) \qquad \textbf{(3-20)}$$

$$S_{R_{current}} \approx S_{R_{baseline}} \cdot (0.3 + 0.7 \cdot (1 - \Delta S_R)\,)$$

Regarding fin surface, a similar procedure is carried out taking into account that the ventral fins are $\sim 10\%$ of the total fin surface:

$$S_{fin_{current}} \approx S_{fin_{baseline}} \cdot \left(0.1 + 0.9 \cdot \left(1 - \Delta S_{fin}\right)\right)$$

3.3.2 Issues classification and representation

In this section a discussion of the nature of issues presented at section 1.4 is performed, assigning a way to represent the fault or deviation with parameters. These parameters are presented in section 3.3.1.

It may be needed to visit first section 3.3.3 in order to understand some notation.

3.3.2.1 Faults

Regarding the classified faults in section 1.4, the following decompositions and interpretation of them can be done:

- Control surfaces faults

- Blockages: Can be understood as impositions on actuators dynamics
    - At current deflection: Stop the dynamics of correspondent actuator (section 3.1.5) at current deflection ignoring further commanded signals
    - At a given deflection: Command a constant (given) deflection ignoring further commanded signals
- Floating or loss: Can be understood as a loss of the respective term in aerodynamic model. Loss of 100% of effectiveness → Eliminate effect of the effector in aerodynamic coefficients.

- Structure Faults
    - Loss of lifting surface: Can be understood as a 100% loss of surface →Eliminate the effect in coefficients and add asymmetry. Also, compute new Aerodynamic Centre position and compute cross terms
    - Change in dragging properties: Compute C_D,modify it and recomputed correspondent C_{X_X} , C_{Z_X}

3.3.2.2 Deviations

Regarding the classified deviations in section 1.4, the following decompositions and interpretation of them can be done:

- Effectors Deviations
    - Partial loss of a surface: Reduction of effectiveness → Reduce the contribution of the correspondent surface in coefficients
    - Aerodynamics interference and icing of control surfaces: can be seen as reduction of effectiveness
- Structure Deviations
    - Partial loss of lifting surface (wing or fin): Reduction on surface area → Given a reduction of surface area, weight the correspondent contributions in coefficients with $\frac{S_{current}}{S_{baseline}}$
        - Wing: Can be seen as a change in C_{X_X} , C_{Z_X} , C_{m_X} due to changes in Lift, Drag and pitching coefficients. Also, compute new Aerodynamic centre position and compute cross terms
        - Fin: Can be seen as a change in C_{n_X} , C_{l_X}

        - Change in dragging properties: Compute C_D,modify it and recomputed correspondent C_{X_X} , C_{Z_X}
- Global changes:
    - o Changes due to Mach and Altitude: They are already considered in the aerodynamic model
    - o Mass properties variations: Given changes of Moments of inertia, CG and mass → compute current mass properties and used them

3.3.3 Aerodynamic model

In this section, modifications to the model explained in section 3.2 are explained. These modifications try to collect all the effects that every fault or deviation taken into account could have in reality, keeping always the maximum level of realism.

The first things to deal with are the effects of wing surface loss and asymmetries. As it can be understood from equations (3-13), the all aerodynamic forces are applied at the Aerodynamic Centre which is located at x_{cg_r}. Also, translational and rotational dynamics equations were formulated at CG position, so, aerodynamic forces create moments as shown in equations (3-13).

When wing asymmetry occurs, wing forces are applied at the Aerodynamic Centre of the wing which is no longer the previous one. So, the first step is to estimate the new wing AC position.

Working with each semi-wing parameter, semi-spans are (see [78]):

$$b_{current_L} = \frac{2 \cdot S_{L_{current}}}{c_{root} \cdot (1+\lambda)} \qquad b_{current_R} = \frac{2 \cdot S_{R_{current}}}{c_{root} \cdot (1+\lambda)}$$

Being c_{root} the root chord and λ the wing taper ratio.

Then, defining the wing AC as $\{y_{AC}, x_{cg_r}\}$

$$y_{AC} = \frac{1}{2} \cdot \left(y_{MAC_R} - y_{MAC_L}\right)$$

$$y_{MAC_L} = \frac{b_{current_L}}{3} \cdot \frac{(1+2\cdot\lambda)}{(1+\lambda)} \qquad y_{MAC_R} = \frac{b_{current_R}}{3} \cdot \frac{(1+2\cdot\lambda)}{(1+\lambda)}$$

Where y_{MAC_R} is the transversal position of the MAC of the right semi-wing (see [78]).

From aircraft geometry and taken into account wing taper ratio, it is possible to compute which is the longitudinal (x-axis) displacement of the AC.

$$x_{cg_r} = 0.35 + 2.11 \cdot \left(\frac{y_{AC}}{b/_2}\right) \cdot \frac{1}{\bar{c}}$$

Note that variations in x_{cg_r} are negligible when compared with variations in y_{AC}.

Note also that transversal variations of CG positions are not computed due to the fact they are negligible compared with transversal variations of wing Aerodynamic Centre variations. As well, in case of wing surface asymmetries, taking into account this CG transversal variation will reduce the coupling effect of the transversal variation of the Aerodynamic Centre. So, not taking it into account is a conservative choice.

3.3.3.1 Aerodynamic coefficients

Modifications due to actuators on this model are performed separating the contributions of control surfaces and airframe to the total coefficients. That is, if coefficients are generally a non-linear function f:

$$\{\mathrm{C_X}, \mathrm{C_Y}, \mathrm{C_Z}, \mathrm{C_l}, \mathrm{C_m}, \mathrm{C_n}\} = f(\alpha, \beta, p, q, r, \{\delta_e, \delta_a, \delta_r, \delta_{LEF}\})$$

Both contributions are separated as follows:

$$\{\mathrm{C_X}, \mathrm{C_Y}, \mathrm{C_Z}, \mathrm{C_l}, \mathrm{C_m}, \mathrm{C_n}\} = f_X + f_D$$

$$f_X = f(\alpha, \beta, p, q, r, \{\delta_e, \delta_a, \delta_r, \delta_{LEF}\} = 0)$$

$$f_D = f(\alpha, \beta, p, q, r, \{\delta_e, \delta_a, \delta_r, \delta_{LEF}\}) - f(\alpha, \beta, p, q, r, \{\delta_e, \delta_a, \delta_r, \delta_{LEF}\} = 0)$$

Where f_X refers to the contribution of the airframe or dynamics (α, β, p, q, r) and f_D refers to the contribution of the control surfaces deflections ($\delta_e, \delta_a, \delta_r, \delta_{LEF}$). Then, f_D can be weighted with an effectiveness parameter for example.

This method is needed since the influence of elevator deflections is not-linear and a weighting in elevator deflection will not provide a realistic effect. In contrast, rudder, aileron and LEF contributions are linear with their correspondent pseudo deflections as stated in equations (3-14) to (3-19), so pseudo deflections can be weighted directly.

Dynamics effect	Left elevator deflection	Right elevator deflection
$C_{X_X} = C_X(\alpha, \beta, \delta_e = 0)$	$C_{X_{D_L}} = C_X(\alpha, \beta, \delta_e = \delta_{e_l}) - C_X(\alpha, \beta, \delta_e = 0)$	$C_{X_{D_R}} = C_X(\alpha, \beta, \delta_e = \delta_{e_r}) - C_X(\alpha, \beta, \delta_e = 0)$
$C_{Z_X} = C_Z(\alpha, \beta, \delta_e = 0)$	$C_{Z_{D_L}} = C_Z(\alpha, \beta, \delta_e = \delta_{e_l}) - C_Z(\alpha, \beta, \delta_e = 0)$	$C_{Z_{D_R}} = C_Z(\alpha, \beta, \delta_e = \delta_{e_r}) - C_Z(\alpha, \beta, \delta_e = 0)$
Same with rest of coefficients: $C_Y, C_m, C_n, C_l, \Delta C_m, \Delta C_{m_{ds}}, \Delta C_{n_\beta}, \Delta C_{l_\beta}$		

Table 3-6 Elevators contributions separation

Table 3-6 shows how this separation is. Note that this separation is performed on all coefficients and derivatives that are function of elevator deflections δ_e: $C_X, C_z, C_Y, C_m, C_n, C_l, \Delta C_m, \Delta C_{m_{ds}}, \Delta C_{n_\beta}, \Delta C_{l_\beta}$ as defined in equations (3-13).

In the other hand, due to the fact that ailerons, LEFs and rudder terms are linear with their pseudo deflections, the correspondent contributions can be weighted through directly weighting the pseudo deflections:

$$\delta_{e_l} \Longrightarrow \delta_{e_l} \qquad \delta_{e_r} \Longrightarrow \delta_{e_r}$$

$$\delta_{a_l} \Longrightarrow \delta'_{a_l} = \frac{\delta_{a_l}}{20} \cdot \eta_{ail_l} \qquad \delta_{a_r} \Longrightarrow \delta'_{a_r} = \frac{\delta_{a_r}}{20} \cdot \eta_{ail_r}$$

$$\delta_r \Longrightarrow \delta'_r = \frac{\delta_r}{30} \cdot \eta_{rudder}$$

$$\delta_{LEF_l} \Longrightarrow \delta'_{LEF_l} = \left(1 - \frac{\delta_{LEF_l}}{25}\right) \cdot \eta_{LEF_l} \qquad \delta_{LEF_r} \Longrightarrow \delta'_{LEF_r} = \left(1 - \frac{\delta_{LEF_r}}{25}\right) \cdot \eta_{LEF_r}$$

Being η_i the effectiveness of each control surface.

Note that symmetric and asymmetric deflections are obtained with the same method as in Table 3-3.

In order to compute effect of wing surface loss, which mainly affect to Lift, Drag and pitching moment coefficients, it is needed to compute these coefficients firstly.

$$C_L = -C_{Z_X} \cdot \cos(\alpha) + C_{X_X} \cdot sin(\alpha) \quad and \quad C_D = -C_{X_X} \cdot cos(\alpha) - C_{Z_X} \cdot sin(\alpha)$$

The change introduced by surface loss is represented as proportional to the ratio of current surface and reference surface, as said. Also, in order to introduce which would be the effect of the fuselage and rest or airframe, fixed contributions to $C_L, C_D \; and \; C_m$ are stipulated as 5%, 15% and 40% respectively. Note that these numbers are only rough estimations.

$$C_L' = C_L \cdot \left(0.05 + 0.95 \cdot \frac{S_{current}}{S}\right) \ and \ C_D' = C_D \cdot \left(0.15 + 0.85 \cdot \frac{S_{current}}{S}\right)$$

Note that $S_{\text{current}} = S_{R_{current}} + S_{L_{current}}$

In order to recover coefficients in body axes it is needed to project again back in body axes. Equations (3-21), also introduce variations of aerodynamic coefficients stated in section 3.3.1, ΔC_D, ΔC_L, ΔC_m:

$$\begin{aligned} C_{X_X}' &= -(C_D' + \Delta C_D) \cdot cos(\alpha) + (C_L' + \Delta C_L) \cdot sin(\alpha) \\ C_{Z_X}' &= -(C_D' + \Delta C_D) \cdot sin(\alpha) - (C_L' + \Delta C_L) \cdot cos(\alpha) \\ C_{m_X}' &= C_{m_X} \cdot \left(0.40 + 0.60 \cdot \frac{S_{current}}{S}\right) + \Delta C_m \end{aligned} \quad \textbf{(3-21)}$$

A similar procedure is performed for fin surface loss, which affects to rolling and yawing moments:

$$C_{n_X}' = C_{n_X} \cdot \left(0.2 + 0.8 \cdot \frac{S_{fin_{current}}}{S_{fin}}\right) , C_{l_X}' = C_{l_X} \cdot \left(0.25 + 0.75 \cdot \frac{S_{fin_{current}}}{S_{fin}}\right)$$

Finally, aerodynamics coefficients are:

Force coefficient in x-axis:

$$\begin{aligned} C_{X_T} &= C_{X_X}' + \frac{1}{2} \cdot C_{X_{D_L}} \cdot \eta_{e_l} + \frac{1}{2} \cdot C_{X_{D_R}} \cdot \eta_{e_r} + C_{X_{LEF}} \cdot \delta_{LEF}^S + C_{X_q}(\alpha, \beta, \delta_{LEF}^S) \cdot q \\ &\quad + C_{X_{cross}} \\ C_{X_{cross}} &= -\frac{b}{l_e} \cdot \left|\frac{1}{2} \cdot C_{n_{\delta_a}}(\alpha, \beta, \delta_{LEF_l}') \cdot \delta_{al}' + \frac{1}{2} \cdot C_{n_{\delta_a}}(\alpha, \beta, \delta_{LEF_r}') \cdot \delta_{ar}'\right| \approx \\ &\quad = -\frac{b}{l_e} \cdot \left|C_{n_{\delta_a}}(\alpha, \beta, \delta_{LEF}^S) \cdot \delta_a^S\right| \end{aligned} \quad \textbf{(3-22)}$$

Force coefficient in z-axis:

$$\begin{aligned} C_{Z_T} &= C_{Z_X}' + \frac{1}{2} \cdot C_{Z_{D_L}} \cdot \eta_{e_l} + \frac{1}{2} \cdot C_{Z_{D_R}} \cdot \eta_{e_r} + C_{Z_{LEF}} \cdot \delta_{LEF}^S + C_{Z_q}(\alpha, \beta, \delta_{LEF}^S) \cdot q \\ &\quad + C_{Z_{cross}} \end{aligned} \quad \textbf{(3-23)}$$

$$C_{Z_{cross}} = -\frac{b}{l_a} \cdot \left(\frac{1}{2} \cdot C_{l_{\delta_a}}(\alpha, \beta, \delta'_{LEF_l}) \cdot \delta'_{al} + \frac{1}{2} \cdot C_{l_{\delta_a}}(\alpha, \beta, \delta'_{LEF_r}) \cdot \delta'_{ar}\right)$$
$$\approx -\frac{b}{l_a} \cdot C_{l_{\delta_a}}(\alpha, \beta, \delta^S_{LEF}) \cdot \delta^S_a$$

Pitching moment coefficient:

$$\begin{aligned} C_{m_T} = C_{m'_X} + \frac{1}{2} \cdot C_{m_{D_L}} \cdot \eta_{e_l} \; + \frac{1}{2} \cdot C_{m_{D_R}} \cdot \eta_{e_r} + C_{Z_T} \cdot (x_{cg_r} - x_{cg}) + C_{m_{LEF}} \cdot \delta^S_{LEF} \\ + C_{m_q}(\alpha, \beta, \delta^S_{LEF}) \cdot q + \Delta C_{m_X} + \frac{1}{2} \cdot \Delta C_{m_{D_L}} \cdot \eta_{e_l} + \frac{1}{2} \cdot \Delta C_{m_{D_R}} \cdot \eta_{e_r} \\ + \Delta C_{m_{ds_X}} + \frac{1}{2} \cdot \Delta C_{m_{ds_{D_L}}} \cdot \eta_{e_l} + \frac{1}{2} \cdot \Delta C_{m_{ds_{D_R}}} \cdot \eta_{e_r} \end{aligned} \quad \textbf{(3-24)}$$

Force coefficient in y-axis:

$$\begin{aligned} C_{Y_T} = C_{Y_X} + \frac{1}{2} \cdot C_{Y_{D_L}} \cdot \eta_{e_l} \; + \frac{1}{2} \cdot C_{Y_{D_R}} \cdot \eta_{e_r} + C_{Y_{LEF}} \cdot \delta^S_{LEF} + C_{Y_{ail}} + C_{Y_{\delta_r}} \cdot \delta'_r \\ + C_{Y_r}(\alpha, \beta, \delta^S_{LEF}) \cdot r + C_{Y_p}(\alpha, \beta, \delta^S_{LEF}) \cdot p \end{aligned} \quad \textbf{(3-25)}$$

$$C_{Y_{ail}} = \frac{1}{2} \cdot C_{Y_{\delta_a}}(\alpha, \beta, \delta'_{LEF_l}) \cdot \delta'_{al} - \frac{1}{2} \cdot C_{Y_{\delta_a}}(\alpha, \beta, \delta'_{LEF_r}) \cdot \delta'_{ar}$$
$$\approx C_{Y_{\delta_a}}(\alpha, \beta, \delta^S_{LEF}) \cdot \delta^A_a$$

Yawing moment coefficient:

$$\begin{aligned} C_{n_T} = C_{n'_X} + \frac{1}{2} \cdot C_{n_{D_L}} \cdot \eta_{e_l} \; + \frac{1}{2} \cdot C_{n_{D_R}} \cdot \eta_{e_r} + C_{n_{LEF}} \cdot \delta^S_{LEF} + -C_{Y_T} \cdot (x_{cg_r} - x_{cg}) \\ \cdot \frac{\bar{c}}{b} + C_{n_{ail}} + C_{n_{\delta_r}} \cdot \delta'_r + C_{n_r}(\alpha, \beta, \delta^S_{LEF}) \cdot r + C_{n_p}(\alpha, \beta, \delta^S_{LEF}) \cdot p \\ + \left(\Delta C_{n_{\beta_X}} + \frac{1}{2} \cdot \Delta C_{n_{\beta_{D_L}}} \cdot \eta_{e_l} + \frac{1}{2} \cdot \Delta C_{n_{\beta_{D_R}}} \cdot \eta_{e_r}\right) \cdot \beta + C_{n_{cross}} \end{aligned} \quad \textbf{(3-26)}$$

$$C_{n_{ail}} = \frac{1}{2} \cdot C_{n_{\delta_a}}(\alpha, \beta, \delta'_{LEF_l}) \cdot \delta'_{al} - \frac{1}{2} \cdot C_{n_{\delta_a}}(\alpha, \beta, \delta'_{LEF_r}) \cdot \delta'_{ar}$$
$$\approx C_{n_{\delta_a}}(\alpha, \beta, \delta^S_{LEF}) \cdot \delta^A_a$$

$$C_{n_{cross}} = \frac{l_e}{b} \cdot \left(\frac{1}{2} \cdot C_{X_{D_L}} \cdot \eta_{e_l} \; - \frac{1}{2} \cdot C_{X_{D_R}} \cdot \eta_{e_r}\right) + \frac{l_{LEF}}{b} \cdot C_{X_{LEF}} \cdot \delta^A_{LEF} - \frac{\mathrm{y_{AC}}}{\mathrm{b}} \cdot C_{X'_X}$$

Rolling moment coefficient:

$$
\begin{aligned}
C_{l_T} = C_{l_X}' + \frac{1}{2}\cdot C_{l_{D_L}}\cdot \eta_{e_l} \; + \frac{1}{2}\cdot C_{l_{D_R}}\cdot \eta_{e_r} + C_{l_{LEF}}\cdot \delta_{LEF}^{S} + C_{l_{ail}} + C_{l_{\delta_r}}\cdot \delta_r' \\
+ C_{l_r}\left(\alpha,\beta,\delta_{LEF}^{S}\right)\cdot r + C_{l_p}\left(\alpha,\beta,\delta_{LEF}^{S}\right)\cdot p \\
+ \left(\Delta C_{l_{\beta_X}} + \frac{1}{2}\cdot \Delta C_{l_{\beta_{D_L}}}\cdot \eta_{e_l} + \frac{1}{2}\cdot \Delta C_{l_{\beta_{D_R}}}\cdot \eta_{e_r}\right)\cdot \beta
\end{aligned}
\qquad \textbf{(3-27)}
$$

$$
\begin{aligned}
C_{l_{ail}} &= \frac{1}{2}\cdot C_{l_{\delta_a}}\left(\alpha,\beta,\delta_{LEF_l}'\right)\cdot \delta_{al}' - \frac{1}{2}\cdot C_{l_{\delta_a}}\left(\alpha,\beta,\delta_{LEF_r}'\right)\cdot \delta_{ar}' \\
&\approx C_{l_{\delta_a}}\left(\alpha,\beta,\delta_{LEF}^{S}\right)\cdot \delta_a^{A}
\end{aligned}
$$

$$
C_{l_{cross}} = -\frac{l_e}{b}\cdot\left(\frac{1}{2}\cdot C_{Z_{D_L}}\cdot \eta_{e_l} - \frac{1}{2}\cdot C_{Z_{D_R}}\cdot \eta_{e_r}\right) + \frac{l_{LEF}}{b}\cdot C_{Z_{LEF}}\cdot \delta_{LEF}^{A} + \frac{\mathrm{y_{AC}}}{\mathrm{b}}\cdot C_{Z_X}'
$$

Note the cross effects of asymmetric wing surface loss on the yawing and rolling moment, that is produced by the additional cross terms $C_{l_{cross}} = + \frac{\mathrm{y_{AC}}}{\mathrm{b}}\cdot C_{Z_X}'$ and $C_{n_{cross}} = - \frac{\mathrm{y_{AC}}}{\mathrm{b}}\cdot C_{X_X}'$.

3.3.4 Implementation in Matlab/Simulink

In order to let the user of the software developed and to have a simple way of injecting failures and deviations during the development stage, a Graphical User Interface was built which operates within Simulink.

This interface lets the user introduce any change or fault at any moment of the simulation, enabling an easy way to evaluate performance under fault or deviation. As it can be seen in appendix A.4, all the variables stated in section 3.3.1 can be changed with the GUI.

As well, this GUI is programed to avoid the user introduces non-consistent combinations of parameters: for example, if the user set as floating a control surface, the GUI automatically removes options for blocking and reducing effectiveness of that surface.

This GUI can be found in appendix A.4.

4 Non-linear Dynamic Inversion and Control allocation

This section presents two of the main algorithms that compose the principal core of the controller: Non-linear Dynamic Inversion (NDI) and Control Allocation (CA).

These techniques have been used extensively as said in sections 2.5.1 and 2.5.2. Their simplicity and usefulness make them good tool to be used in control systems as long as the designer is provided with a model of the plant to be controlled.

Also, these techniques allow reconfigurations, so, if the design implements a FDI system, the controller can be reconfigured in consequence.

The first section presented here is a brief introduction of how NDI and CA are fit in the general picture presented in section 1.3, what their basic theory is and what their inputs and outputs are.

Secondly, each technique is described separately, explaining the general theory, equations and applications.

As well, a different approach of implementing NDI and CA to those found in literature, is explained in section 4.4.

Finally, implementation of reconfiguration (advanced concept) of these algorithms is explained.

4.1 General approach

The aim is to build a closed loop system that allows the control of the aircraft. In fact, in order to control the plant on the desired variables, several control loops will be required.

As first approach, let us assume there is a fully realistic invertible model $G(x, \delta_i)$ of the plant with no uncertainty. That is, $G(x, \delta_i)$ represents exactly the real plant and there exist a way to obtain $G^{-1}(x, \delta_i)$.

Figure 4-1 shows that the real purpose of using NDI and CA is to achieve the actual inverse of the plant to perform its control. The input to this perfect inverse is given by the Desired Dynamics block which generates the time derivative of the reference signal.

With that, if $G(x, \delta_i)$ really represents the actual plant and $G^{-1}(x, \delta_i)$ is an actual inverse of the plant dynamics, the simple open loop in Figure 4-1 can make a perfect reference tracking, making: $y = y_{ref}$.

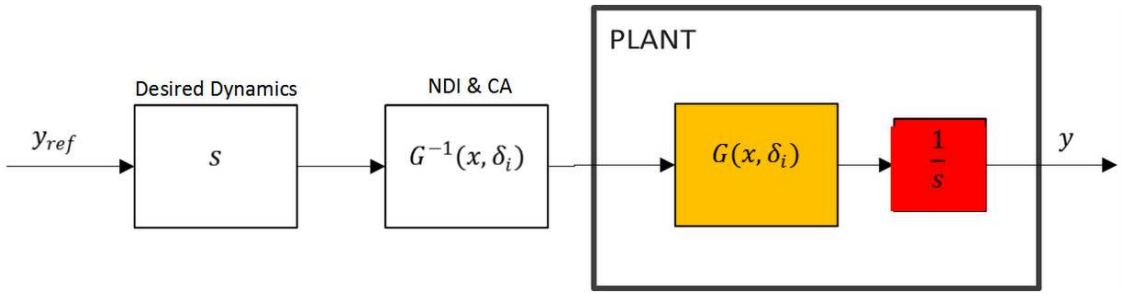


Figure 4-1 NDI & CA general approach Open loop

But, of course, those assumptions are not realistic, so something else has to be done, as explained in next sections.

Note that the purpose of the Desired Dynamics and controller module is to generate a desired time derivative of the controlled variable. Then, NDI and CA perform an inverse of this desired time derivative through an approximate non-linear representation of the plant, generating control signals.

In the case of a piloted aircraft, where the inner controller controls body rates at first instance, the input to the NDI block are angular accelerations, so the model used in these algorithms is the model of the rotational dynamics of the aircraft: equation (3-6).

General theory of NDI states a full method to achieve a complete inverse from angular accelerations to control surfaces deflections (see [32]). But here, only first part of the NDI method is used, while the real inversion (to obtain control deflections) is performed by CA.

This approach was chosen mainly because it is an over-actuated system and pure inversion cannot manage to achieve an inverse with this kind of systems. As well, CA performs an allocation of control effort at the same time of performing an inverse of its control input, so CA is preferred.

4.2 Non-linear Dynamic Inversion

Presented aircraft models are very non-linear systems. Indeed, if any general plant is needed to be controlled, its model would be usually non-linear.

Control systems using NDI have a great advantage over classical approaches. The main advantage is that the controller actually takes into account the operating point of the aircraft, like Mach and altitude. This fact means that the controller can operate in many

conditions as modelled without having to make any scheduling, reducing significantly the designing effort.

4.2.1 Dynamic Inversion concept

Dynamic Inversion recently has gained more attention in control systems community, as presented in section 2.5.1. Ducard presents the theory here presented as well in [31].

Assuming a representation of the plant dynamics as follows:

$$\dot{x}(t) = f(x) + g(x) \cdot u(t)$$
$$y(t) = h(x) \qquad \textbf{(4-1)}$$

where $y(t)$ is the state vector, $y(t)$ is the measurement vector and $\vec{u}$ is the control vector. Note that the model is considered to be linear with $\vec{u}$.

Now, deriving the measurement vector in terms of time:

$$\frac{\partial\, y(t)}{\partial\, t} = \frac{\partial\, h}{\partial\, x} \cdot \frac{\partial\, x}{\partial\, t} = \frac{\partial\, h}{\partial\, x} \cdot f(x) + \frac{\partial\, h}{\partial\, x} \cdot g(x) \cdot u(t) = F(x) + G(x) \cdot u(t) \qquad \textbf{(4-2)}$$

If only NDI is used to perform the inversion instead of CA, $G(\vec{x})$ must comply with some conditions:

Using CA	Using only NDI
$G(x) \neq 0\ \forall t, x$	$G(x) \neq 0\ \forall t, x$ $G(x)\ must\ be\ square\ \forall t, x$ $G(x)\ must\ be\ full\ rank\ \forall t, x$

Being y_c the reference signal to be tracked and y_{meas} the measurement that has to track the reference y_c, control signal can be defined as:

$$u(t) = u_c = G^{-1}(x) \cdot \left(\dot{y}_{des} - F(x)\right) \qquad \textbf{(4-3)}$$

where $\dot{y}_{des}$ is the desired time derivative of the controlled variable as shown in Figure 4-2.

Figure 4-2 shows the general scheme of the concept. Note that if CA is not used, NDI should also include the inversion done by CA.

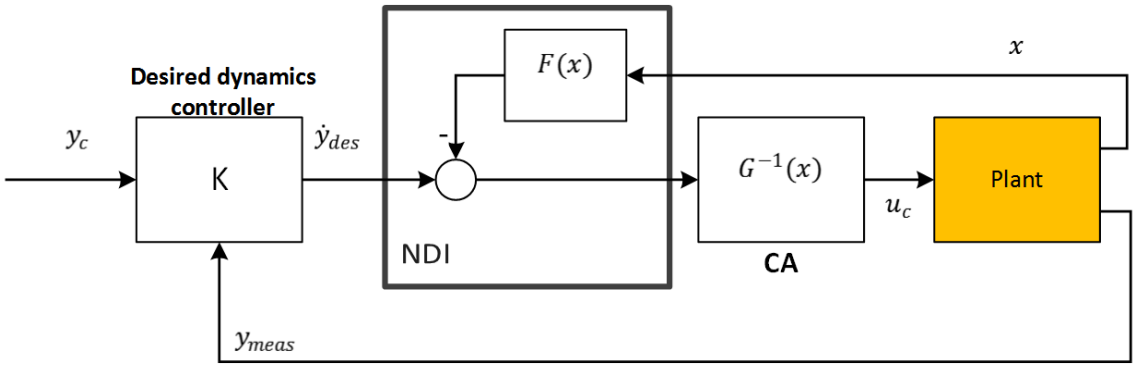


Figure 4-2 NDI concept

Generally, equation (4-3) fully defines the control signal, but it is function of $\dot{y}_{des}$ which is provided by the Desired Dynamics controller. Hence, desired dynamics controller has to be design in such a way that, working together with NDI, provides convergence. That is, it generates a $\dot{y}_{des}$ signal that allows the tracking error $y_c - y_{meas}$ converge to zero.

$$\dot{y}_{des} = K(y_c, y_{meas}) \quad : \quad \lim_{t\to\infty} |y_c - y_{meas}| = 0 \tag{4-4}$$

Where $K(y_c, y_{meas})$ represents the controller, which has two inputs: $y_c \ and \ y_{meas}$.

4.2.2 Stability proof: Controller design

In this section, stability of the inner control loop presented is proven, using a Lyapunov candidate function. In fact, the controller philosophy is designed to provide stability when working with NDI and CA.

Firstly, the error of the loop is defined as in equation (4-4):

$$e(t) = y_c(t) - y_{meas}(t) \tag{4-5}$$

Secondly, the Lyapunov candidate function is defined like the energy function of this error:

$$V(t) = \frac{1}{2} \cdot e^T(t) \cdot e(t) \tag{4-6}$$

$$V : \mathbb{R}^n \to \mathbb{R}$$

where n is the number of controlled variables. Time derivative of this function is:

$$\dot{V} = e^T \cdot \dot{e} = e \cdot (\dot{y}_c - \dot{y}_{meas})$$

Now, taking as assumption that Dynamic inversion is performed without error, that is, NDI and CA perform a Perfect Dynamic Inversion: $\dot{y}_{meas}(t) = \dot{y}_{des}(t)$

$$\dot{e} = (\dot{y}_c - \dot{y}_{des}) \quad \textbf{(4-7)}$$

In order to achieve asymptotical stability:

$$\dot{V}(t) \leq 0 \;\; \forall x, t$$

The following law is introduced:

$$\dot{e}(t) = -K \cdot e(t) \quad \textbf{(4-8)}$$

Achieving:

$$\dot{V}(t) = -K \cdot e^2(t) \leq 0 \quad \forall x, t, K > 0 \quad \textbf{(4-9)}$$

Using (4-7) and (4-8):

$$\dot{e} = -K \cdot e = (\dot{y}_c - \dot{y}_{des})$$

Finally, the control law that ensures stability is:

$$\dot{y}_{des} = \dot{y}_c + K \cdot (y_c - y_{meas}) \quad \textbf{(4-10)}$$

Equation (4-10) establishes then the basis of Explicit Model Following which is achieved through the design of a Desired Dynamics and Controller, in section 5. Note that equation (4-10) is a simple approximation of what would be the final design.

4.2.3 Perfect Dynamic Inversion

It has been assumed that if plant is represented by equations (4-1), there exists an absolute knowledge about these equations, so that they can be used in NDI and CA. That is, $F(x)\, and\, G(x)$ are perfectly known and actuators and sensors are also considered as ideal or perfect. This is Ideal or Perfect Dynamic inversion.

Under these conditions stability of the control loop has been demonstrated: equation (4-9).

With these assumptions, NDI, CA together with the Plant become a pure integrator (with the dimension of y_{meas}) that integrate directly the desired $\dot{y}_{des}$ into y_{meas}. Figure 4-3 illustrates better how this is made.

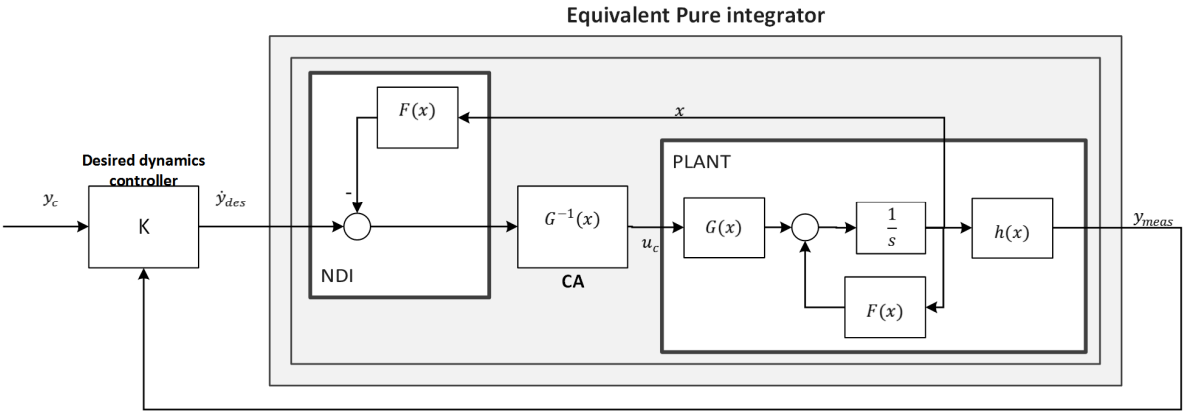


Figure 4-3 Perfect Dynamic inversion

Being:

$$y_{meas} = h(x) = \int \dot{y}_{meas} \cdot dt$$

Where $\dot{y}_{meas}$ is defined using equation (4-2) and assuming perfect sensors:

$$\dot{y}_{meas} = F(x) + G(x) \cdot u(t)$$

Using equation (4-3):

$$\dot{y}_{meas} = F(x) + G(x) \cdot \left(G^{-1}(x) \cdot \left(\dot{y}_{des} - F(x)\right)\right) = \dot{y}_{des}$$

So,

$$y_{meas} = \int \dot{y}_{des} \cdot dt \qquad \textbf{(4-11)}$$

Equation (4-11) shows how perfect dynamic inversion works and enables the control system to control the dynamics in an easy way, with just using a desired dynamics and controller: equation (4-10).

4.2.4 Dealing with uncertainties

Many considerations have been made like perfect knowledge of the plant or perfect sensors, but in reality this is not true:

- It is impossible to get a perfect mathematical definition of the plant
- There exist uncertainties that cannot be modelled
- Measurements are not perfect
- Actuators are not perfect and present noises and bias

All these factors result in that the assumed representation of the plant in equation (4-1) is just an approximate representation and $f(x), g(x)\ and\ h(x)$ cannot be considered as perfect, neither $F(x)\ nor\ G(x)$.

In order to represent all uncertainties in a simple manner, let us adopt the following notation:

- The actual plant is represented as:

$$\dot{y} = F(x) + G(x) \cdot u = \varphi(\mathrm{x}, \mathrm{u})$$

- But the representation of the plant used is:

$$\dot{\hat{y}} = \hat{F}(x) + \hat{G}(x) \cdot u = \hat{\varphi}(\mathrm{x}, \mathrm{u})$$

- So the inversion achieved is:

$$u_c = \hat{\varphi}^{-1}(\mathrm{x}, \dot{y}_{des})$$

So that, the inversion error Δ is defined:

$$\Delta = \varphi(\mathrm{x}, \mathrm{u}) - \hat{\varphi}(\mathrm{x}, \mathrm{u})$$

Hence: $\dot{y}_{meas} = \varphi(\mathrm{x}, \mathrm{u}) = \hat{\varphi}(\mathrm{x}, \mathrm{u}) + \Delta = \dot{y}_{des} + \Delta$

The error time derivative is now:

$$\dot{e} = \dot{y}_c - \dot{y}_{meas} = \dot{y}_c - \dot{y}_{des} - \Delta$$

Introducing the obtained law of equation (4-10):

$$\dot{e} = -K \cdot e - \Delta$$

$$\dot{V} = e \cdot \dot{e} = -K \cdot e^2 - \mathrm{e} \cdot \Delta \qquad \textbf{(4-12)}$$

Being $-K \cdot e^2 \leq 0 \quad \forall\, \mathrm{x}, \mathrm{t}, \mathrm{k} > 0$, but the term $-\mathrm{e} \cdot \Delta$ is <u>unknown</u> and <u>cannot</u> ensure asymptotical stability.

So, in order to modify equation (4-12) and make the term $-\mathrm{e} \cdot \Delta \leq 0$, something else has to be done. To solve this problem:

- Introduce an integral term in the controller: Robust approach
- Introduce an Adaptive controller that stabilize the system whatever Δ is: Adaptive approach

The robust approach is addressed in [31], where a robustness analysis is made for every design. This approach relies entirely in this robustness analysis in order to know stability margins.

In the other hand, a neuro-adaptive approach (section 6) can be adopted (the aim of this thesis), designing it in order to ensure stability regardless the magnitude of Δ. This approach reduces design effort, improves performance and ensures stability.

Note in equation (4-12) that even the proportional controller in equation (4-10) can provide stability, but this stability is limited by the robustness of the design of K and the magnitude of Δ. That is, stability margins. So, these margins must be evaluated as in [31].

4.2.5 Implementation

Once NDI theory developed, real implementation for an airplane plant is done. Precisely, the main control loop controls angular rates in body axes. So, the plant considered is the rotational dynamics of equation (3-6).

Inner controller only controls body rates, but if lower order (slower) variables are wanted to be controlled, additional control loops have to be added. This will be seen in section 5.

Note that main control loop (Inner controller) can also be implemented to control lower order variables directly (like AoA), without the necessity of arranging more control loops (see [22]). Here a non-linear desired dynamics is implemented, so that the inner loop only controls body rates (see section 5).

Controlling body rates the output of the desired dynamics and controller of section 4.2.1 is $\dot{y}_{des} = \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{des}$.

Using notation in section 3.1, the first step is to compute the desired moments $I_{CG} \cdot \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{des}$ and then subtract the component of these moments that are already being done by the dynamics and trust. With this, aerodynamic moments are obtained as follows:

$$M_{Aero} = I_{CG} \cdot \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix}_{des} - \left[-\vec{\omega}_b \times (I_{CG} \cdot \vec{\omega}_b) - \vec{\omega}_b \times \vec{\mathcal{L}}_{eng} + \begin{pmatrix} 0 \\ \mathrm{F_T} \cdot z_T \\ 0 \end{pmatrix} \right]$$

Then, using the equation (3-7), moment coefficients can be obtained.

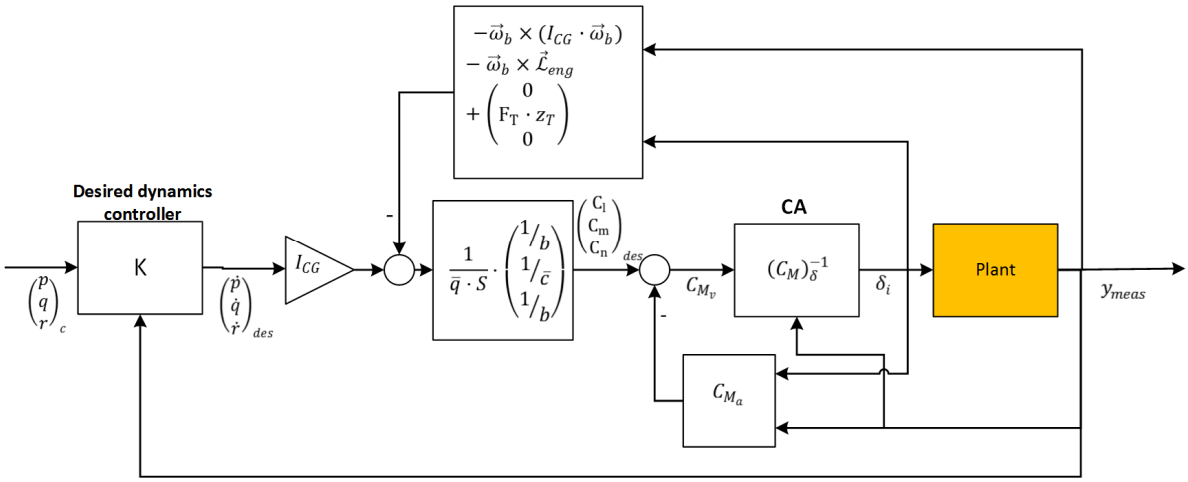


Figure 4-4 NDI implementation

As stated in sections 3.1.7.1, 3.2 and 3.3.3.1, expressions of moment coefficients are complex equations and not linear, not even with control deflections. Assuming a general expression of moment coefficients:

$$\begin{pmatrix} \mathrm{C_l} \\ \mathrm{C_m} \\ \mathrm{C_n} \end{pmatrix} = f(\alpha, \beta, p, q, r, [\delta_i]) \quad being \quad [\delta_i] = [\delta_{el}, \delta_{er}, \delta_{al}, \delta_{ar}, \delta_r, \delta_{LEF_l}, \delta_{LEF_r}] \qquad \textbf{(4-13)}$$

And following a similar technique used in section 3.3.3, contributions of the dynamics (α, β, p, q, r) and control surfaces can be separated:

$$[\mathrm{C_l}, \mathrm{C_m}, \mathrm{C_n}] = \Gamma_a + \Gamma_v$$

$$\Gamma_a = C_{M_a} = f(\alpha, \beta, p, q, r, [\delta_i] = [\delta^*])$$

$$\Gamma_v = C_{M_v} = f(\alpha, \beta, p, q, r, [\delta_i]) - f(\alpha, \beta, p, q, r, [\delta_i] = [\delta^*])$$

So, the contribution of the dynamics can be subtracted, obtaining the desired moment coefficients due to the control surfaces deflections:

$$C_{M_v} = \begin{pmatrix} \mathrm{C_l} \\ \mathrm{C_m} \\ \mathrm{C_n} \end{pmatrix}_{des} - C_{M_a} \qquad \textbf{(4-14)}$$

Being C_{M_v} the moment coefficients to be inverted, by CA for instance, to obtain control deflections. That is, the desired control deflections should generate C_{M_v} regardless how they are computed.

Now, several options can be adopted:

- Perform non-linear CA:

$$[\delta_i] = \underset{[\delta_i]}{\operatorname{argmin}} \left\| C_{M_v} - \Gamma_v(\alpha, \beta, p, q, r, [\delta_i], [\delta^*]) \right\|$$

  This approach would be similar to perform directly

$$[\delta_i] = \underset{[\delta_i]}{\operatorname{argmin}} \left\| \begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix}_{des} - f(\alpha, \beta, p, q, r, [\delta_i]) \right\|$$

- Perform linear CA, finding a method to express Γ_v linearly with deflections:

$$\Gamma_v = B(\alpha, \beta, p, q, r, [\delta_i] = [\delta^*]) \cdot \left[[\delta_i] - [\delta^*] \right]$$
$$with\ B \in \mathbb{R}^{3 \times p}\ \ and\ \ [\delta_i] \in \mathbb{R}^p$$

  where p is the number of control surfaces. The model is approximated as a Taylor expansion:

$$\begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix} = f(\alpha, \beta, p, q, r, \{\delta_i\} = \{\delta^*\}) + B(\alpha, \beta, p, q, r, \{\delta_i\} = \{\delta^*\}) \cdot \left[[\delta_i] - [\delta^*] \right] + \mathcal{O}([\delta_i] - [\delta^*])^2 \quad \textbf{(4-15)}$$

  - Finding B analytically:

$$B(\alpha, \beta, p, q, r, [\delta_i]) = \left. \frac{\partial\ f(\alpha, \beta, p, q, r, [\delta_i])}{\partial\ \delta_i} \right|_{[\delta_i]=[\delta^*]} \quad \textbf{(4-16)}$$

  - Finding by B as a numerical gradient, using $\epsilon \to 0$

$$B(\alpha, \beta, p, q, r, [\delta_i]) = \left. \frac{f(\alpha, \beta, p, q, r, [\delta_i + \epsilon]) - f(\alpha, \beta, p, q, r, [\delta_i - \epsilon])}{2 \cdot \epsilon} \right|_{[\delta_i]=[\delta^*]} \quad \textbf{(4-17)}$$

Due to de complexity of the model and due to the fact non-linear CA would be too expensive computationally, a last option is chosen. This was possible thanks to a fast implementation of the moment coefficients model in C (see [69]).

Now, $[\delta^*]$ can be defined as the control deflection the Taylor expansion in equation (4-15) is performed about. $[\delta^*]$ can be:

- Initial trim control deflection: $[\delta_i] = [\delta_i]_{TRIM}$
- Estimated trim control deflections for the given flight condition: $\alpha, \beta, p, q, r, V_{TAS}, h$ ...
- Previous commanded deflection: proposed in section 4.4
- Measured current deflections

Note that the Taylor expansion in equation (4-15) has an error:

$$\varepsilon \sim \mathcal{O}([\delta_i] - [\delta^*])^2$$

So, that the intention is to minimize ε with any strategy. Section 4.4 presents a methods so that ε is minimized.

4.3 Control Allocation

This section describes the design of the CA in order to work inverting what given by NDI.

CA is, as NDI, a well-known technique widely used in the industry ([38], [79], [80], [39], [31], [22]) mainly for over-actuated systems. It has been used, not only to resolve overactuations, but also to overcome control surfaces faults, resulting in a system able to perform reconfiguration without necessity of redesign.

The main idea of CA is to find the control deflections required to generate the required pseudo control inputs to the CA. The entire method is based in a model that can be linear or nor linear, although most of the authors use a linear model due to its computational simplicity.

The main advantages of CA are:

- Allow a robust way to perform an inverse (eq. (4-3)) even when the matrix is not square or full rank
- Performs control effort allocation: if a control surfaces get saturated, remaining controls surfaces are also taken into account to fulfil required performance
- Takes into account actuator constrains: limits, rate limits, etc.
- Takes advantages of overactuations and allows the control system to minimize different objectives: Total control deflection, deflection rates, total drag, etc.
- In case of fault, it can be reconfigure, providing as much control authority as available

There exist many methods to perform CA and finding control deflections like pseudo-inverse, direct CA, daisy chain or error minimization (using dynamic programing for example), among others. See [38] and [79].

Most of the methods are based on an optimization problem, with/without constrains. Some of them, like pseudo inverse, solve it analytically, providing low computational load but not ensuring optimality in all conditions. Other methods solve the problem using a

dynamic programing scheme on-line that increases computational load, even more when complex optimization objectives or constrains are used.

In this section, basic theory of the method is presented. As well, algorithms and designs used in this thesis are also presented.

4.3.1 Basic theory

As outlined in section 4.2.1, the goal of CA is to achieve the required C_{M_v} by generating correspondent control deflection and taking into account constrains. This virtual command is the output of NDI.

With the model describe as in 4.2.5, defining $C_{M_\delta} = B(\alpha, \beta, p, q, r, [\delta_i])$ as in Figure 4-4, the problem is to solve the set of equations:

$$C_{M_\delta}(t) \cdot \left[[\delta_i] - [\delta^*]\right] = C_{M_v}(t) \qquad \textbf{(4-18)}$$

As said, for the case of this thesis a linear CA and on-line linearized approach is chosen. That is, effectiveness matrix B is computed as in equation (4-17).

CA, as said, is based on an optimization problem solution in which control deflections are computed. For instance, the general problem can be formulated as follows:

$$u = \underset{u}{\operatorname{argmin}} J(x, u, v)$$

$$Subject\ to: B \cdot u = v$$

Defining u as control signals, $J(x, u)$ any cost function and $B \cdot u = v$ the equality constrain CA has to solve, that is, equation (4-18).

Generally, and depending on how $J(x, u)$ is, this problem has an analytical solution. For instance, pseudo-inverse method is the solutions of a particular case.

But physically, actuators have limits and cannot operate beyond certain limits, so that these constrains have to be added to the problem:

$$u = \underset{u}{\operatorname{argmin}} J(x, u, v)$$

$$Subject\ to: \begin{cases} B \cdot u = v \\ \underline{u}_i(t) \leq u_i(t) \leq \overline{u}_i(t) \end{cases}$$

These added inequality constrains complicate the resolution of the problem and make it not resolvable by direct analytical minimization.

An important comment to be done once reached this point is that, it was considered that it is not possible to know the actual deflection of control surfaces, but every servo-actuator acts in closed loop, providing no information. This assumption may not be true in many well developed systems but is true for many small UAVs, which the theory developed here is applicable to.

With this consideration, and knowing control surfaces in the model have rate limits and physical limits, the computed set of deflection δ_i should met the inequality constrains which are defined as follows:

$$\underline{u_i}(t) = \max(u_{i\,,min}\,, u_i(t - \Delta T)\ + r_{u_i}^{-} \cdot \Delta t)$$

$$\overline{u}_i(t) = \min(u_{i\,,max}\,, u_i(t - \Delta T)\ + r_{u_i}^{+} \cdot \Delta t)$$

Where ΔT is the time step CA is being executed, r_δ^- and r_δ^+ the rate limits and $\delta_{i\,,min}$ and $\delta_{i\,,max}$ actual physical limits.

Within this problem framework, where there exists equalities and inequalities constrains, error minimization using quadratic programing (QP) techniques were chosen.

4.3.1.1 Error minimization using quadratic programing

Quadratic programing optimization methods refer to dynamic programing methods using a cost function that is a 2-norm. That is, the problem now is:

$$u = \underset{u}{\operatorname{argmin}} \sum_i \|\Psi_{\mathrm{i}}(x, u, v)\|^2$$

$$Subject\ to: \begin{cases} B \cdot u = v \\ \underline{u_i}(t) \le u_i(t) \le \overline{u}_i(t) \end{cases}$$

Being Ψ_{i} functions defined by the user.

Generally, there are two main different methods in CA depending on whether time evolutions are taken into account or not:

- Static CA: Computed control deflections at time t only depend on virtual command on same time $v(t)$

$$u(t) = \Theta\big(v(t)\big)$$

- Dynamic CA (see [40]): frequency division is performed, that is, Computed control deflections at time t are function of their time history, virtual command at same time and its time history:

$$u(t) = \Theta\left(\begin{pmatrix} \vdots \\ u(t-1\cdot\Delta T) \\ u(t) \end{pmatrix}, \begin{pmatrix} \vdots \\ v(t-1\cdot\Delta T) \\ v(t) \end{pmatrix}\right)$$

Typically, these problems are formulated in the following ways (see [40]):

Method	Typical formulation
Static CA	$J(x,u,v) = \|\mathrm{W_u}\cdot(u-u_s)\|^2$
Dynamic CA ([40]**)**	$J(x,u,v) = \|\mathrm{W_1}\cdot(u-u_s)\|^2 + \|\mathrm{W_2}\cdot(u(t)-u(t-\Delta T))\|^2$

Table 4-1Static/Dynamic CA formulations

For the particular case of this Thesis, Static CA was chosen since advanced frequency division of actuators was not the objective of the work.

Note that u_s is de desired control deflections. u_s can be chosen in many ways to comply with many objectives. For example, it can be the control deflection set that minimizes Drag, the trim deflection for the actual flight condition, etc. Here zero deflections were chosen, that is, minimize the total deflections while providing required control authority.

Finally, the problem is reformulated:

$$u = \underset{u}{\operatorname{argmin}} \|\mathrm{W_u}\cdot(u-u_s)\|^2$$

$$Subject\ to: \begin{cases} B\cdot u = v \\ \underline{u}_i(t) \le u_i(t) \le \overline{u}_i(t) \end{cases}$$

Regarding the algorithms which numerically solve the problem and handle de equality and inequality constrains, Active Set [80] is the most famous and used. But methods like Fixed-Point, Interior Point and Trust-Region-Reflective have been also used and are available. Note that some of these techniques and its variants allow the user use a gradient for the optimization, that is, gradient-based optimization.

In order to remove the equality constrain, some algorithms, as said, introduce the equality constrain as other quadratic error to minimize. Here Sequential Least Squares and Weighted Least Squares methods are presented.

WLS	$u = \underset{u}{\operatorname{argmin}} \lVert \mathrm{W_u} \cdot (u - u_s) \rVert^2 + \gamma \cdot \lVert \mathrm{W_v} \cdot (B \cdot u - v) \rVert^2$ $Subject\ to: \underline{u_i}(t) \leq u_i(t) \leq \overline{u}_i(t)$	
SLS	$u = \underset{u}{\operatorname{argmin}} \lVert \mathrm{W_u} \cdot (u - u_s) \rVert^2$ $Subject\ to: u \in \Omega$	Being Ω the subspace defined as: $\Omega = \underset{\Omega}{\operatorname{argmin}} \lVert \mathrm{W_v} \cdot (B \cdot u - v) \rVert^2$ $Subject\ to: \underline{u_i}(t) \leq u_i(t) \leq \overline{u}_i(t)$

These reformulated problems can be solved using some of the algorithms stated, but here Active Set was chosen since it one of the best algorithms managing with inequality constrains.

4.3.2 Implementation

For the purpose of this Thesis, both methods WLS and SLS were used using active set (see [80]), providing similar performance, although WLS is preferred because of its simplicity. So, the final formulations of the problem using the nomenclature of the initial problem are presented in next table:

WLS	$\delta_i = \underset{\delta_i}{\operatorname{argmin}} \lVert \mathrm{W_u} \cdot ([\delta_i] - [0]) \rVert^2 + \gamma \cdot \left\lVert \mathrm{W_v} \cdot \left(C_{M_\delta}(t) \cdot [[\delta_i] - [\delta^*]] - C_{M_v}(t) \right) \right\rVert^2$ $Subject\ to: \underline{\delta_i}(t) \leq \delta_i(t) \leq \overline{\delta}_i(t)$
SLS	$\delta_i = \underset{\delta_i}{\operatorname{argmin}} \lVert \mathrm{W_u} \cdot ([\delta_i] - [0]) \rVert^2$ $Subject\ to: \delta_i \in \Omega$ Being Ω the subspace defined as: $\Omega = \underset{\Omega}{arg\min} \left\lVert \mathrm{W_v} \cdot \left(C_{M_\delta}(t) \cdot [[\delta_i] - [\delta^*]] - C_{M_v}(t) \right) \right\rVert^2$ $Subject\ to: \underline{\delta_i}(t) \leq \delta_i(t) \leq \overline{\delta}_i(t)$

Defining constrains as follows:

$$\underline{\delta_i}(t) = \max(\delta_{i,min}, \delta_i(t - \Delta T) + r_{\delta_i}^- \cdot \Delta T)$$

$$\overline{\delta}_i(t) = \min(\delta_{i,max}, \delta_i(t - \Delta T) + r_{\delta_i}^+ \cdot \Delta T)$$

As said, rate limits were used to ensure control surface actuators could reach the commanded deflections, so the rate limits are the same limits defined in Table 3-1, for this specific aircraft:

$$r_{\delta_i}^{+} = -r_{\delta_i}^{-} = [40,40,80,80,120,25,25]$$

Where the limits $\delta_{i\,,min}$ and $\delta_{i\,,max}$ are the limits of Table 3-1.

Control weighting matrix W_u can be defined in such a way so that some control surfaces are more/less active than others, that is: every δ_i can be weighted so that some of them are allow to get further away from u_s than others. If equal importance is given to every variable, they can be scaled with their ranges.

$$W_u = \begin{pmatrix} \frac{1}{\delta_{1\,,max} - \delta_{1\,,min}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{\delta_{p\,,max} - \delta_{p\,,min}} \end{pmatrix}$$

So, using limits in Table 3-1 $W_u = diag\left(\left[\frac{1}{24}, \frac{1}{24}, \frac{1}{21.5}, \frac{1}{21.5}, \frac{1}{30}, \frac{2}{25}, \frac{2}{25}\right]\right)$

And for the virtual control weighting matrix $W_v = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, since there is no preferences among channels.

These implementations where carried out using algorithms developed by Ola Härkegard in (QCAT library [80]) and implementing them, among others, together in a Simulink block, were the user can select the Algorithm, Static/Dynamic CA and many other parameters and options.

4.4 Incremental Control Allocation

Usually, the plant to be controlled does not present linearity in its behaviour with control deflections. If any way to represents these non-linearities are available, the model would be more precise and, hence, the control of the plant.

The models to be controlled in this work, presented in section 3, are not linear in control deflections, so a Taylor expansion has to be done as said in section 4.2.5. The error of this expansion is:

$$\varepsilon \sim \mathcal{O}([\delta_i] - [\delta^*])^2$$

So, in order to minimize this error, δ^* is chosen as the previous set of deflections commanded.

Building now a discrete notation, where k is the current time step, $\delta_i^* = \delta_i^k$ the actual or last commanded deflection and δ_i^{k+1} the next commanded deflection, the coefficients model is:

$$\begin{pmatrix} C_l \\ C_m \\ C_n \end{pmatrix} = f\left(\alpha, \beta, p, q, r, \left[\delta_i^k\right]\right) + B\left(\alpha, \beta, p, q, r, \left[\delta_i^k\right]\right) \cdot \Delta\delta^k + \mathcal{O}\left(\Delta\delta^k\right)^2$$

Where $\Delta\delta^k$ is the increment of control deflections required, so that:

$$\delta_i^{k+1} = \delta_i^k + \Delta\delta_i^k \qquad \textbf{(4-19)}$$

Performing CA with (4-20) and obtaining $\Delta\delta_i^k$:

$$C_{M_\delta}^k \cdot \Delta\delta_i^k = C_{M_v}^k \qquad \textbf{(4-20)}$$

The aim of this method is that the error, when inverting and using linear CA, is smaller than if using any other method. Also, it can be said the error is bounded if CA is performed with rate limits:

$$\varepsilon \leq \left(\Delta T \cdot \sum_i \max\left(\left|r_{\delta_i}^+\right|, \left|r_{\delta_i}^-\right|\right)\right)^2 \qquad \textbf{(4-21)}$$

Note that equation (4-21) shows how the maximum error is proportional to the square of the time step $\varepsilon \sim \Delta T^2$, so that using the higher execution frequency is, the better the inversion is.

Note that if using equation (4-20) and performing CA to obtain $\Delta\delta_i^k$, constrains and desired deflection in CA algorithm have to be displaced accordingly using δ_i^k.

4.5 Reconfiguration: Fault-Tolerant approach

In order to design a reconfigurable control system, the NDI and CA should reconfigure accordingly with the parameters estimated by the Dynamic Inversion system and FDI system.

In principle, this system would provide representative information of the aircraft in the form of the set of parameters presented in section 3.3. That is, if model of section 3.3 is

reconfigured, FDI and SI should detect the new parameters that best define the new plant. These reconfiguration parameters are presented in section 3.3.1.

So, with the help of FDI and SI, NDI and CA models can be reconfigured in such a way their models are those who better represent the actual plant accordingly with SI and FDI.

FDI module provides information about control surfaces Blockages, so they are implemented in CA as a shrinkage of the physical limits to the estimated blocked position of every surface. That means, that $\delta_{i\,,min} = \delta_{i\,,max} = \delta_{i\,,Block}$

FDI module also provides information about estimated effectiveness η_i of every control surface. This information is passed directly to the model used in NDI and CA in equation (4-13).

SI module is design to perform identification of surfaces areas:

$$S_{L_{Current}}, S_{R_{current}}, S_{fin_{current}}$$

These parameters are also passed to the model used by NDI and CA.

Rest of parameters that could represent a real fault or deviation in the aircraft, $\Delta\, mass, x_{cg}, \Delta\, I_i, \Delta C_D, \Delta C_L, \Delta C_m$, are not estimated, assuming they are actually modelling errors (uncertainties) in the models used by NDI and CA.

These uncertainties are the most common issue in control systems. If a robust strategy is choose, the robustness should be as big as the uncertainty of these parameters.

In this work, an adaptive strategy is chosen, son that uncertainties of these and any other parameters can be compensated in real time, ensuring stability.

5 Reference model following and controller

In this section Reference models and controllers are design to work with NDI and CA, providing the control loop or loops to control the aircraft in the desired variables.

Since his work is oriented to a manned aircraft like F-16, the variables chosen to control the aircraft were Angle of Attack, roll rate and Sideslip. Alternatively to AoA, load factor (normal acceleration) can be chosen. That is, the pilot can control the aircraft either in AoA or load factor, roll rate and sideslip. These variables were thought to be the most typical for manned aircrafts piloting and most intuitive for a pilot. Note that, these commands of the pilot are the inputs or references of the controller, so that the controller has to ensure the aircraft tracks these inputs.

Regarding to the control in normal acceleration, it can be chosen between command acceleration normal to the airspeed or normal to the trajectory.

The main idea of implement a controller is to build control loop/loops that ensure an appropriate reference tracking. As seen in section 4.2.2 this desired dynamics and controller is needed to ensure stability of this loop. Equation (4-10) presents the basic idea for the design of the controller: a feed-forward and a proportional controller.

But, as shown in equation (4-10), the feed-forward term may not be possible to obtain directly from pilot inputs since it has not to be continuous and may contain noise, so something else has to be done. The solution is to ask to the control loop/loops to behave in a desired way or track a desired dynamics: Explicit Model Following.

Figure 5-1 shows how the desired dynamics is implemented along with a P controller and a feed-forward, achieving expression in equation (4-10) and a stable control loop.

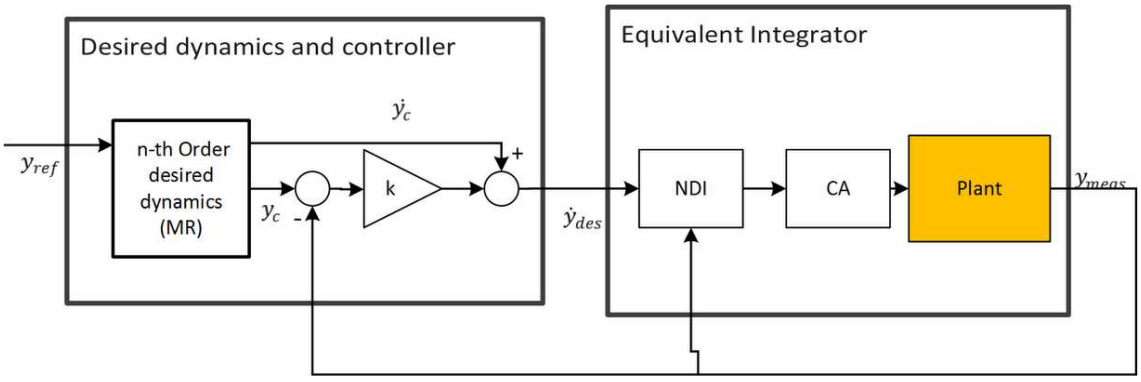


Figure 5-1 Model reference and controller

Regarding the desired dynamics or reference model, it should be design to provide the required performance, along with the gain K. Its order depends on the order of the variable to be controlled in relation with angular accelerations. That is, if controlling in pitching rate, the reference model should be at least a 1st order model and, if controlling directly AoA, the reference model should be at least of 2nd order. That is because there should exist a feed-forward if the n-th derivative of the controlled variable that correspond to the angular accelerations.

Desired dynamics and controller should be designed according with the required performance (e.g. specifications) and accordingly with the capabilities of the plant. A common technique to design is assuming, as shown in Figure 5-1, a perfect Dynamic inversion: NDI+CA+Plant are equivalent to an integrator.

In this section an introductive general theory of Reference models and controllers is presented for the cases of 1st and 2nd order reference models.

Finally, different architectures of Reference model and controllers are presented. The first architecture is based in singles reference models that assume linearity between derivatives (see [22]).

The last architecture presented is composed with several 1st order reference models assuming non-linearity between derivatives, implementing P controllers. This architecture was finally the chosen to work with adaptive NN, so only details of the final design of this architecture are provided, while only design procedure of the other architecture is shown.

5.1 General theory

In this section, methods for reference models and controllers design are presented.

5.1.1 1st Order reference model and controller

The first case is the case where the desired dynamics is a first order model with a P controller. This case can be the roll channel in which roll rate is controlled.

Assuming an input form the pilot y_{ref}, as shown in Figure 5-2, the outputs of the reference model are:

$$y_c = \frac{K_{ref}}{s + K_{ref}} \cdot y_{ref} \quad and \quad \dot{y}_c = \frac{s \cdot K_{ref}}{s + K_{ref}} \cdot y_{ref} = K_{ref} \cdot (y_{ref} - y_c) \qquad \textbf{(5-1)}$$

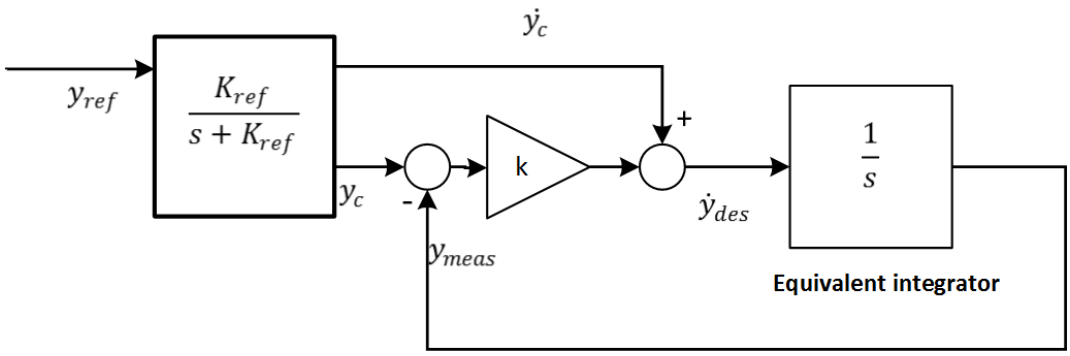


Figure 5-2 1st order MR and P controller

Where $K_{ref} = \frac{1}{\tau_{ref}}$ and τ_{ref} is the desired time constant. Note that if desired rise time (τ_{rise}) is preferred to be given for designing: $\tau_{ref} = \frac{\tau_{rise}}{\ln(9)}$

Now, using equation (4-10) and (5-1):

$$\dot{y}_{des} = K_{ref} \cdot (y_{ref} - y_c) + K \cdot (y_c - y_{meas})$$

Assuming perfect dynamic inversion $\dot{y}_{meas} = \dot{y}_{des}$

$$\dot{y}_{meas} = K_{ref} \cdot (y_{ref} - y_c) + K \cdot (y_c - y_{meas})$$

If K is chosen as $K = K_{ref}$, :

$$\dot{y}_{meas} = K_{ref} \cdot (y_{ref} - y_{meas})$$

Obtaining that the desired dynamics is achieved from the reference signal to the actual dynamics:

$$\frac{y_{meas}}{y_{ref}} = \frac{K_{ref}}{s + K_{ref}}$$

Design strategy is based in the next steps:

1. Design $K_{ref} = \frac{1}{\tau_{ref}}$ so that rise time is acceptable
2. Tune K starting with K_{ref} and, checking its performance with real simulations, decrease the value until reach the desired performance.
    a. K should not be less than 1. If $K < 1$ feed-forward in eq. (4-10) is more important than actual error, something not desired

5.1.2 2nd Order reference model and controller

Here a general 2nd order reference model and controller is presented, analysing the theoretical response. An example of where this can be applied could be to the longitudinal channel to perform control in AoA, so two loops are created: the 1st is control in angular rate and the second is in AoA.

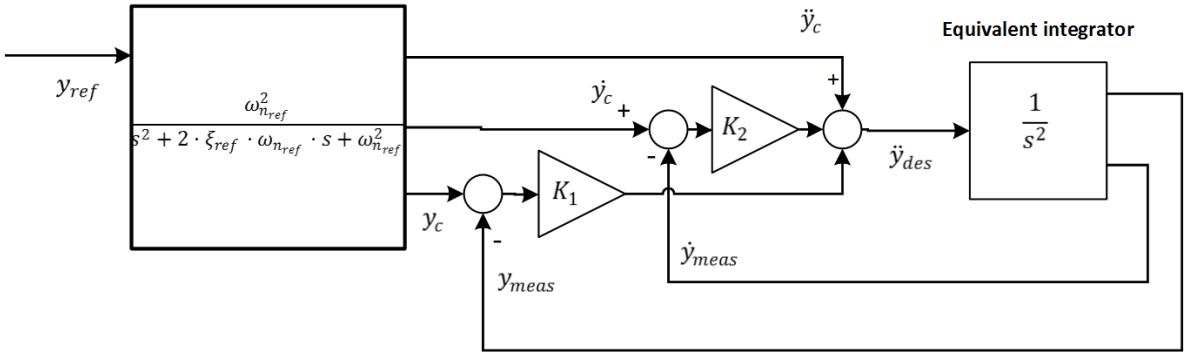


Figure 5-3 2nd order reference model and controller

The model reference is expressed as:

$$\frac{y_c}{\mathrm{y_{ref}}} = \frac{\omega_n^2}{s^2 + 2 \cdot \xi_{ref} \cdot \omega_n \cdot s + \omega_n^2} \text{ OR } \ddot{y}_c + 2 \cdot \xi_{ref} \cdot \omega_n \cdot \dot{y}_c + \omega_n^2 \cdot y_c = \omega_n^2 \cdot \mathrm{y_{ref}} \quad \textbf{(5-2)}$$

Using a formulation similar to equation (4-10) and (5-1), the controller expression:

$$\ddot{y}_{des} = \ddot{y}_c + K_2 \cdot (\dot{y}_c - \dot{y}_{meas}) + K_1 \cdot (y_c - \mathrm{y_{ref}}) \quad \textbf{(5-3)}$$

Using now equation (5-2) and assuming perfect dynamic inversion:

$$\ddot{y}_{meas} = \omega_n^2 \cdot \mathrm{y_{ref}} - 2 \cdot \xi_{ref} \cdot \omega_n \cdot \dot{y}_c - \omega_n^2 \cdot y_c + K_2 \cdot (\dot{y}_c - \dot{y}_{meas}) + K_1 \cdot (y_c - \mathrm{y_{ref}})$$

Where if controller constants are chosen $K_2 = 2 \cdot \xi_{ref} \cdot \omega_n$ and $K_1 = \omega_n^2$, it can be obtained the desired dynamic from the reference input to the measured signal:

$$\frac{y_{meas}}{\mathrm{y_{ref}}} = \frac{\omega_n^2}{s^2 + 2 \cdot \xi_{ref} \cdot \omega_n \cdot s + \omega_n^2}$$

This model reference design is usually easy to design since most of specifications about longitudinal and yaw channels (where this design is applicable) are in terms of ξ_{ref} and ω_n. Nevertheless, the designer may need to retune these parameters to achieve a desired performance. For this purpose real simulations may be needed.

Regarding gains design, the values $K_2 = 2 \cdot \xi_{ref} \cdot \omega_n$ and $K_1 = \omega_n^2$ are only indicative:

1. Start with $K_2 = 2 \cdot \xi_{ref} \cdot \omega_n$ and $K_1 = \omega_n^2$
2. Decrease these values until response converges and/or it achieves a good performance taking into account:
    a. K_1 should be always more than K_2 since tracking error is more important than its derivative, which usually contains many uncertainties
    b. Neither K_2 nor K_1 should not be less than 1. If any of they are so, feed-forward in eq. (4-10) is more important than actual error or error derivative, something not desired

This design offers an easy way to implement control in variables that are 2 orders superior to angular accelerations like AoA. But, as it can be noticed in equation (5-3), AoA rate is not an easy variable to obtain and if it is obtained, it may be noisy.

5.2 Single Reference model and P controller: Linear controller

This architecture is presented in [22], where their authors implement it with a NN as adaptive control. Here, only a description of that design is performed using the theory developed in section 5.1.

Also a stability analysis is performed, which can be used to design an Adaptive Neural Network.

5.2.1 Design

The design of this architecture is proposed [22] and is based in configure 1st and 2nd order reference models and controllers in different channels:

- Longitudinal channel: 2nd order reference model in AoA. Note that pitching angular acceleration is commanded
- Roll channel: 1st order reference model in roll rate
- Yaw channel: 2nd order reference model in sideslip. Note that yaw angular acceleration is command

Figure 5-4 shows how these designs are put together and working with NDI and CA.

Note that the commanded variables to NDI are $\begin{pmatrix} \dot{p}_{des} \\ \dot{q}_{des} \\ \dot{r}_{des} \end{pmatrix}$, that is, desired body angular accelerations.

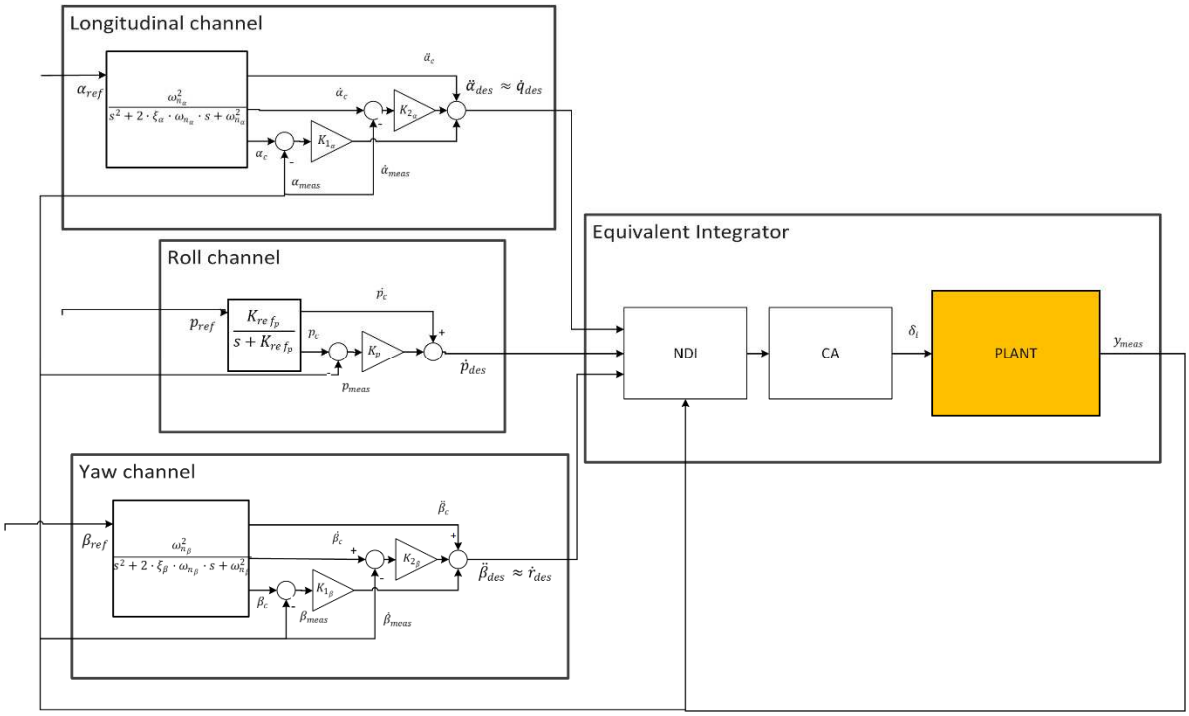


Figure 5-4 Reference models and controllers Architecture of [22]

This design of reference models and controllers does not contain much complexity but it has a fundamental disadvantage: It assumes that $\dot{\alpha}$ and $\dot{\beta}$ can be measured, something that is not possible at all in most of aircrafts. Estimations of these variables can be made using a discrete differentiation or based in GPS speed, but they are inaccurate and noisy.

As well this design assumes that there are a linear relationship between $\dot{\alpha}$ and q, and between $\dot{\beta}$ and r which is not right at all, so there exist an error when implementing these 2nd order reference models.

In section 5.3 a solution to these problems is presented.

5.2.2 Stability formulation

In this section a stability analysis of the error dynamics similar to the one performed in section 4.2.4, is carried out. The objective is to see how uncertainties on the model in NDI and CA affect to stability and enable a formulation to implement a NN.

Taking now a similar formulation to the used in section 4.2.4, being x a general expression of the dynamics and δ_i control deflections:

- Actual plant is represented by:

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \varphi(\mathrm{x}, \delta_{\mathrm{i}})$$

- Plant representation used in NDI is:

$$\begin{pmatrix} \hat{\dot{p}} \\ \hat{\dot{q}} \\ \hat{\dot{r}} \end{pmatrix} = \hat{\varphi}(\mathrm{x}, \delta_{\mathrm{i}})$$

And adding an adaptive term as shown in Figure 6-1:

$$\upsilon = \dot{y}_{des} - \upsilon_{\mathrm{ad}} = \begin{pmatrix} \dot{p}_{des} \\ \dot{q}_{des} \\ \dot{r}_{des} \end{pmatrix} - \begin{pmatrix} \upsilon_{ad_p} \\ \upsilon_{ad_q} \\ \upsilon_{ad_r} \end{pmatrix} \quad \textbf{(5-4)}$$

Where, υ is the commanded angular acceleration to NDI and υ_{ad} is the adaptive term.

Hence, inversion achieved with NDI and CA is: $\delta_{\mathrm{i}} = \hat{\varphi}^{-1}(\mathrm{x}, \upsilon)$

So that there exists an uncertainty Δ in each channel such that:

$$\Delta = \begin{pmatrix} \Delta_{\mathrm{p}} \\ \Delta_{\mathrm{q}} \\ \Delta_{\mathrm{r}} \end{pmatrix} = \varphi(\mathrm{x}, \upsilon) - \hat{\varphi}(\mathrm{x}, \upsilon)$$

Hence: $\begin{pmatrix} \dot{p}_{meas} \\ \dot{q}_{meas} \\ \dot{r}_{meas} \end{pmatrix} = \varphi(\mathrm{x}, \delta_{\mathrm{i}}) = \hat{\varphi}(\mathrm{x}, \delta_{\mathrm{i}}) + \Delta = \upsilon + \begin{pmatrix} \Delta_{\mathrm{p}} \\ \Delta_{\mathrm{q}} \\ \Delta_{\mathrm{r}} \end{pmatrix} = \begin{pmatrix} \dot{p}_{des} \\ \dot{q}_{des} \\ \dot{r}_{des} \end{pmatrix} + \begin{pmatrix} \Delta_{\mathrm{p}} \\ \Delta_{\mathrm{q}} \\ \Delta_{\mathrm{r}} \end{pmatrix} - \begin{pmatrix} \upsilon_{ad_p} \\ \upsilon_{ad_q} \\ \upsilon_{ad_r} \end{pmatrix}$

Defining now a tracking error as:

$$e = \begin{pmatrix} e_p \\ e_\alpha \\ e_{\dot{\alpha}} \\ e_\beta \\ e_{\dot{\beta}} \end{pmatrix} = \begin{pmatrix} p_c - p_{meas} \\ \alpha_c - \alpha_{meas} \\ \dot{\alpha}_c - \dot{\alpha}_{meas} \\ \beta_c - \beta_{meas} \\ \dot{\beta}_c - \dot{\beta}_{meas} \end{pmatrix} \quad \textbf{(5-5)}$$

Which are the errors that multiply each P controller in Figure 5-4.

Introducing now all the reference models and controllers laws developed in section 5.2.1, the error time derivative is:

$$\dot{e} = \begin{pmatrix} -K_p & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -K_{1_\alpha} & -K_{2_\alpha} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -K_{1_\beta} & -K_{2_\beta} \end{pmatrix} \cdot \begin{pmatrix} e_p \\ e_\alpha \\ e_{\dot{\alpha}} \\ e_\beta \\ e_{\dot{\beta}} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \upsilon_{ad_p} - \Delta_\mathrm{p} \\ \upsilon_{ad_q} - \Delta_\mathrm{q} \\ \upsilon_{ad_r} - \Delta_\mathrm{r} \end{pmatrix}$$

Obtaining the time derivative of the Lyapunov candidate functions as

$$\dot{V} = e^T \cdot A \cdot \begin{pmatrix} e_p \\ e_\alpha \\ e_{\dot{\alpha}} \\ e_\beta \\ e_{\dot{\beta}} \end{pmatrix} + e^T \cdot B \cdot \begin{pmatrix} \upsilon_{ad_p} - \Delta_\mathrm{p} \\ \upsilon_{ad_q} - \Delta_\mathrm{q} \\ \upsilon_{ad_r} - \Delta_\mathrm{r} \end{pmatrix} \quad \textbf{(5-6)}$$

It can in equation (5-6) that the first term is always negative while the second is not, unless υ_{ad} compensates Δ, that is the uncertainty of modelling error.

5.3 Non-linear multiple Reference model and P controller

The purpose of this section and the reference model and controller presented here is to eliminate the inconveniences of the previous design stated in section 5.2.1.

The way to solve this problem is to perform feedback only in the variables that can be measured and not in $\dot{\alpha}$ nor $\dot{\beta}$, and, at the same time, include the existing non-linearities that exist between the rates.

5.3.1 Design

The design proposed here only uses 1st order reference models in α, β, p, q, r, and performing a nonlinear conversion between $\dot{\alpha}$ and q and $\dot{\beta}$ and r.

So, the design in longitudinal and yaw channels is that there exist two loops, each one with a 1st order reference model and P controller.

Note that design of the outer loops is performed assuming that inner loops achieve perfect tracking. That is, inner loops have to be fast enough in comparison with outer loops, so that they achieve perfect racking much faster than outer loops.

This condition is imposed by using the criteria:

$$\tau_{ref_{out}} > 10 \cdot \tau_{ref_{inner}} \quad \textbf{(5-7)}$$

5.3.1.1 Longitudinal channel

As said, the longitudinal channel can be used to control either in AoA or normal acceleration.

5.3.1.1.1 AoA controller

The design shown in Figure 5-5 shows how there exist two control loops, one in AoA and other in body pitch rate.

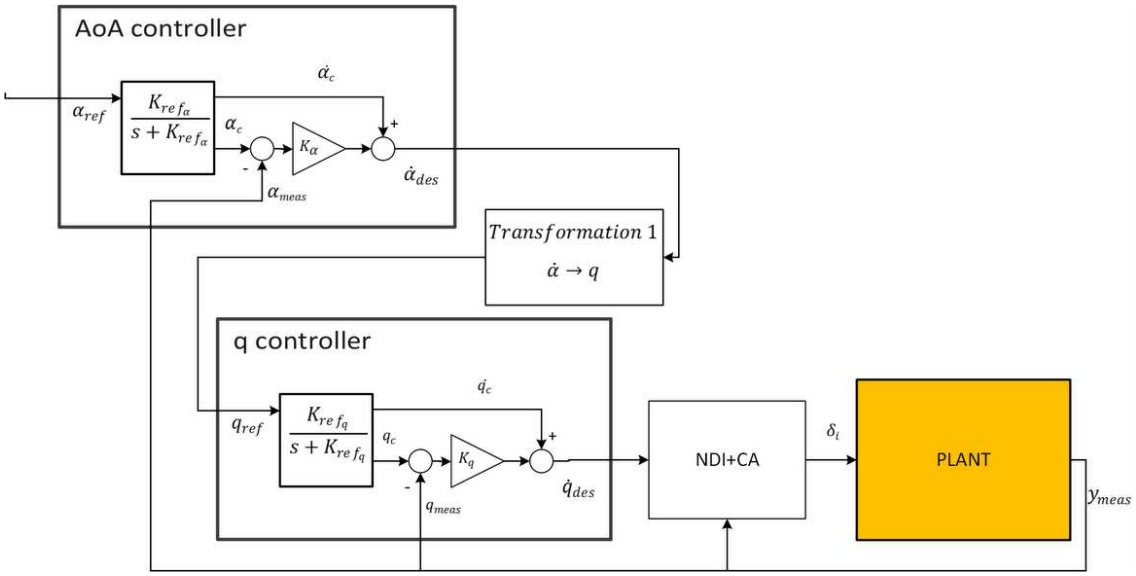


Figure 5-5 MR and Control Long AoA

Both of reference models are 1st order (section 5.1.1) and they have to meet the equation (5-7).

The key point in this design lies in the Transformation 1, that translates AoA rate into body rate:

$$q = \dot{\alpha} + \left(p \cdot \cos(\alpha) + r \cdot \sin(\alpha)\right) \cdot tan(\beta) - \frac{A_{w_z}}{V_{TAS} \cdot cos(\beta)} \qquad \textbf{(5-8)}$$

Where A_{w_z} is the component in z-axis of the acceleration of the CG in wind axes.

In equation (5-8) q is very similar to $\dot{\alpha}$, while the others two terms are usually small but important when manoeuvring far way form level flight. So, using equation (5-8) in this design increases accuracy.

5.3.1.1.2 ACC controller

A similar approach is carried out when controlling in normal acceleration.

Figure 5-6 shows the controller in normal acceleration where similar approach to the one in Figure 5-5 is taken but adding a second transformation.

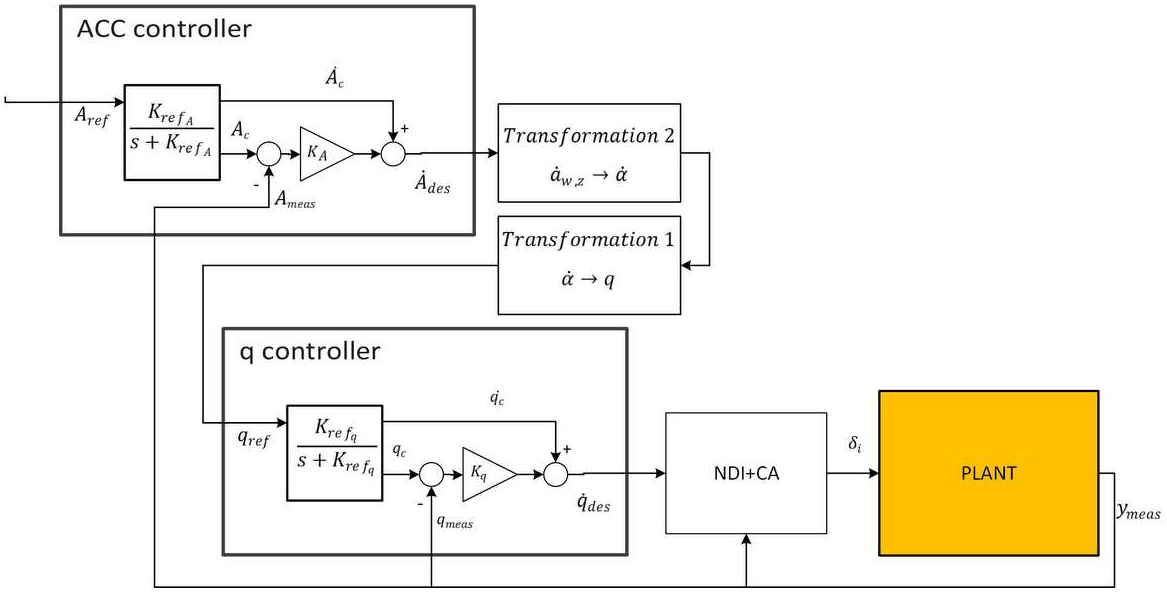


Figure 5-6 MR and Control Long ACC

The first point to explain is how to obtain normal acceleration from the dynamics.

Assuming that accelerometer sensor measures acceleration at the CG in body axes a_b and subtracting gravity:

$$A_{F_b} = T_{F_h \rightarrow F_b} \cdot \left(T_{F_b \rightarrow F_h} \cdot a_b - \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} \right)$$

Where A_{F_b} is the acceleration of the CG.

So, the acceleration in wind axes is then obtain: $A_w = T_{F_b \rightarrow F_w} \cdot A_{F_b}$

Assuming $T_{F_b \rightarrow F_w}$ is rotation matrix from body axes to wind axes. So, with that the pilot input is $A_{w_{z_{ref}}}$ and the measurement is A_{w_z}, performing control in load factor as in Figure 5-6.

In order to find the Transformation 2, relationship between $\dot{\alpha}\ and\ \dot{a}_{w_z}$ has to be found.

Assuming that g_{w_z} is the gravity component in z-axis of wind axes:

$$a_{w_z} = A_{w_z} + g_{w_z}$$

Where a_{w_z} is the component of the acceleration measured perpendicular to the velocity and in the symmetry plane, that is, the load factor n: $n = \frac{a_{w_z}}{g}$

From the lift equation: $L = \bar{q} \cdot S \cdot C_L = n \cdot W$ and $C_L = \frac{a_{w_z} \cdot m}{\bar{q} \cdot S}$

Assuming now the lift curve is an expression like: $C_L = a_0 + C_{L_\alpha} \cdot \alpha$

$$\alpha = -\frac{a_0}{C_{L_\alpha}} + \frac{a_{w_z} \cdot m}{\bar{q} \cdot S \cdot C_{L_\alpha}}$$

And deriving, second the Transformation is obtained:

$$\dot{\alpha} = \frac{m}{\bar{q} \cdot S \cdot C_{L_\alpha}} \cdot \dot{a}_{w_z} \quad \textbf{(5-9)}$$

Where C_{L_α} is taken as $C_{L_\alpha} \approx 3.58$ obtained from a linear fit of the lift curve of the model.

Assuming now $\dot{g}_{w_z}$ is negligible in comparison with $\dot{A}_{des}$:

$$\dot{\alpha}_{des} = \frac{m}{\bar{q} \cdot S \cdot C_{L_\alpha}} \cdot \dot{A}_{des}$$

With this framework, the pilot can control the aircraft in load factor.

But, as said, the pilot also can control the aircraft in normal acceleration to the trajectory. The way this is carried out is performing a translation between trajectory axes and wind axes. So, defining as A_{γ_z} as the acceleration commanded by the pilot:

$$A_w = T_{F_b \to F_w} \cdot \left(T_{F_\gamma \to F_b} \cdot \begin{pmatrix} 0 \\ 0 \\ A_{\gamma_z} \end{pmatrix} \right)$$

Obtaining A_{w_z} that is the commanded load factor and assuming $T_{F_b \to F_\gamma} = \left(T_{F_\gamma \to F_b} \right)^T$ is the rotation matrix from body axes to trajectory axes, assuming it can be determined.

With that, the pilot can control in the normal component of acceleration in wind axes.

5.3.1.2 Roll channel

Roll channel is similar to the one described in section 5.2 and its scheme can be found in Figure 5-7.

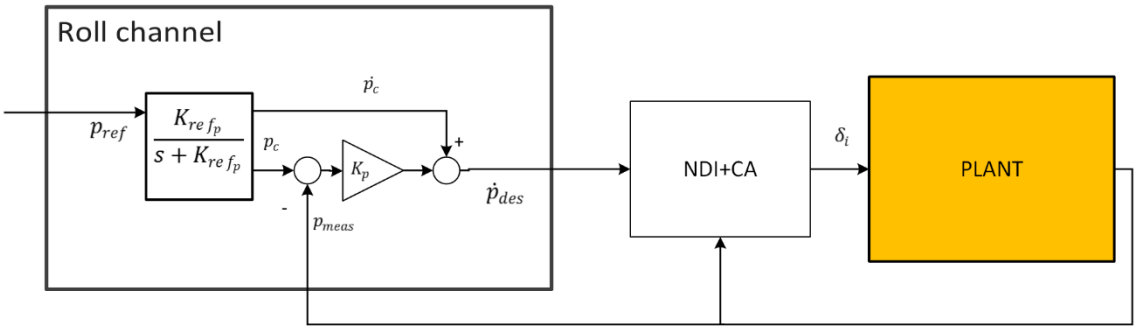


Figure 5-7 MR and control Roll

5.3.1.3 Yaw channel

Similar procedure to the performed in section 5.3.1.1 is performed here.

The controller in sideslip is based in two 1st order reference models and controllers, while having a conversion between $\dot{\beta}$ and r.

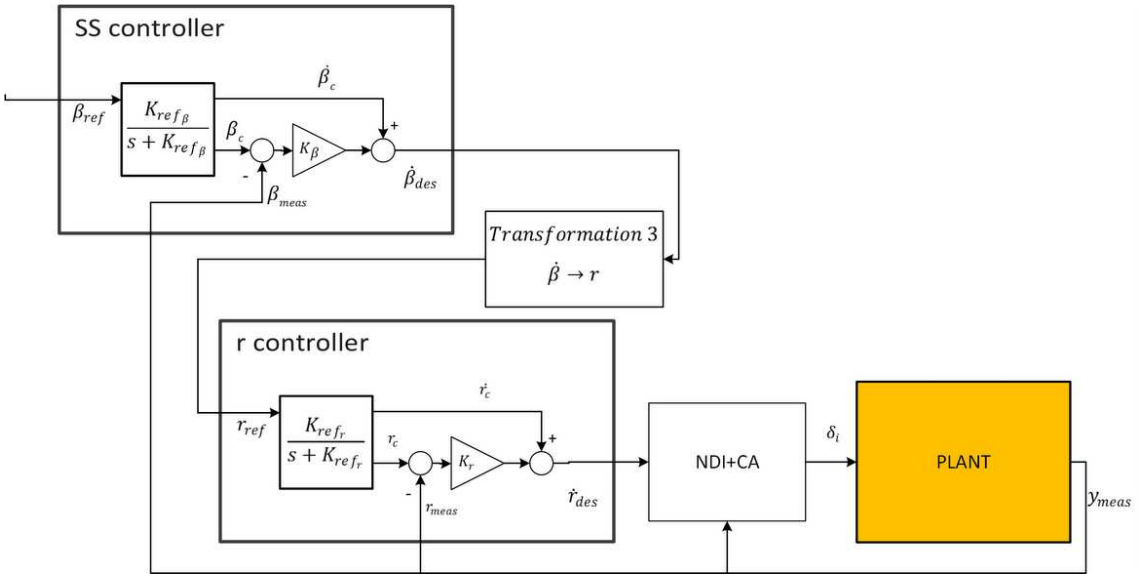


Figure 5-8 MR and Control Yaw

The Transformation 3 is defined:

$$r = -\frac{\dot{\beta}}{\cos(\alpha)} + p \cdot \tan(\alpha) + \frac{\mathrm{A_{w_y}}}{V_{TAS} \cdot \cos(\alpha)} \qquad \textbf{(5-10)}$$

Where $\mathrm{A_{w_y}}$ is the lateral acceleration of the CG in wind axes.

5.3.2 Stability formulation

In this section an analysis similar to the one performed in section 4.2.4 is performed, where also an adaptive term is added as Figure 6-1 architecture shows. This analysis reveals the effect of uncertainty and a framework for the design of adaptive NN.

As done in section 5.2.2, the actual plant is represented with $\varphi(\mathrm{x}, \delta_\mathrm{i})$ and the plant used in NDI and CA is $\widehat{\varphi}(\mathrm{x}, \delta_\mathrm{i})$.

Adding the adaptive term, as shown in Figure 6-1:

$$\upsilon = \dot{y}_{des} - \upsilon_{\mathrm{ad}} = \begin{pmatrix} \dot{p}_{des} \\ \dot{q}_{des} \\ \dot{r}_{des} \end{pmatrix} - \begin{pmatrix} \upsilon_{ad_p} \\ \upsilon_{ad_q} \\ \upsilon_{ad_r} \end{pmatrix}$$

The uncertainty Δ in each channel is:

$$\Delta = \begin{pmatrix} \Delta_\mathrm{p} \\ \Delta_\mathrm{q} \\ \Delta_\mathrm{r} \end{pmatrix} = \varphi(\mathrm{x}, \upsilon) - \widehat{\varphi}(\mathrm{x}, \upsilon)$$

And:

$$\begin{pmatrix} \dot{p}_{meas} \\ \dot{q}_{meas} \\ \dot{r}_{meas} \end{pmatrix} = \varphi(\mathrm{x}, \delta_\mathrm{i}) = \widehat{\varphi}(\mathrm{x}, \delta_\mathrm{i}) + \Delta = \upsilon + \begin{pmatrix} \Delta_\mathrm{p} \\ \Delta_\mathrm{q} \\ \Delta_\mathrm{r} \end{pmatrix} = \begin{pmatrix} \dot{p}_{des} \\ \dot{q}_{des} \\ \dot{r}_{des} \end{pmatrix} + \begin{pmatrix} \Delta_\mathrm{p} \\ \Delta_\mathrm{q} \\ \Delta_\mathrm{r} \end{pmatrix} - \begin{pmatrix} \upsilon_{ad_p} \\ \upsilon_{ad_q} \\ \upsilon_{ad_r} \end{pmatrix}$$

Defining the tracking error vector as follows:

$$e = \begin{pmatrix} e_p \\ e_\alpha \\ e_q \\ e_\beta \\ e_r \end{pmatrix} = \begin{pmatrix} p_c - p_{meas} \\ \alpha_c - \alpha_{meas} \\ q_c - q_{meas} \\ \beta_c - \beta_{meas} \\ r_c - r_{meas} \end{pmatrix}$$

In the roll channel, the error derivative, adding the controller equation:

$$\dot{e}_p = \dot{p}_c - \left(\dot{p}_c + K_p \cdot e_p + \Delta_\mathrm{p} + \upsilon_{ad_p}\right) = -K_p \cdot e_p + \left(\upsilon_{ad_p} - \Delta_\mathrm{p}\right)$$

Regarding control loop of q and r, the error derivative expressions are very similar to p:

$$\dot{e}_q = K_q \cdot e_q + \left(\upsilon_{ad_q} - \Delta_\mathrm{q}\right) \quad and \quad \dot{e}_r = K_r \cdot e_r + \left(\upsilon_{ad_r} - \Delta_\mathrm{r}\right)$$

Longitudinal and yaw channels are very similar, so only AoA channels is explained.

It is assumed equation (5-7) is used, that is, for the dynamics of the AoA loop, the reference model transient of q is negligible: $q_c \approx q_{ref}$

Defining now q_{ref} using equation (5-8) Transformation as:

$$q_{ref} = \dot{\alpha}_{des} + \xi(p, r, \alpha, \beta, V_{TAS})$$

Where $\xi(p, r, \alpha, \beta, V_{TAS})$ was assumed to be small.

Similarly, in the actual dynamics $q_{meas} = \dot{\alpha}_{meas} + \hat{\xi}(p, r, \alpha, \beta, V_{TAS})$

Being ξ and $\hat{\xi}$ both small and very similar, if not the same, and evaluated at the same dynamics, so they both cancel when doing:

$$e_q = q_{ref} - q_{meas} = \dot{\alpha}_{des} + \xi(p, r, \alpha, \beta, V_{TAS}) - \dot{\alpha}_{meas} - \hat{\xi}(p, r, \alpha, \beta, V_{TAS}) = \dot{\alpha}_{des} - \dot{\alpha}_{meas}$$

Where, introducing again the controller law:

$$e_q = \dot{\alpha}_c + K_\alpha \cdot (\alpha_c - \alpha_{meas}) - \dot{\alpha}_{meas} = K_\alpha \cdot e_\alpha + \dot{e}_\alpha$$

And finally:

$$\dot{e}_\alpha = e_q - K_\alpha \cdot e_\alpha$$

The expression of the error vector derivative can be expressed now:

$$\dot{e} = \begin{pmatrix} -K_p & 0 & 0 & 0 & 0 \\ 0 & -K_\alpha & F & 0 & 0 \\ 0 & 0 & -K_q & 0 & 0 \\ 0 & 0 & 0 & -K_\beta & F \\ 0 & 0 & 0 & 0 & -K_r \end{pmatrix} \cdot \begin{pmatrix} e_p \\ e_\alpha \\ e_q \\ e_\beta \\ e_r \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \upsilon_{ad_p} - \Delta_\mathrm{p} \\ \upsilon_{ad_q} - \Delta_\mathrm{q} \\ \upsilon_{ad_r} - \Delta_\mathrm{r} \end{pmatrix} \quad \textbf{(5-11)}$$

Where F is $F = 1$ if controlling the longitudinal channel in AoA, or $F = \frac{\bar{q} \cdot S \cdot C_{L_\alpha}}{m}$ if controlling in normal Acceleration.

Finally, the time derivative of the Lyapunov candidate function is:

$$\dot{V} = e^T \cdot \dot{e} = e^T \cdot A \cdot e + e^T \cdot B \cdot (\upsilon_{ad} - \Delta)$$

Where it can be seen that asymptotical stability cannot be ensured due to the second term, unless υ ensures it. This expression will be used again in section 6 to design the NN making υ_{ad} stabilizes the system.

5.3.3 Design example values

Table 5-1 shows the values of the controller of an implementation that was found to work well in most of the flight enveloped of the F-16.

AoA	**ACC**	**SS**	p	q **with AoA**	q **with ACC**	r
$\tau_{ref} = 0.6$	$\tau_{ref} = 2$	$\tau_{ref} = 0.8$	$\tau_{ref} = 0.25$	$\tau_{ref} = 0.06$	$\tau_{ref} = 0.2$	$\tau_{ref} = 0.08$
$K = 3.5$	$K = 1.5$	$K = 5$	$K = 7$	$K = 3.5$	$K = 2.5$	$K = 4$

Table 5-1 Example values controller design

Explicit Model Following architectures presented in this work present a major advantage respecting any other classical approach: They are based on a desired dynamics or performance. So, the designer can ask to the system to behave in the way he wants just adjusting the time constants presented in Table 5-1 which have a physical meaning, making it easy and intuitive the design process.

Values in Table 5-1 are only indicatives and are not optimized to provide good performance in all flight envelope.

Lift driven aircrafts have a particular feature that designer of their FCS should take into account. The entire behaviour of the aircraft is based on the dynamic pressure $\bar{q}$, so that the performance is dependable of this parameter: $\bar{q} = \frac{1}{2} \cdot \rho \cdot V_{TAS}^2$

Analysing equations, it can be said that for this kind of aircrafts, the mechanical characteristic time is inverse to the dynamic pressure:

$$\tau_{characteristic} \sim \frac{1}{\rho \cdot V_{TAS}^2}$$

This can be seen easily through the flight envelope, where the lower the speed and the lower the density is, the less manoeuvrable the aircraft is.

So, with that in mind, the designer cannot design the controller at any point without demonstrating stability in all flight conditions, that is, longitudinally for example, the designer cannot ask to the aircraft to be faster than it can physically be.

For this purpose, Table 5-1 gives a set of design parameters chosen at the design point in Figure 5-9: 16000m and M 0.89. This strategy ensures that the controller would be

stable in most of the flight conditions since it captures the slowest dynamics the aircraft can have.

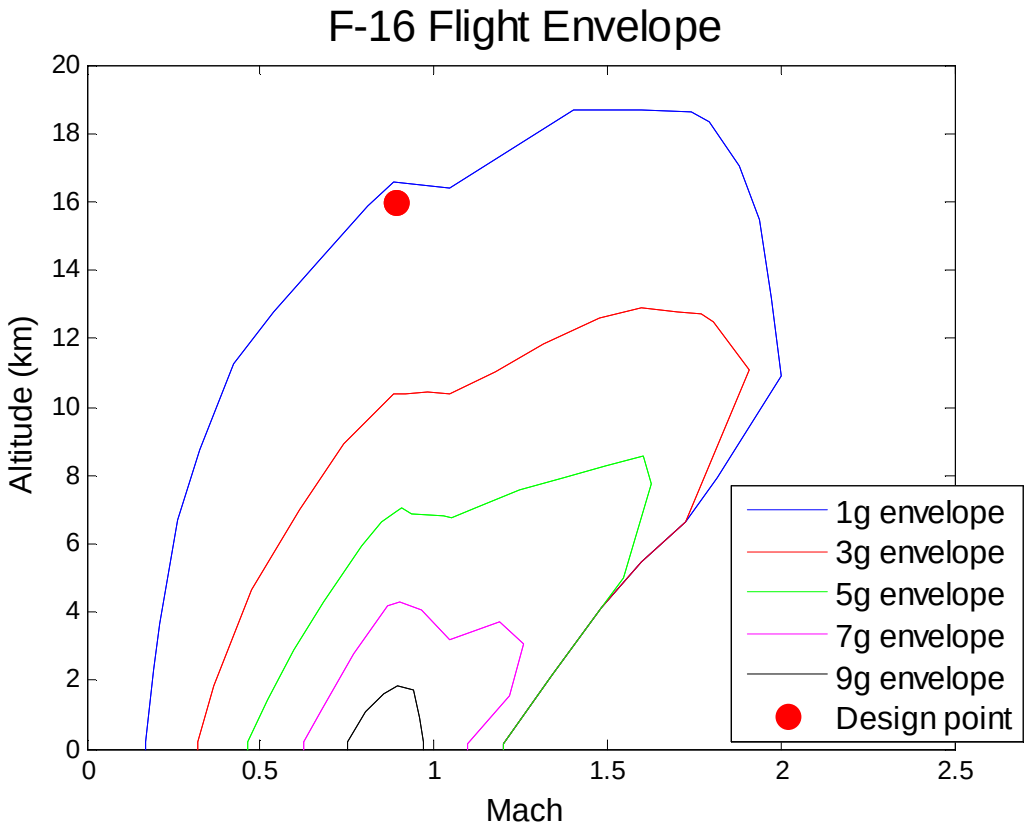


Figure 5-9 Flight envelope and example design point

But this strategy does not exploits all the potential of the aircraft at every flight condition, that is, the aircraft can be faster at highest speeds than it is at the chosen design point. Even, specifications ask to the FCS to have different performances depending of the flight condition: e.g. landing manoeuvre and fighting manoeuvres require having very different performances.

Adaptive controllers, like the presented in this work, improves performance of the controller at any flight condition, even if a single stable design of reference model is chosen, simplifying significantly the design process.

But, even when performing adaptation in flight, the designer may want to get the best possible performance for every flight condition. So, for that gain scheduling has to be performed.

5.3.4 Gain scheduling: Longitudinal channel

As explain in previous section, performance of the aircraft strongly depends on dynamic pressure. A single design can be chosen to be stable in all flight conditions, but something else can be done.

Note that gain scheduling is not necessary for the work presented, but it can be used to improve the performance further.

Longitudinal channel was found to be very sensible to the influence of dynamic pressure variation, so a Gain scheduling design was performed only for the longitudinal controllers: AoA, ACC, pitch rate.

The method used was a Particle Swarm Optimization (PSO) based on a cost function, which was designed following the next principle:

$$Cost = \begin{cases} settling_{time} + k_1 \cdot rise_{time} + k_2 \cdot overshoot & if\ response\ is\ stable \\ \infty & otherwise \end{cases}$$

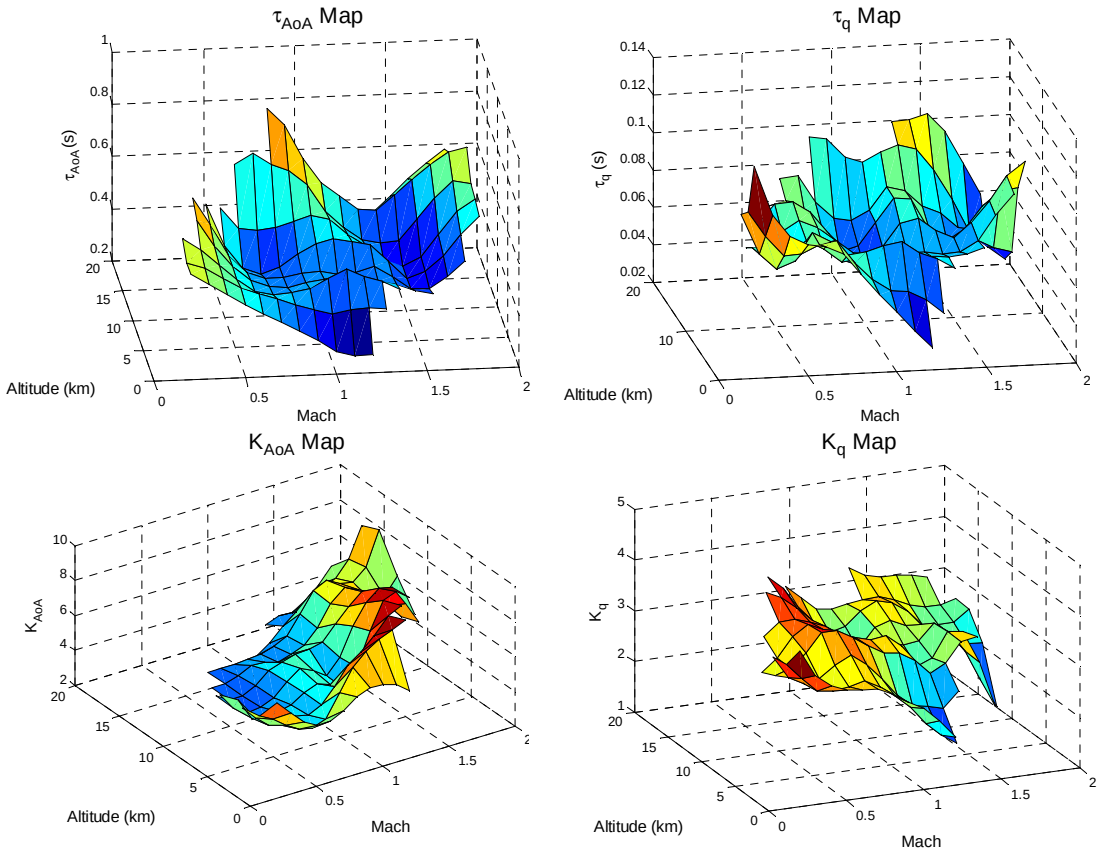


Figure 5-10 Gain scheduling design Long Channel in AoA

The idea is that the PSO not only ensures the set of parameters is stable, but they are the set that provides best performance.

Optimization was performed using steps commands in the longitudinal channel at any flight condition inside the flight envelope, building the cost function of the response.

The variables of the optimization are:$\tau_{ref_{AoA}}, K_{AoA}, \tau_{ref_q}, K_q$, and their respective parameters if normal acceleration control is desired. Note that inequalities constrains were used to implement (5-7), so that values of these maps comply this condition.

Figure 5-10 shows the four maps of the parameters of the longitudinal controller in AoA $\tau_{ref_{AoA}}, K_{AoA}, \tau_{ref_q}, K_q$ optimal for every point in the flight envelope. It can be noticed that points near stall boundary, low dynamic pressure, have bigger time constants $\tau_{ref_{AoA}}\ and\ \tau_{ref_q}$, something expected due to the analysis previously performed.

Note as well that the design point selected in section 5.3.3 is the worst point in flight envelope that can be found in maps of Figure 5-10.

6 Adaptive Neural Networks and Anti-windup

In this section Adaptive Neural Networks (Adaptive NN) are introduced to work together with the basic controllers introduced already. The controller architecture that is achieved including the Adaptive NN is the basic adaptive concept which is presented in this work.

Some of the key advantages of performing adaptive control using Adaptive NN are that the needs for a perfect model to perform dynamic inversion or controller design are relaxed. Also, due to its purpose, it simplifies the designing process by reducing the amount of work needed to analyse the dynamics and tune and validate classical control approaches, even more if failures are taken into account.

Regarding fault-tolerant approaches, Adaptive NN eliminate the needed of a SI when recovering from failures and significantly reduce the effort for SI when manoeuvring after failure.

Generally, it can be seen as well, that use a neuro-adaptive approach not only allow perform fault-tolerant control but, significantly reduces the amount of work, effort and money expended when developing and applying Flight Control Systems.

Adaptive NN are integrated in the controller framework already developed through an adaptive term as presented in Figure 6-1.

In this section, an introduction to the general architecture of NN and the way of implementing them in Control systems is performed. Secondly, several architectures of NN are presented along with their demonstrations.

Also, a general algorithm for adaptive NN of any number of layers is given.

Finally, an anti-wind up system is design and implemented along with the design of NN.

6.1 Implementing Adaptive Neural Networks

NN are easy and powerful tools when complex problems are given. In this section a general architecture neural network is implemented along with the already explained controller based on explicit model following.

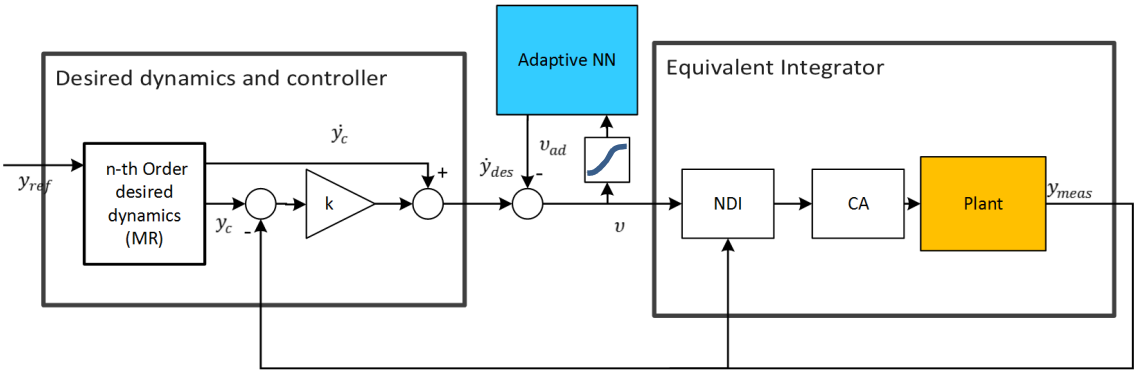


Figure 6-1 Adaptive controller scheme

Figure 6-1 shows where NN fit in the scheme of the controller. The idea is that the term v_{ad} , the output of the NN, compensates the uncertainties Δ.

NN are composed by neurons arranged in layers. Each neuron has what is called activation function $\sigma(x)$ that normally is a sigmoidal function, but depending on the desired design it can vary from layer to layer.

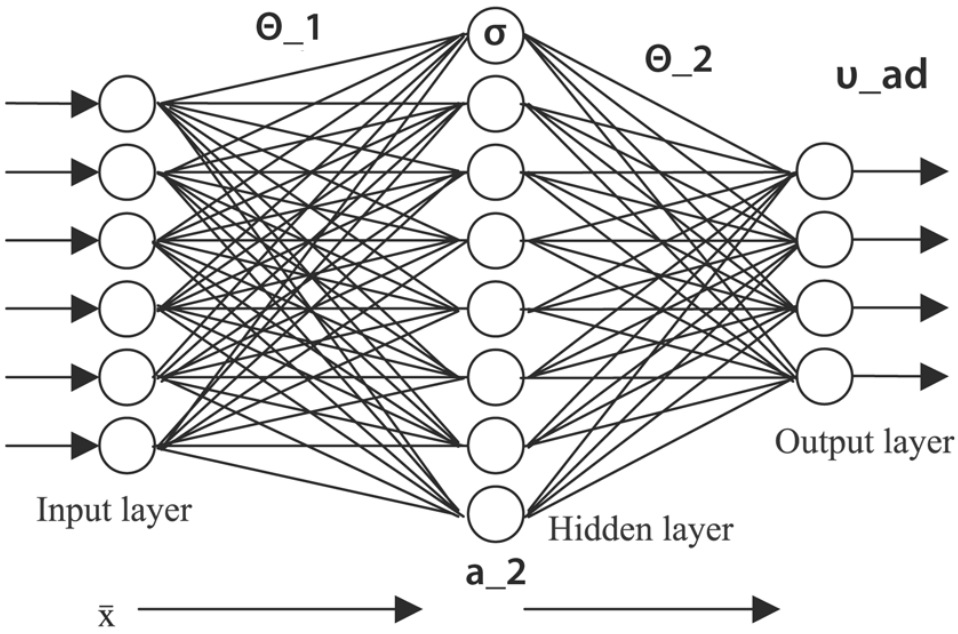


Figure 6-2 General NN

NN have a first layer that is the input layer and normally neurons in this layer do not have activation function, that is, the activation function is the unit. The size of this input layer is the size of the input variable $\bar{x}$, which is designed by the user.

The last layer is the output layer and it has the dimension of the desired output or target output. This is also design by the user.

Intermediate layers are called hidden layers and the number of them and the number of neurons on each one can be design by the user.

In principle, NN are used to identify or reproduce complicate patterns. So, the idea is that the more complex the NN is (more layers and more neurons), the more precise it would be or more complex patterns it would reproduce.

The general theory of NN establishes that given a data set of inputs and desired outputs $\{\bar{x}_m, y_m\}$ a NN can be trained to be used by inputting a $\bar{x}$ and obtaining an output y that would match the pattern in the training set. This training procedure is based in compute the error in each layer and performs an optimization routine in function of the weighting matrices Θ_i.

But, for the case of flight control systems there exist no training set. Instead the inputs are given in real flight and the desired outputs (ideally Δ) are not possible to obtain. So, Adaptive NN have to be developed.

The working principle of NN is shown in Table 6-1 where forward propagation of input is performed up to obtain the output.

Forward propagation	
L+1 Layers notation	**L Layers notation**
$a_1 = \bar{x}$ $z_1 = \Theta_1 \cdot a_1$ $for\ i = 2{:}\,L$ $\quad a_i = \sigma_{i-1}(z_{i-1})$ $\quad z_i = \Theta_i \cdot a_i$ end $v_{ad} = z_L$	$a_1 = \bar{x}$ $for\ i = 2{:}\,L$ $\quad z_i = \Theta_{i-1} \cdot a_{i-1}$ $\quad a_i = \sigma_i(z_i)$ end $v_{ad} = a_L$

Table 6-1 Forward propagation

Being the number of layers in Figure 6-1 equal to 3, note that there are two different notations: “L+1 layers” or “L layers”.

The difference is that “L+1 layers” notation considers that there are $L+1$ layers and there is not activation function at the last layer. “L layers” Notation considers the last layer has activation function.

For the given problem, regression problem, the last layer of the NN does not have activation function, that is, it is the unity. An activation function at the last layers is more suitable for classification problems.

So, most of the demonstrations in this section and expressions are given in “L+1 layers” notation, although a final algorithm is presented in both notations.

Note that every layer’s activation function can be chosen as desired. Usually:

$$\sigma(z) = \frac{1}{1+e^{-a\cdot z}} \;\; OR \;\; \sigma(z) = \frac{1-e^{-a\cdot z}}{1+e^{-a\cdot z}}$$

Where a is a scaling parameter which is usually 1.

Note that NN are very sensible to scaling problems, so that all the inputs must be scaled in a similar range.

Regarding Table 6-1 forward implementation, note that every $\Theta_i \cdot a_i$ multiplication must include a first element equals to 1, in order to keep a bias in each layer. That is, $\Theta_i \cdot a_i$ is implemented like $\Theta_i \cdot \begin{bmatrix} 1 \\ a_i \end{bmatrix}$

Fusing the attention now, on how the NN works along with the controller, let us take the stability analysis of section 5. The general expression of the tracking error derivative is:

$$\dot{e} = A \cdot e + B \cdot (v_{ad} - \Delta) \qquad \textbf{(6-1)}$$

Where A is assumed to have been designed properly, being Hurtwitz.

Hence, it can be found a positive definite matrix P using the Lyapunov equation:

$$A^T \cdot P + P \cdot A = -2 \cdot \Phi \cdot I \quad with \;\; \{\Phi \geq 1, \Phi \in \mathbb{N}\}$$

So that, the error used in the NN learning process is:

$$\xi = \mathrm{e}^{\mathrm{T}} \cdot \mathrm{P} \cdot \mathrm{B} \qquad \textbf{(6-2)}$$

In next section, it will be seen how to use this error in the adaptation laws and how to design these laws.

Regarding the architecture of NNs to implement an adaptive controller there exist two possibilities:

- Implement NN by channels
- Implement a single NN collecting the 3 channels

In the work presented both options were used, providing similar performance. Only last option is presented here since it is more efficient computationally although it increase the complexity of the problem.

Inputs used in the NN selected include the main variables of the tracking problem, defined by the reference model and controller, as well as the main parameters that influence the flight dynamics of the aircraft. Parameters selected were:

$$\bar{x}^T = [1, h, M, M^2, \alpha, \alpha^2, \beta, p, r, Roll\ chan.Vars, Long\ chan.Vars, Yaw\ chan.Vars, \upsilon]$$

$$Roll\ chan.Vars = \begin{pmatrix} e_p \\ p_c \\ \dot{p}_c \end{pmatrix}, Long\ chan.Vars = \begin{pmatrix} e_\alpha \\ e_q \\ \alpha_c \\ \dot{\alpha}_c \\ \dot{q}_c \end{pmatrix}, Yaw\ chan.Vars = \begin{pmatrix} e_\beta \\ e_r \\ \beta_c \\ \dot{\beta}_c \\ \dot{r}_c \end{pmatrix}$$

where every component of $\bar{x}^T$ is normalized. Note that υ is passed trough a sigmoidal function firstly, so a parameter has to be design properly.

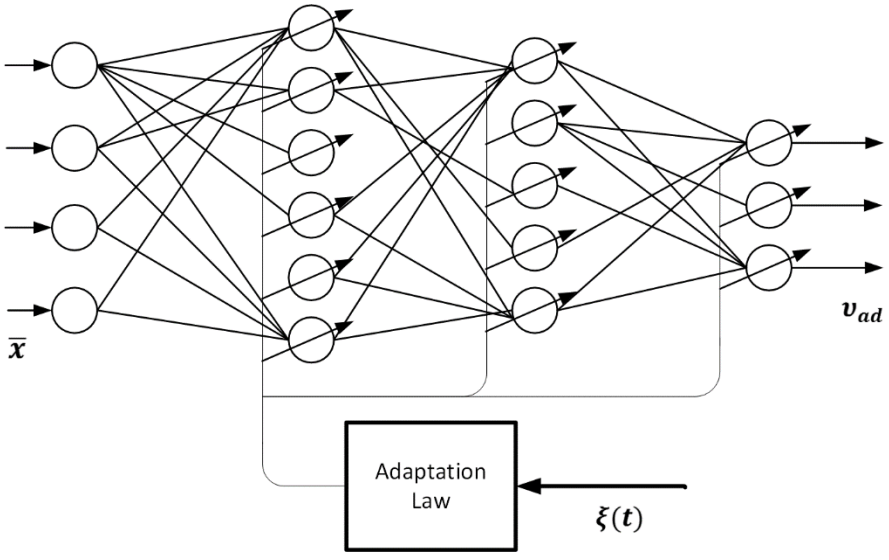


Figure 6-3 Adaptive Neural Network scheme

As seen in Table 6-1, the key parameters of the NN are the NN weights matrices Θ_i. As shown in Figure 6-3, Adaptive NN are based in estimating the NN weights matrices so that the error is minimized and, at the same time, the learning process is stable.

In next section, adaptation laws are developed for several NN architectures.

6.2 NN architectures

In this section NN adaptive control is developed, presenting several NN architectures. Note that "L+1 layers" notation is used.

Generally, for a NN of L+1 layers, there is L weights matrices that define entirely the NN: Θ_i

Taking into account the error dynamics of the controller, defined generally in equation (6-1), and taking into account the on-line learning process in which Θ_i is adapted in each time step using an adaptation law, a Lyapunov candidate Function can be built in order to check stability of the entire controller: Stability of the adaptive controller in Figure 6-1.

Defining the output of a general NN in Table 6-1 $v_{ad} = f_{NN}(\Theta_{\mathrm{i}}, \bar{x})$ and taking that the normalized input is ensured to comply $\bar{x} \in \mathcal{D}$, where $\mathcal{D}$ is a bounded domain, there exist a set of ideal weights Θ_{i}^* that:

$$v_{ad}^*(\Theta_{\mathrm{i}}^*, \bar{x}) - \Delta = f_{NN}(\Theta_{\mathrm{i}}^*, \bar{x}) - \Delta = \epsilon\,, \quad where \ \ \|\epsilon\| \leq \bar{\epsilon} \tag{6-3}$$

These ideal weight matrices can be seen as the solution of:

$$\{\Theta_{\mathrm{i}}^*\} = \underset{\{\Theta_1, \ldots \Theta_{\mathrm{L}}\}}{\operatorname{argmin}} \|\epsilon\|^2 = \underset{\{\Theta_1, \ldots \Theta_{\mathrm{L}}\}}{\operatorname{argmin}} \left\| f_{NN}(\Theta_{\mathrm{i}}, \bar{x}) - \Delta \right\|^2 \tag{6-4}$$

What is the typical method of NN training, but, as said at the introduction, Δ remains always unknown so that it cannot be used for any learning process.

Defining the error in the weight matrices $\widetilde{\Theta}_i$, that defines the convergence of the learning process, as:

$$\widetilde{\Theta}_i = \Theta_{\mathrm{i}} - \Theta_{\mathrm{i}}^* \tag{6-5}$$

The Lyapunov candidate function of the entire controller in Figure 6-1 is written:

$$V = \frac{1}{2} \cdot e^T \cdot P \cdot e + \frac{1}{2} \cdot \sum_{i=1}^{L} tr\{\widetilde{\Theta}_i \cdot \Gamma_i^{-1} \cdot \widetilde{\Theta}_i^T\} \quad \textbf{(6-6)}$$

So, adaptations laws have to ensure stability of the system under uncertainties as defined in the tracking error dynamics. In fact, adaptation laws are found when forcing $\frac{\partial V}{\partial t}$ to be negative semidefinite $\forall \bar{x} \in \mathcal{D}$.

In next sections, several architectures of NN are presented with their demonstrations, but firstly, Z is defined as:

$$Z = \begin{pmatrix} \Theta_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \Theta_L \end{pmatrix}$$

Assuming the Frobenius norm $[\![Z]\!] \leq \bar{Z}$ so that the Frobenius norm of Z is bounded and defining $\tilde{Z} = Z - Z^*$.

Note also, that the activation function derivative is defined as $\sigma_z(z) = \frac{\partial\, \sigma(z)}{\partial\, z}$.

6.2.1 2 Layers architecture

The easiest implementation of the NN implementation would be a 2 layers architecture or the called linear-in-the-parameters NN. Implementation of this architecture is shown in [17] and [46]. These implementations showed good results when approaching uncertainties in non-linear controllers.

The design is based on an expression of the output as follows:

$$v_{ad} = \Theta \cdot \sigma(\bar{x}) \quad \textbf{(6-7)}$$

So that there is only one weight matrix in the design.

The basic adaptation law is given by:

$$\dot{\Theta} = -\Gamma \cdot (\xi \cdot \sigma(\bar{x})) \quad \textbf{(6-8)}$$

Where $\Gamma > 0$ is the learning rate parameter and ξ the adaptation error of equation (6-2).

Configuring the Lyapunov candidate function as stated in equation (6-6), knowing that $\dot{\widetilde{\Theta}} = \dot{\Theta}$, the time derivative is:

$$\dot{V} = -2 \cdot \Phi \cdot e^T \cdot e + \xi \cdot (v_{ad} - \Delta) + \frac{1}{2} \cdot \left(\dot{\Theta} \cdot \Gamma^{-1} \cdot \widetilde{\Theta}^T + \widetilde{\Theta} \cdot \Gamma^{-1} \cdot \dot{\Theta} \right)$$

Substituting v_{ad}, Δ and $\dot{\Theta}$ with their expressions, noting $\Theta^* = \Theta - \widetilde{\Theta}$:

$$\dot{V} = -2 \cdot \Phi \cdot e^T \cdot e + \xi \cdot \left(\Theta \cdot \sigma(\overline{x}) - \left[\Theta - \widetilde{\Theta} \right] \cdot \sigma(\overline{x}) + \epsilon \right) - \left(\widetilde{\Theta} \cdot \left(\xi \cdot \sigma(\overline{x}) \right) \right)$$

Expanding more, the main terms of the output of the NN cancel, as well as the NN error terms, remaining:

$$\dot{V} = -2 \cdot \Phi \cdot e^T \cdot e + \xi \cdot \epsilon$$

Where it can be found a superior bound for $\dot{V}$ such that:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot \bar{\epsilon}$$

So, the following conditions, gives $\dot{V} \leq 0$, ensuring it is negative semedefinite $\forall \bar{x} \in \mathcal{D}$:

$$\Phi \cdot \|e\| > \frac{\|PB\| \cdot \bar{\epsilon}}{2} \qquad \textbf{(6-9)}$$

Letting Φ as a design parameter, that has to be designed according with equation (6-9), to ensure enough robustness in the NN learning process.

In presence of “persistent excitations” the weight of the NN may drift, so that a modification of the adaptation law presented in (6-8) is introduced:

$$\dot{\Theta} = -\Gamma \cdot (\xi \cdot \sigma(\overline{x}) + \lambda \cdot \|\xi\| \cdot \Theta) \qquad \textbf{(6-10)}$$

This added term is known as a regularization term that avoids overfiting in the NN learning process. Now, this regularization parameter $\lambda > 0$ is another design parameter.

Stability demonstration of this approach is shown in [19], [49] and many others. Here a brief outline is shown of the principles of the convergence conditions.

Introducing the modification, the Lyapunov candidate function time derivative:

$$\dot{V} = -2 \cdot \Phi \cdot \|e\|^2 + \xi \cdot \epsilon - \lambda \cdot \|\xi\| \cdot tr\{\widetilde{\mathrm{Z}} \cdot \mathrm{Z}\}$$

Using $2 \cdot tr\{A \cdot \mathrm{B}\} = [\![A + B]\!]^2 - [\![B]\!]^2 - [\![A]\!]^2 \geq -[\![B]\!]^2 - [\![A]\!]^2$

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot \bar{\epsilon} + \lambda \cdot \|\xi\| \cdot \left([\![\widetilde{\mathrm{Z}}]\!]^2 + \overline{\mathrm{Z}}^2 \right)$$

Finally the convergence condition:

$$\Phi \cdot \|e\| > \frac{\|PB\| \cdot \left(\bar{\epsilon} + \lambda \cdot \left([\![\tilde{\mathrm{Z}}]\!]^2 + \overline{\mathrm{Z}}^2 \right) \right)}{2}$$

As seen, regularization parameter λ is a design parameter that has to be chosen as a trade-off between robustness of the learning process and overfiting reduction.

$[\![\tilde{\mathrm{Z}}]\!]^2$ is an error square that, for designing purpose, it can be thought to be bounded $[\![\tilde{\mathrm{Z}}]\!] \leq 2 \cdot \overline{\mathrm{Z}}$, so that:

$$\Phi \cdot \|e\| > \frac{\|PB\| \cdot \left(\bar{\epsilon} + \lambda \cdot 5 \cdot \overline{\mathrm{Z}}^2 \right)}{2} \qquad \textbf{(6-11)}$$

Note that λ regularizes the weights matrix, so that, they get move away of Θ^*, slowing down the convergence. That is the reason because λ make the minimum Φ to increase. But this regularization term is not bad at all since it avoids the NN gets to the perfect solution Θ^* (overfitting) provoking the error to have an inferior limit $\|e\| \geq \underline{e}$, and helping to comply with (6-11). So, at the end the design of λ and is Φ a trade-off between performance and robustness.

6.2.2 3 Layers architecture

A more complex architecture than the presented in previous section, presented by many authors at [22], [21], [49], [48], [34] and many others, presents better performance when compensating uncertainties. Note that some of the mentioned publications include detailed demonstrations of stability of the solution here presented, while here only a brief demonstration is performed.

The architecture is based in a 3 layers NN, that is, a single hidden layer.

The expression that gives the output of the NN is:

$$v_{ad} = \Theta_2 \cdot \sigma(\Theta_1 \cdot \bar{x}) \qquad \textbf{(6-12)}$$

Being Θ_1 the weight matrix that passes the input a_1 to the second layer, and Θ_2 the weight matrix that passes the output of the second layer to the final one, that is, the output.

With this architecture, several approaches have been developed in the literature, depending on if it includes regularization or not. A. Calise presents in [22] same architecture but including two robustifying terms.

Note the following definitions for demonstrations in this section $\sigma = \sigma(\Theta_1 \cdot \bar{x})$ and $\sigma_z = \sigma_z(\Theta_1 \cdot \bar{x})$.

The simplest approach does not include regularization term and is given by next expressions:

Approach 1
$v_{ad} = \Theta_2 \cdot \sigma(\Theta_1 \cdot \bar{x})$
$\dot{\Theta}_2 = -\Gamma_2 \cdot \{\xi \cdot \sigma - \xi \cdot \sigma_z \cdot (\Theta_1 \cdot \bar{x})\}$
$\dot{\Theta}_1 = -\Gamma_1 \cdot \{\Theta_2^T \cdot \xi \cdot \sigma_z \cdot \bar{x}\}$
With: $\Gamma_2 > 0$, $\Gamma_1 > 0$, $\lambda > 0$

Table 6-2 3 Layers adaptive NN. No regularization

Table 6-2 presents adaptation laws that are derived from a Lyapunov stability analysis, as done in section 6.2.1.

The Lyapunov candidate function is defined:

$$V = \frac{1}{2} \cdot e^T \cdot P \cdot e + \frac{1}{2} \cdot tr\left\{\tilde{\Theta}_1 \cdot \Gamma_1^{-1} \cdot \tilde{\Theta}_1^{\mathrm{T}}\right\} + \frac{1}{2} \cdot tr\{\tilde{\Theta}_2 \cdot \Gamma_2^{-1} \cdot \tilde{\Theta}_2^{\mathrm{T}}\}$$

Error derivative is expressed, using (6-5) and (6-12):

$$\dot{e} = A \cdot e + B \cdot \left(\Theta_2 \cdot \sigma(\Theta_1 \cdot \bar{x}) - \left[\Theta_2 - \tilde{\Theta}_2\right] \cdot \sigma\left(\Theta_1^* \cdot \bar{x}\right) + \epsilon\right)$$

Now, the function $\sigma(\Theta_1^* \cdot \bar{x})$ is represented with a Taylor expansion with respect to Θ_1^* such that:

$$\sigma(\Theta_1^* \cdot \bar{x}) = \sigma(\Theta_1 \cdot \bar{x}) + \sigma_z(\Theta_1 \cdot \bar{x}) \cdot (\Theta_1^* - \Theta_1) \cdot \bar{x} + \mathcal{O}\left(\tilde{\Theta}_1 \cdot \bar{x}\right)^2$$

So that the error derivative is:

$$\dot{e} = A \cdot e + B \cdot \left(\tilde{\Theta}_2 \cdot [\sigma - \sigma_z \cdot \Theta_1 \cdot \bar{x}] + \Theta_2 \cdot \sigma_z \cdot \tilde{\Theta}_1 \cdot \bar{x} + \epsilon + \omega\right)$$

Being the error $\omega = \Theta_2^* \cdot \mathcal{O}\left(\widetilde{\Theta}_1 \cdot \bar{x}\right)^2 + \widetilde{\Theta}_2 \cdot \sigma_z \cdot \Theta_1^* \cdot \bar{x}$

Using this expression, the Lyapunov candidate function time derivative is:

$$\dot{V} = -2 \cdot \Phi \cdot e^T \cdot e + \xi \cdot \left(\widetilde{\Theta}_2 \cdot [\sigma - \sigma_z \cdot \Theta_1 \cdot \bar{x}] + \Theta_2 \cdot \sigma_z \cdot \widetilde{\Theta}_1 \cdot \bar{x} + \epsilon + \omega\right) + \frac{1}{2} \cdot \left(\dot{\Theta}_1 \cdot \Gamma_1^{-1} \cdot \widetilde{\Theta}_1{}^T + \widetilde{\Theta}_1 \cdot \Gamma_1^{-1} \cdot \dot{\Theta}_1\right) + \frac{1}{2} \cdot \left(\dot{\Theta}_2 \cdot \Gamma_2^{-1} \cdot \widetilde{\Theta}_2{}^T + \widetilde{\Theta}_2 \cdot \Gamma_2^{-1} \cdot \dot{\Theta}_2\right)$$

Which using now the adaptation laws gives:

$$\dot{V} = -2 \cdot \Phi \cdot e^T \cdot e + \xi \cdot (\epsilon + \omega)$$

Now, using upper bounds:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot (\bar{\epsilon} + \|\omega\|)$$

Where it can be defined an upper bound of the errors such that $\bar{\epsilon} + \|\omega\| \leq c_0 + c_1 \cdot [\![\tilde{Z}]\!]$. Where c_0 and c_1 depend on the number of layers and neurons.

So, the next condition ensures $\dot{V}$ is negative semidefinite:

$$\Phi \cdot \|e\| \geq \frac{\|PB\| \cdot \left(c_0 + c_1 \cdot [\![\tilde{Z}]\!]\right)}{2}$$

As previously, for design purposes $[\![\tilde{Z}]\!]$ can be thought to be bounded. Using $[\![\tilde{Z}]\!] = [\![Z + (-Z^*)]\!]$ and the Frobenius triangular inequality:

$$[\![\tilde{Z}]\!]^2 = [\![Z + (-Z^*)]\!]^2 \leq ([\![Z]\!] + [\![Z^*]\!])^2 \leq (\bar{Z} + [\![Z^*]\!])^2 \leq 4 \cdot \bar{Z}^2$$

$$\Phi \cdot \|e\| \geq \frac{\|PB\| \cdot (c_0 + c_1 \cdot 2 \cdot \bar{Z})}{2}$$

Approach 2
$\upsilon_{ad} = \Theta_2 \cdot \sigma(\Theta_1 \cdot \bar{x})$
$\dot{\Theta}_2 = -\Gamma_2 \cdot \{\xi \cdot \sigma - \xi \cdot \sigma_z \cdot (\Theta_1 \cdot \bar{x}) + \lambda \cdot \|\xi\| \cdot \Theta_2\}$
$\dot{\Theta}_1 = -\Gamma_1 \cdot \{\Theta_2^T \cdot \xi \cdot \sigma_z \cdot \bar{x} + \lambda \cdot \|\xi\| \cdot \Theta_1\}$
With: $\Gamma_2 > 0$, $\Gamma_1 > 0$, $\lambda > 0$

Table 6-3 3 Layers NN Regularization

The second approach, Table 6-3, implements a regularization term that avoids overfitting. Similarly to the solution in section 6.2.1, it consists on introduce in the adaptation law a term proportional to the weight matrix, what tries to minimize these weights, preventing this problem.

Note that this stability analysis of this architecture is presented [48] and [55].

Doing a similar procedure as before, the time derivative of the Lyapunov candidate function is:

$$\dot{V} = -2 \cdot \Phi \cdot \|e\|^2 + \xi \cdot (\epsilon + \omega) - \lambda \cdot \|\xi\| \cdot tr\{\tilde{Z}^T \cdot Z\}$$

Which, using they adequate inequalities, gives:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|e\| \cdot \|PB\| \cdot \left(c_0 + c_1 \cdot [\![\tilde{\mathrm{Z}}]\!]\right) + \lambda \cdot \|e\| \cdot \|PB\| \cdot \left([\![\tilde{\mathrm{Z}}]\!]^2 + \overline{\mathrm{Z}}^2\right)$$

So, $\dot{V}$ is negative semidefinite if either:

- Robustness of the design can be used to compensate the learning errors:

$$\Phi \cdot \|e\| \geq \frac{\|PB\| \cdot \left(c_0 + \lambda \cdot \overline{\mathrm{Z}}^2 + \lambda \cdot [\![\tilde{\mathrm{Z}}]\!]^2 + c_1 \cdot [\![\tilde{\mathrm{Z}}]\!]\right)}{2}$$

  Remember that for design, $[\![\tilde{\mathrm{Z}}]\!] < 2 \cdot \overline{Z}$ can be used.

- Or page attention only in the errors and regularization terms, such that:

$$[\![\tilde{\mathrm{Z}}]\!] > \frac{c_1}{2 \cdot \lambda} + \sqrt{\frac{c_1^2}{4 \cdot \lambda^2} - \frac{c_0}{\lambda} - \overline{\mathrm{Z}}^2}$$

The last approach is presented and its stability demonstrated in [22]. It consists in the same approach as the already presented, but introducing two robustifying terms:

Approach 3
$\upsilon_{ad} = \Theta_2 \cdot \sigma(\Theta_1 \cdot \bar{x})$
$\dot{\Theta}_2 = -\Gamma_2 \cdot \{\xi \cdot \sigma - \xi \cdot \sigma_z \cdot (\Theta_1 \cdot \bar{x}) + \lambda \cdot \|\xi\| \cdot \Theta_2\}$
$\dot{\Theta}_1 = -\Gamma_2 \cdot \{\Theta_2^T \cdot \xi \cdot \sigma_z \cdot \bar{x} + \lambda \cdot \|\xi\| \cdot \Theta_1\}$
$\upsilon_r = \upsilon_{r_0} + \upsilon_{r_1}$

$v_{r_0} = -K_{r_0} \cdot \xi^T$
$v_{r_1} = -K_{r_1} \cdot (\llbracket Z \rrbracket + \bar{Z}) \cdot \xi^T$
With: $\Gamma_2 > 0$, $\Gamma_1 > 0$, $\lambda > 0$, $K_{r_1} > 0$, $K_{r_0} > 0$
And $\hat{v} = v_r + v_{ad}$ The actual output of the NN

Table 6-4 3 layers NN robust terms

These robustifying terms have a direct impact in stability and have to be designed to achieve stability.

In a similar way as done before, the time derivative of the Lyapunov candidate function is:

$$\dot{V} = -2 \cdot \Phi \cdot \|e\|^2 + \xi \cdot (\epsilon + \omega + v_r) - \lambda \cdot \|\xi\| \cdot tr\{\tilde{Z}^T \cdot Z\}$$

And using the appropriate inequalities:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot (\overline{\epsilon} + \|\omega\|) - \left(K_{r_0} + K_{r_1} \cdot (\llbracket Z \rrbracket + \bar{Z})\right) \cdot \|\xi\|^2 + \lambda \cdot \|\xi\| \cdot \left(\llbracket \tilde{\mathrm{Z}} \rrbracket^2 + \bar{\mathrm{Z}}^2\right)$$

So that $\dot{V}$ is negative semidefinite if:

$$\|\xi\| \geq \frac{\lambda \cdot \llbracket \tilde{\mathrm{Z}} \rrbracket^2 + c_1 \cdot \llbracket \tilde{\mathrm{Z}} \rrbracket + c_0 + \lambda \cdot \bar{\mathrm{Z}}^2}{K_{r_0} + 2 \cdot K_{r_1} \cdot \bar{\mathrm{Z}}}$$

As seed, the robustifying terms help when compensating any error in the NN learning process. Note that regularization term ensures there is a minimum $\llbracket \tilde{\mathrm{Z}} \rrbracket$ and thus, a minimum $\|\xi\|$, depending on λ.

Note that, for the work presented in this thesis, this last approach was chosen.

6.2.3 General architecture: Proposed methods

Once several architectures have been developed, in this section a general architecture is proposed along with its stability analysis. In fact, the adaptation law was obtained from a Lyapunov stability analysis for a general NN as the one presented in Table 6-1.

Note that theory about Adaptive NN has been found for the already presented solutions, but not for the general architecture presented here.

Firstly, Table 6-1 establishes the general algorithm to obtain the output of a general NN of L or L+1 layers (depending on the notation) that depends on the weight matrices Θ_i. Note that here "L+1 layers" notation is used here and that $\sigma_i' = \frac{\partial\, \sigma_i(z)}{\partial\, z}$.

The adaptation laws, for any NN can be expressed as follows:

Adaptation laws for a general adaptive NN: L+1 Layers notation
$\delta_L = \psi_L = \xi$
$for\ i = L-1{:}\,1$
$\quad \delta_i = \Theta_{i+1}^T \cdot \delta_{i+1} \cdot \sigma_i'(z_i)$
$\quad \psi_i = \sigma_i'(z_i) \cdot (\psi_{i+1})^T$
end
$\dot{\Theta}_L = -\Gamma_L \cdot [\xi \cdot a_L^T - \sum_{i=1}^{L-1}(\psi_i \cdot z_i)^T + \lambda \cdot \|\xi\| \cdot \Theta_L]$
$for\ i = 1{:}\,L-1$
$\quad \dot{\Theta}_i = -\Gamma_i \cdot [\delta_i \cdot a_i^T + \lambda \cdot \|\xi\| \cdot \Theta_i]$
end

Table 6-5 Adaptation laws general NN L+1 notation

Adaptation laws for a general adaptive NN: L layers notation
$\delta_L = \psi_L = \xi$
$for\ i = L-1{:}\,2$
$\quad \delta_i = \Theta_i^T \cdot \delta_{i+1} \cdot \sigma_i'(z_i)$
$\quad \psi_i = \sigma_i'(z_i) \cdot (\psi_{i+1})^T$
end
$\dot{\Theta}_{L-1} = -\Gamma_{L-1} \cdot [\xi \cdot a_{L-1}^T - \sum_{i=2}^{L-1}(\psi_i \cdot z_{i+1})^T + \lambda \cdot \|\xi\| \cdot \Theta_{L-1}]$
$for\ i = 1{:}\,L-2$
$\quad \dot{\Theta}_i = -\Gamma_i \cdot [\delta_{i+1} \cdot a_i^T + \lambda \cdot \|\xi\| \cdot \Theta_i]$
end

Table 6-6 Adaptation laws general NN L notation

Note as well that for "L+1 layers" notation $\sigma_i(z)$ is the activation function of the $i+1th$ layer counting the input layer as the 1st layer, that is, in "L layers" notation.

In these algorithms, note that the products $\psi_i \cdot z_{i+1}$ and $\delta_{i+1} \cdot \sigma_i'(z_i)$ are colum-wise products.

In order to obtain these adaptation laws, proving their stability, a Lyapunov candidate function as expressed in equation (6-6) is proposed.

As well, taking a general expression of the error derivative as in equation (6-1) with a general expression of the NN output as $f_{NN}(\Theta_{\mathrm{i}}, \bar{x})$:

$$\dot{e} = A \cdot e + B \cdot \left(f_{NN}(\Theta_{\mathrm{i}}, \bar{x}) - f_{NN}(\Theta_{\mathrm{i}}^*, \bar{x}) + \epsilon\right)$$

Using the expressions in Table 6-1 ("L+1 layers" notation), the equation can be rewritten:

$$\dot{e} = A \cdot e + B \cdot \left(\Theta_{\mathrm{L}} \cdot a_L(\Theta_{\mathrm{i}}, \bar{x}) - \left[\Theta_{\mathrm{L}} - \widetilde{\Theta}_L\right] \cdot a_L(\Theta_{\mathrm{i}}^*, \bar{x}) + \epsilon\right)$$

Now, the term of the ideal output can expressed as a Taylor expansion:

$$a_L(\Theta_{\mathrm{i}}^*, \bar{x}) = a_L(\Theta_{\mathrm{i}}, \bar{x}) + \sum_{i=1}^{L-1} \left.\frac{\partial\, a_L(\Theta_{\mathrm{i}}, \bar{x})}{\partial\, \Theta_{\mathrm{i}}}\right|_{\substack{\Theta_{\mathrm{i}}=\Theta_{\mathrm{i}} \\ \bar{x}=\bar{x}}} \cdot (\Theta_{\mathrm{i}}^* - \Theta_{\mathrm{i}}) + \mathcal{O}\left(\left[\!\left[\tilde{Z}\right]\!\right]\right)^2$$

And using this expansion in the error dynamics equation:

$$\dot{e} = A \cdot e + B \cdot \left(\widetilde{\Theta}_L \cdot \left(a_L(\Theta_{\mathrm{i}}, \bar{x}) - \sum_{i=1}^{L-1} \left.\frac{\partial\, a_L(\Theta_{\mathrm{i}}, \bar{x})}{\partial\, \Theta_{\mathrm{i}}}\right|_{\substack{\Theta_{\mathrm{i}}=\Theta_{\mathrm{i}} \\ \bar{x}=\bar{x}}} \cdot \Theta_{\mathrm{i}} \right) - \Theta_{\mathrm{L}} \cdot \sum_{i=1}^{L-1} \left.\frac{\partial\, a_L(\Theta_{\mathrm{i}}, \bar{x})}{\partial\, \Theta_{\mathrm{i}}}\right|_{\substack{\Theta_{\mathrm{i}}=\Theta_{\mathrm{i}} \\ \bar{x}=\bar{x}}} \cdot \widetilde{\Theta}_i + \epsilon + \omega \right) \quad \textbf{(6-13)}$$

with $\omega = \widetilde{\Theta}_L \cdot \sum_{i=1}^{L-1} \left.\frac{\partial\, a_L(\Theta_{\mathrm{i}}, \bar{x})}{\partial\, \Theta_{\mathrm{i}}}\right|_{\substack{\Theta_{\mathrm{i}}=\Theta_{\mathrm{i}} \\ \bar{x}=\bar{x}}} \cdot \Theta_{\mathrm{i}}^* + \mathcal{O}\left(\left[\!\left[\tilde{Z}\right]\!\right]\right)^2$

Now, using (3-12) and the proposed adaptation laws, the time derivative of the Lyapunov candidate function in (6-6) can be expressed:

$$\dot{V} = -2 \cdot \Phi \cdot \|e\|^2 + \xi \cdot (\epsilon + \omega) - \lambda \cdot \|\xi\| \cdot tr\{\tilde{Z}^T \cdot Z\}$$

where $\|\omega\| \leq \mathrm{c}_0 + c_1 \cdot \left[\!\left[\tilde{Z}\right]\!\right] + c_2 \cdot \left[\!\left[\tilde{Z}\right]\!\right]^2$, being c_i dependent of the number of layers and number of neurons.

Using the presented inequalities:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot \left(\overline{\epsilon} + c_0 + c_1 \cdot [\![\tilde{Z}]\!] + c_2 \cdot [\![\tilde{Z}]\!]^2\right) + \lambda \cdot \|\xi\| \cdot \left([\![\tilde{Z}]\!]^2 + \overline{Z}^2\right)$$

So, either of the following expressions ensures $\dot{V}$ to be negative semidefinite:

$$\Phi \cdot \|e\| \geq \frac{\|PB\|}{2} \cdot \left[(\overline{\epsilon} + c_0) + c_1 \cdot [\![\tilde{Z}]\!] + (c_2 + \lambda) \cdot [\![\tilde{Z}]\!]^2 + \lambda \cdot \overline{Z}^2\right]$$

$$[\![\tilde{Z}]\!] \geq \frac{c_1 + \sqrt{c_1^2 - 4 \cdot (c_2 + \lambda) \cdot \left((\overline{\epsilon} + c_0) + \lambda \cdot \overline{Z}^2\right)}}{2 \cdot (c_2 + \lambda)} \quad \textbf{(6-14)}$$

Where λ plays a fundamental part as a design parameter, since this regularization term provokes a drifting of weight matrices to the origin, providing a minimum $\|e\|$ and $[\![\tilde{Z}]\!]$ that gives robustness but, at the same time, limits performance slowing down the problem (6-4).

Note that the design parameters Φ and λ are a trade-off between robustness and stability.

Similarly to section 6.2.2, robustness terms can be added in order to increase the robustness of the convergence of the learning process.

Adding robustifying terms
$\hat{v} = v_r + v_{ad}$
$v_r = v_{r_0} + v_{r_1}$
$v_{r_0} = -K_{r_0} \cdot \xi^T$
$v_{r_1} = -K_{r_1} \cdot ([\![Z]\!] + \bar{Z}) \cdot \xi^T$
With: $K_{r_1} > 0$, $K_{r_0} > 0$

Table 6-7 General NN with robust. Terms

Adding these terms, as shown in Table 6-7, and using the adaptation laws as done before, the time derivative of the Lyapunov candidate function:

$$\dot{V} \leq -2 \cdot \Phi \cdot \|e\|^2 + \|\xi\| \cdot \left(\overline{\epsilon} + c_0 + c_1 \cdot [\![\tilde{Z}]\!] + c_2 \cdot [\![\tilde{Z}]\!]^2\right) + \lambda \cdot \|\xi\| \cdot \left([\![\tilde{Z}]\!]^2 + \overline{Z}^2\right) - K_{r_0} \cdot \|\xi\|^2 - K_{r_1} \cdot ([\![Z]\!] + \bar{Z}) \cdot \|\xi\|^2 \quad \textbf{(6-15)}$$

Ignoring now in equation (6-15) the robustness provided by the term $-2 \cdot \Phi \cdot \|e\|^2$, stability can be entirely provided by the robustifying terms:

$$\|\xi\| \geq \frac{\overline{\epsilon} + c_0 + c_1 \cdot [\![\tilde{Z}]\!] + (c_2 + \lambda) \cdot [\![\tilde{Z}]\!]^2 + \lambda \cdot \overline{Z}^2}{K_{r_0} + 2 \cdot K_{r_1} \cdot \bar{Z}} \quad \textbf{(6-16)}$$

Equation (6-16) ensures $\dot{V}$ to be negative semidefinite, so that $[\![\tilde{Z}]\!]$ and $\|e\|$ remain bounded.

6.3 Anti-windup implementation

Theory explained in this section covers most of the proposed solutions in literature for adaptive NN, including a general theory for a general L layers NN.

In terms of control actions or in terms of classical control, Adaptive NN can be thought as integral action. That is, if for example a linear-in-the-parameters NN is chosen as in equation (6-7) without regularization, the adaptation term in function of time would be:

$$v_{ad}(\mathrm{t}) = \Theta_0 \cdot \sigma\big(\bar{x}(\mathrm{t})\big) - \Gamma \cdot \sigma\big(\bar{x}(\mathrm{t})\big) \cdot \int_0^t e^T(\tau) \cdot PB \cdot \sigma\big(\bar{x}(\tau)\big) \cdot \mathrm{d}\tau$$

If using this expression in the definition of error derivative in equation (6-1):

$$\dot{e} = A \cdot e + B \cdot \left(\Theta_0 \cdot \sigma\big(\overline{x}(\mathrm{t})\big) - \Gamma \cdot \sigma\big(\overline{x}(\mathrm{t})\big) \cdot \int_0^t e^T(\tau) \cdot PB \cdot \sigma\big(\overline{x}(\tau)\big) \cdot \mathrm{d}\tau - \Delta \right)$$

Where it can be seen that the time derivative of the error has a proportional action given by a Hurwitz matrix A and an integral action given by the adaptation law. This integral action is not defined as in classical control but in such a way it changes dynamically depending on the dynamics and responses of the plant, that is, it can be considered as an “intelligent integral action”.

The sense of integral action is to provide a control command proportional to the integrated error over time. But, when control saturations occur and the system is not physically capable of reach the commanded signals from the controller, the integral action would continue accumulating an error and commanding a bigger and bigger command, even when it is unreachable. The problem is that when saturations of the control surfaces are no more needed and the integral action keep commanding an excessive control command, ending in an uncontrollable response.

The way of solving this problem is to implement an anti-windup system that prevents the integral controller from incorporate non-feasible tracking errors.

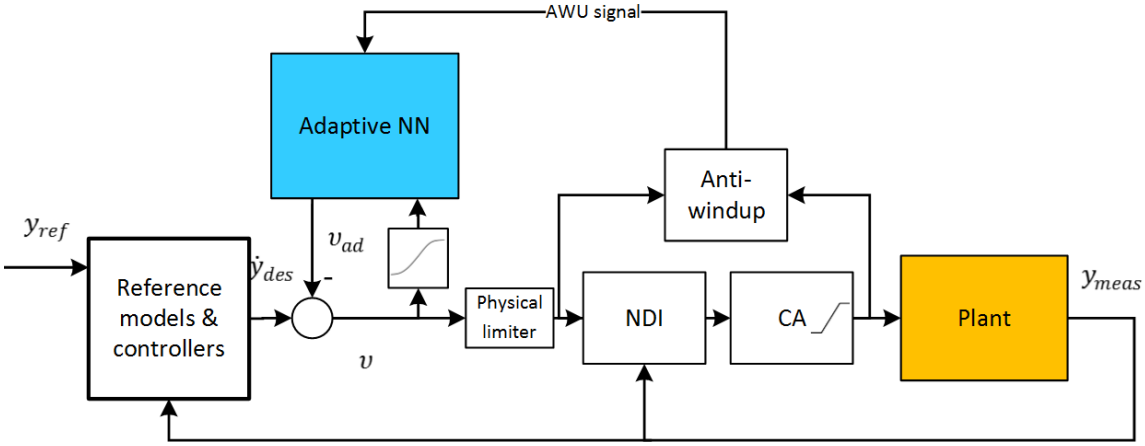


Figure 6-4 AWU implementation

Figure 6-4 shows how this anti-windup scheme is implemented along with the presented adaptive controller. The idea is to observe the commanded signals coming from the controller and compare them with the angular accelerations that would be created by the deflections that NDI and CA have computed, due to the fact NDI and CA include the physical limits of the control surfaces. Note that anti-windup system, through that comparison, computes excess angular accelerations that could not be reached.

The aim now is to inject these excess angular accelerations into the NN stopping or slowing down the learning process, so that the NN does not command unfeasible commands. Hence, there is two ways to implement it:

- Modifying the learning process error ξ to slow down the learning: reduce the integrated error as classical control establishes
- Stopping the learning

Table 6-8 shows how the two approaches are implemented with the NN variables, where the excess angular acceleration computed by the anti-windup system is called AWU.

For the first option, affecting tracking error, the selection of an appropriate K_{AWU} is critical while the second option, stopping the learning, maybe be excessive in some cases. For the work presented, first option was chosen and, as a guideline, K_{AWU} can be design to be K_{AWU}~3.

Affecting tracking error	**Stopping learning**
for i = 1: 3 $\xi'(\mathrm{i}) = \xi(\mathrm{i}) \cdot \max(0{,}1 - K_{AWU} \cdot \lvert AWU(i) \rvert)$ end $K_{AWU} > 0$	$\Gamma_i' = \begin{cases} \Gamma_\mathrm{i} & if \ \lVert AWU \rVert = 0 \\ 0 & otherwise \end{cases}$

Table 6-8 AWU implementation

These implementations ensure that the learning process is unaltered while saturations do not occur, but when they do, the NN behaviours changes.

Figure 6-5 shows a particular case in which it is asked for a negative roll rate of $200\ deg/s$ at $4000m$ of altitude and $100m/s$ of V_{TAS}. This command is excesive for the given flight condition and aircraft capabilities, so anti-windup signal becomes non-zero when ailerons get saturated: between second ~8 and ~16. Note that for this case, sensor noise was not added in order to clarify the results and make them visible.

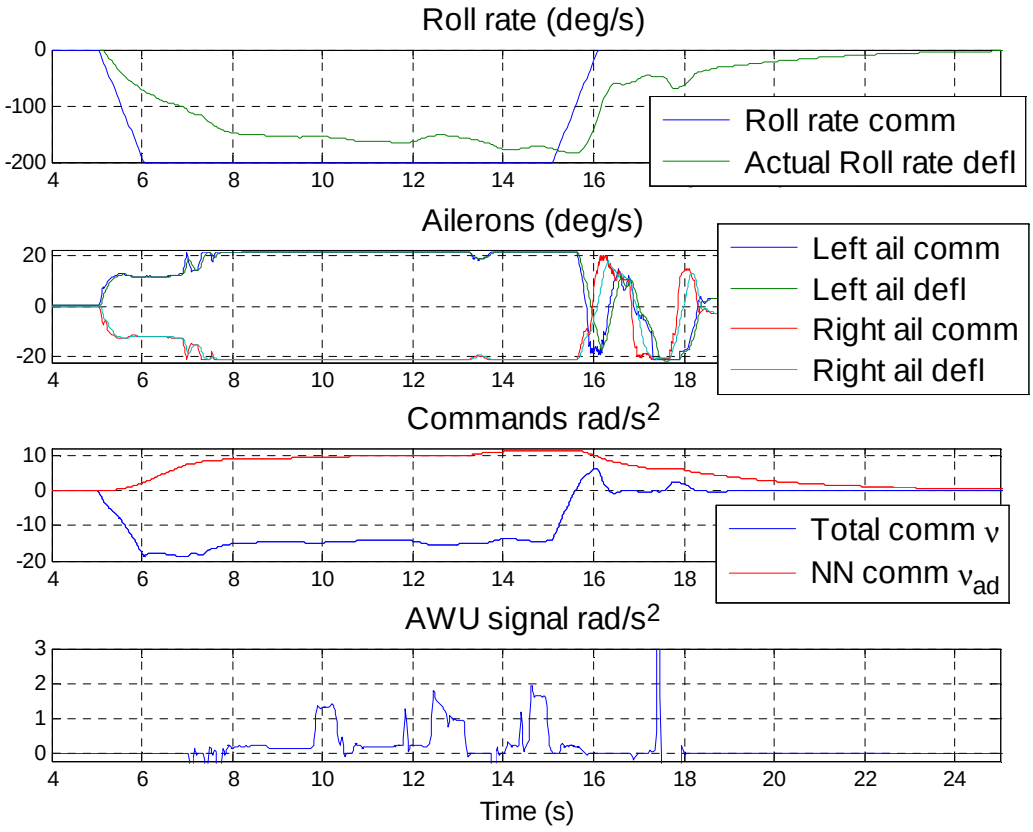


Figure 6-5 AWU Roll demo

The most important fact that can be seen in Figure 6-5 is that the NN output υ_{ad} remain almost constant while saturations occur, making evident that the implemented algorithm performs as required. If no anti-windup system were implemented, the NN signal would not stop growing under the given steady error.

Similarly, in Figure 6-5, the commanded angular acceleration υ also remains almost constant, which is the objective of the presented solution. This fact means that the control system is commanding only the feasible commands for the given saturations and aircraft speed.

So, finally, it has been considered that the anti-windup system implemented and the way to inject the anti-windup signal into the NN, provide the required effect and performance.

Analysing the problem better, it was noticed that this algorithm must not work alone, unless total commands given to NDI υ are limited to ensure they are realistic and reachable. For this purpose, a physical limiter, whose representation is shown in Figure 6-4, is placed at the output of the controller and NN. The aim is to limit the commanded angular accelerations so that they can be reached.

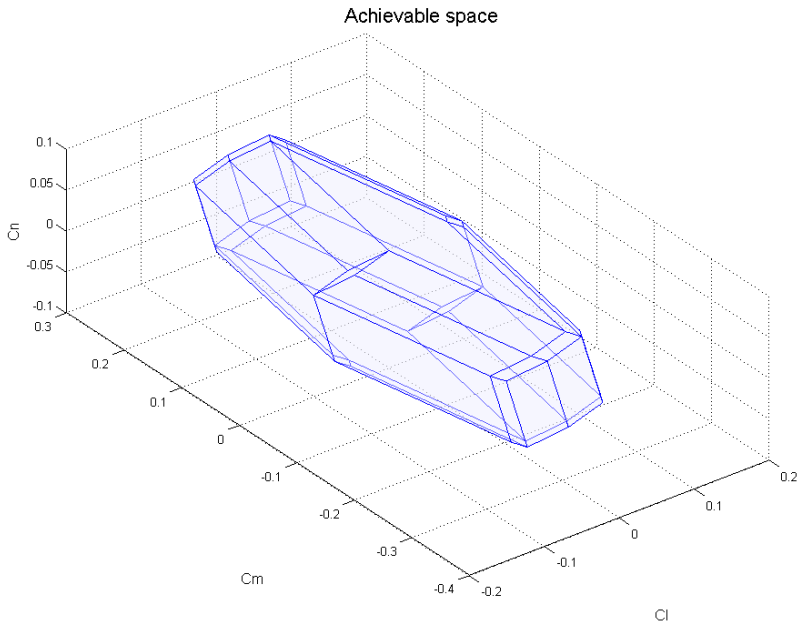


Figure 6-6 Achievable space

These limits are created using the presented model representation and evaluating it at the current dynamics and at all the extremes deflections, in order to compute the achievable space in terms of moment coefficients, as shown in Figure 6-6.

So that, the limiter does not allow any angular acceleration that exceed the limits of the airplane, as represented in Figure 6-6, to be commanded and, as a consequence, unreachability of the commands and coupling with anti-windup system are avoided.

7 System Identification and Fault Detection and Isolation

The adaptive control system already presented belongs to the basic concept of adaptive controller. That is, it can manage certain faults and deviations, but not large problems.

In order to perform fault-tolerant control something else is needed. The most common techniques in fault-tolerant control are System Identification (SI) and Fault Detection and Isolation (FDI).

FDI are usually implemented as a separate module in fault-tolerant control systems: It is a system that monitors the health of the aircraft. The responsibility of this module is to identify whether or not a fault occurs, estimate its severity or grade and pass this information to the reconfigurable control system, that, in this case, are the NDI and CA modules. With this information, reconfigurable control will reconfigure the control effort to keep controlling the aircraft. Note that if control objectives are not possible to reach due to the high degradation of the health, it is considered that failure occurs.

System Identification is usually used as part of adaptive control systems, since some of the methods explained in section 2 of adaptive control, are based in identification of the plant to control or need it as secondary information. In the work presented here SI is used to identify only major changes or damages in the airframe.

The reason because SI is used in the fault-tolerant concept and not in the Adaptive one, is because when large damages to the airplane occur, big errors in NDI appear and, although NN may make the system stable, controllability of the aircraft is reduced. That is, regardless stability of the damaged aircraft, control authority is reduced under big damages and SI is needed to identify these large deviations and recover as much controllability as the aircraft still has.

In this section SI general theory is presented, along with the method used in this work, specifying which parameters are identified.

FDI algorithms are usually quite complex and involve a significant of design effort to get good results, as well as, big computational load. For these reasons, in the presented work FDI module was assumed as already implemented and developed, generating the signals that a normal FDI module in literature (see [45], [44], [31]) would produce, including transients, delays, noises, etc. A presentation of the signals generated and their characteristics are done in lastly in this section.

7.1 System identification

System identification consist basically in identify parameters of a given model based on measured information, so that both model and measurements match.

As said, SI is used to configure the approach called fault-tolerant since this approach contemplates major damages, deviations and faults in the airframe.

In fact, the way SI is implemented in the work presented and the way to use the information it generates, can be classified as direct adaptive control.

So: Why to implement another adaptive method?

The intention is to increase manoeuvrability under big damages of the airplane.

The exact reason is found when large asymmetries appear in the airframe: for instance, loss of a part of right/left wing, large transversal movements of the CG, loss of fin surfaces, etc. When these asymmetries appear, the already implemented NN may compensate the deviation, but a cross-effect appears: now longitudinal and latero-directional channels are coupled and, if no reconfiguration or identification of the coupling is performed, controllability is reduced dramatically and the pilot would be prevent from controlling the aircraft successfully.

So, the solution to cope with this cross-effect is to identify on real time the asymmetries of the aircraft based on the behaviour and, using this information, adapt NDI and CA modules consequently so that they consider the asymmetries of the aircraft.

7.1.1 General approach

System Identification has a well-developed theory along many years. There exist many techniques.

The most famous technique is based in identification of ARX or ARMAX models from time series responses using least square methods for example.

There exist other more recent techniques like estimation in frequency domain, online model estimation, non-parametric frequency domain methods, etc. A good overview of these techniques is found in [41].

Generally, SI can be though as the following problem:

- There exist a generic model $y(t) = \mathcal{F}\left(\begin{Bmatrix} x(t) \\ x(t - k \cdot \Delta t) \end{Bmatrix}, y(t - k \cdot \Delta t), t; \theta\right)$ that is wanted to be identified in function of a set of parameters θ
- There exist a measurement of $y(t)$ at time t such that $\hat{y}(t)$

So, having an historic measurement of a time span T or N samples in a discrete space, using any technique, the generic model is "fitted" into the measurements. A least-square minimization technique can be represented either in continuous time:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \int_{t=0}^{T} V_T(t) \cdot dt$$

Or in discrete domain:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{t=0}^{N} V_N(t)$$

With $V_T\ (t)\ \ OR\ \ V_{N(t)} = \left\| \hat{y}(t) - \mathcal{F}\left(\begin{Bmatrix} x(t) \\ x(t - k \cdot \Delta t) \end{Bmatrix}, y(t - k \cdot \Delta t), t; \theta\right) \right\|^2$

Being $\mathcal{F}\left(\begin{Bmatrix} x(t) \\ x(t - k \cdot \Delta t) \end{Bmatrix}, y(t - k \cdot \Delta t), t; \theta^*\right)$ the identified model.

Techniques to perform this minimization can be for example: Gradient descent, Conjugate gradient, etc.

But, if on-line SI has to be performed, like it is the case, online algorithms have to be used. Considering the time step k and θ_k^* the estimated parameter of the model at that time, the recurrent process can be expressed as follows (see [41]):

$$\theta_{k+1}^* = \theta_k^* - K_k \cdot \vartheta\left(\begin{Bmatrix} x_k \\ x_{k-i} \end{Bmatrix}, y_{k-i}, t_k; \theta\right)\Big|_{\theta=\theta_k^*} \qquad \textbf{(7-1)}$$

Where $\vartheta\left(\begin{Bmatrix} x_k \\ x_{k-i} \end{Bmatrix}, y_{k-i}, t_k; \theta\right) = \hat{y}_k - \mathcal{F}\left(\begin{Bmatrix} x_k \\ x_{k-i} \end{Bmatrix}, y_{k-i}, t_k; \theta\right)$ is the error produced at the time step k between the measurement and the model output and K_k is a weight or weight matrix that provides the learning, minimization or search direction, depending on how it is called.

Common techniques to provide this search direction can be extrapolated form the mentioned techniques for minimization like Gradient Descendent, where:

$$K_k = \mu \cdot R_k^{-1} \cdot \left. \frac{\partial\, \vartheta\left(\begin{Bmatrix} x_k \\ x_{k-i} \end{Bmatrix}, y_{k-i}, t_k; \theta\right)}{\partial \theta} \right|_{\theta=\theta_k^*} , with\ \mu > 0\ and\ \ R_k = I$$

In the work presented here, a similar procedure is implemented. That is, a recurrent algorithm is used, but using an optimal procedure of the problem in equation (7-1): Kalman Filtering.

7.1.2 Implemented methodology

The approach used in this work is a model-based technique. That is, there exists previous knowledge of the model to be identified when identification is performed. Hence, SI is only performed to identify those parameters of the model that are unknown, in this case, asymmetries.

The parameters to be estimated are: Left wing surface, right wing surface and fin surface as stated in section 4.5:

$$S_{L_{Current}}, S_{R_{current}}, S_{fin_{current}}$$

Since cross-effect are intended to be identified and control of rotational dynamics is performed with NDI and CA, a rotational dynamics model as presented in equation (3-6) is used. Since, this algorithm is going to be used in the fault-tolerant concept, reconfiguration of the mentioned model using FDI estimated parameters is performed in the used model in SI.

Since the model in equation (3-6) gives angular accelerations in body axes, the measurements must be angular accelerations in body axes. But in INSs these measurements are not available so they have to be estimated using a numbering approximation and its correspondent filter to deal with noise. This noise is mainly due to IMU noise and discrete derivative noise, which generally together are very big.

In fact, the proposed algorithm for SI does not needed any pre-filter to reduce the noise level since it is a low pass filter indeed, since it can deal with noises and uncertainties.

There may exist several ways to use only measured angular rates along with the model of rotational dynamics in any SI algorithm, but, due to the fact that both model and measurements provide variables of different order, all that can be done is to approximate

the derivative (or integral) on an appropriate way. Here a discrete derivative with a LPF is implemented.

As said, a recurrent algorithm was implemented for the purpose of parameter identification, using Kalman Filtering theory. Many authors consider this recurrent algorithm to be optimal since they use search directions that minimize uncertainty.

Kalman Filters, when compared with an online least-square method using a GD direction for instance:

Advantages	**Disadvantages**
Provides an estimation of the goodness of the predictions Provides an search direction that minimizes uncertainty of the estimations	Search direction and error matrix computing requires more computational power

Table 7-1 KF vs. GD

Precisely, Uncested Kalman Filter was implemented in the work presented to carry out parameter estimation. This algorithm was found to be perfect for the task presented since, it is a recurrent on-line algorithm that does not need any gradient to be provided while still providing a measure of the goodness of the estimations.

Let us define now the rotational dynamics model of equation (3-6) as:

$$\dot{\omega}_{model} = y = \mathcal{F}(x; S_i)$$

Where y are angular accelerations, x denotes inputs variables that represent the dynamics and S_i are the parameters of the model, that is, the three surfaces to identify.

But, the measurements of the IMU are ω_{meas} which is noisy. The angular accelerations are computed:

$$\dot{\omega}_{estimated} = \hat{y} = \frac{s}{s \cdot c + 1} \cdot \omega_{meas} \qquad \textbf{(7-2)}$$

With $c \gg 1$.

As commented before, since the difference in order between the measured and model variables, an approximation of the derivative (or integral if wanted) has to be done. Any

approximation to the problem may be based on interpret equation (7-2) in a different way, but essentially the idea is collected in equation (7-2).

So, defining now the algorithm, let us define $z = \hat{y}$ as the measurement, the function $\mathcal{F}(x; S_i)$ as the measurement equation and S_i the variables of the system. The system is described:

$$\begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_{k+1} = f\left(\begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_k, t_k\right) + w_k = A \cdot \begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_k + w_k \quad \textbf{(7-3)}$$

$$z_k = h\left(\begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_k, t_k\right) + v_k = \mathcal{F}\left(x_k; \begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_k\right) + v_k \quad \textbf{(7-4)}$$

Being equation (7-3) the system equation that describes the discrete process of the model in terms of the variables S_i and equation (7-4) the measurement equation of the system. Note that for this problem, the model is used as measurement equation.

Of course, since the problem is about parameters estimation and there not exist any information about how the variables to be estimated, the surfaces S_i, may change with time, $A = I$.

Let us define as well the process noise $w_k \sim N(0, Q_k)$ and the measurement noise $v_k \sim N(0, R_k)$, both zero-mean Gaussian distributions with the given covariance Matrices. It is also assumed that w_k and v_k are not coupled.

Using this notation, and defining the matrix P as the estimation error covariance, the implemented UKF is presented in Table 7-2 , where an appropriate initialization and parameters set should be done. For instance matrices $Q_k = Q$,$R_k = R$ have to be correctly set in order to obtain the required performance. In this particular case, since the measurements are very noisy, Q is very small and R very big:

$$|Q| \ll |R|$$

Noting $|Q|$ as the determinant of Q.

Initial conditions have to be also set properly. For instance, initial error covariance P_0 should be as big as the uncertainty of the given initial condition is. In this case, it can be small. The initial conditions of the variables to estimated are set $\begin{pmatrix} \vdots \\ S_i \\ \vdots \end{pmatrix}_0 = \begin{pmatrix} \frac{27.89}{2} \\ \frac{27.89}{2} \\ 6.578 \end{pmatrix}$

Prediction:	**Update**
$\begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{pred} = I \cdot \begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_k$ $P_{pred} = P_k + Q_k$	$\alpha = 1, \beta = 0, \kappa = 3 - L, \lambda = \alpha^2 \cdot (L + \kappa) - L\,, L = 3$ $W_s^0 = \frac{\lambda}{L+\lambda}, W_c^0 = \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta)$ $W_s^i = W_c^i = \frac{1}{2\cdot(L+\lambda)} \quad , \; i = 1..2 \cdot L$ $\mathcal{X}^0 = \begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{pred}$ $\mathcal{X}^i = \begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{pred} + \sqrt{(L+\lambda)\cdot P_{pred}} \; , \; i = 1..2 \cdot L$ $\gamma_i = \mathcal{F}(x_k; \mathcal{X}^i) \; , \; i = 0..2 \cdot L$ $\hat{Z}_k = \sum_i^{2\cdot L} W_s^i \cdot \gamma_i$ $P_{z,z} = \sum_i^{2\cdot L} W_c^i \cdot (\gamma_i - \hat{Z}_k) \cdot (\gamma_i - \hat{Z}_k)^T$ $P_{x,z} = \sum_i^{2\cdot L} W_c^i \cdot \left(\mathcal{X}^i - \begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{pred}\right) \cdot (\gamma_i - \hat{Z}_k)^T + R_k$ $K_k = P_{x,z} \cdot P_{z,z}^{-1}$ $\begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{k+1} = \begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{pred} + K_k \cdot (\dot{\omega}_{estimated} - \hat{Z}_k)$ $P_{k+1} = P_{pred} - K_k \cdot P_{z,z} \cdot K_k^T$

Table 7-2 Applied UKF for SI

In Table 7-2 the general algorithm is presented where the estimated surfaces are given by $\begin{pmatrix}\vdots\\ S_i\\ \vdots\end{pmatrix}_{k+1}$ and the quality of this estimation is given by P_{k+1}, its determinant $|P_{k+1}|$ or the Shannon definition of "Entropy" for the Gaussian distribution of the estimation, with $n = 3$ in this case:

$$\mathcal{S}_{k+1} = \frac{1}{2} \cdot (\ln(|P_{k+1}|) + n + n \cdot \ln(2 \cdot \pi))$$

Performance of this algorithm can be seen in Figure 7-1 that corresponds to a case in which 50% of the right wing surface is loss. The figure shows the three estimations of surfaces, and it can be seen that, firstly, the noise of measurements is greatly reduced by the filter. Secondly, the estimation of right wing surface converges to the right value after ~30seconds.

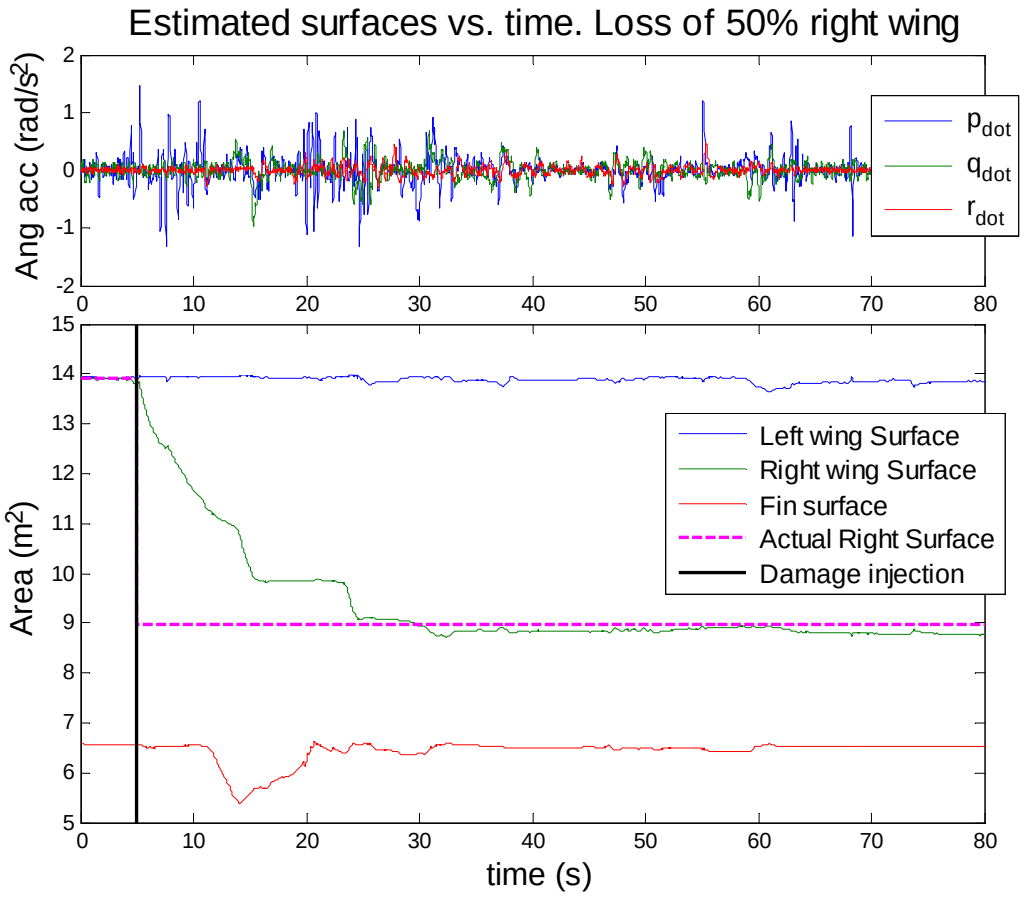


Figure 7-1 SI under 50% loss right wing. Top: meas. Angular accelerations. Bottom: SI estimations

In order to evidence how noisy are the estimation of angular accelerations, Figure 7-1 shows in the top part the angular accelerations estimated with equation (7-2), where the signal is noisy due to the discrete differentiation and the noisy IMU measurements.

Note that Figure 7-1 represents the mentioned damage case under pilot inputs, that is, the pilot recovered from the damage and manoeuvred the aircraft to recover it. That is why there exists a non-monotone convergence, since when pilot excites the system and measurements are not near zero, convergence speed up. In fact, for the presented case, pilot manoeuvred up to the second 60, and did nothing from then on.

Regarding the estimation quality, Figure 7-2 shows the determinant of the error covariance matrix and the Entropy in the Shannon sense of the predictions in the same presented case.

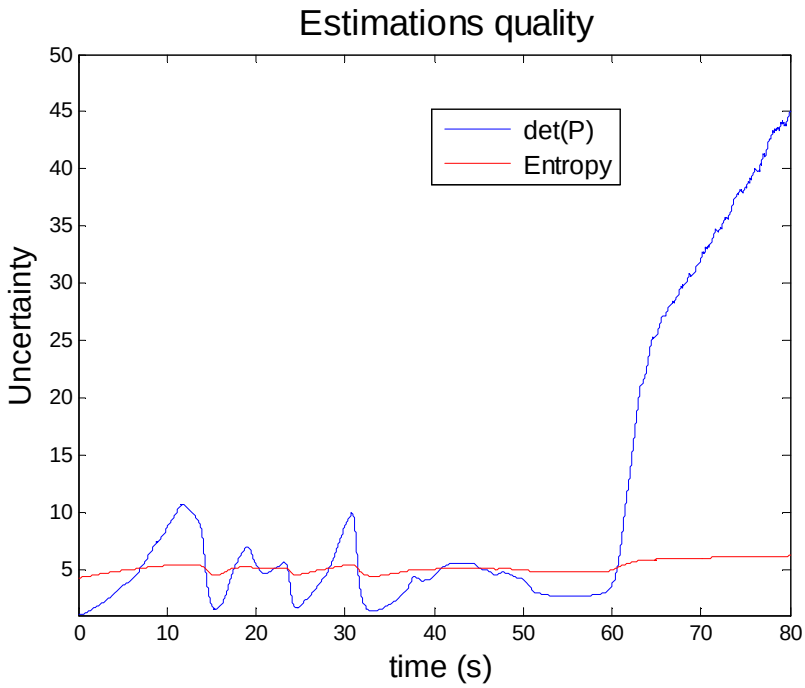


Figure 7-2 SI estimations quality

It can be seen in Figure 7-2 that information quality degrades when failure occurs but, while pilot excites the system, estimations qualities improve. From the second 60 on, the pilot did not perform any manoeuvre and, as seen, quality deteriorates.

So, it can be seen by comparison of Figure 7-2 and Figure 7-1 that if there exist a persistent excitation, SI system provides good estimations with a reduced uncertainty but, when there is not excitement, estimations remain unchanged but their quality decreases.

This effect is not a problem if using estimations directly since, with the prediction model implemented in the UKF, estimations will not change if no excitement exists but, uncertainty increases if there is not a persistent excitement, what may be a problem if the designer wants to ensure certain level of accuracy.

This deterioration of the estimations is due to the "co-linearity" ("ganging" in [23]) of the parameter being estimated, that is, different combinations of parameters may give the same output in the measurement equation. So, this problem is inherent to SI systems.

So, a SI exciter should be implemented in order to keep a constant excitation of the system to help to SI system to speed up the identification and improve the quality of the estimations.

For the presented work and the given parameters to be estimated, normal pilot inputs are thought to be sufficient to identify the parameters to be estimated. Quality of these estimations is taken as a secondary objective and it is not intended to be always improved in the presented work, since correct convergence of the estimations was observed. So, SI exciter was found to be not needed in the presented work due to the correct performance on parameters identification.

Inclusion of this mentioned SI exciter is encouraged for future works. It should include semi-random commands to the control system that must be visible by the SI system. In particular, roll-yaw commands were found to be effective when improving the quality of the estimations.

7.2 FDI

In order to perform fault-tolerant control a FDI system is needed. This system monitors the health of the aircraft, control surfaces typically, to keep informed the rest of the control system about it.

In the work presented, FDI algorithms are not implemented due to the fact they are usually complex systems developed separately of control systems. This thesis is focus in Adaptive and fault-tolerant control and not on FDI, which are usually considered as separated matters.

But, in fact, to perform fault-tolerant control information like the provided by an FDI module is needed. For this reason, a FDI-like module was developed, using actual reconfiguration signals from the faults injection panel (section 3.3.4) and modifying them in order to get outputs like a normal FDI would produce.

For this purpose, FDI results and performances in [81], [31], [82], [45] and [44] were used to design the set of noises, delays and 1^{st} order responses that a real FDI module would have when estimating Blockages, floating and loss of area or effectiveness of control surfaces. Note that total loss of effectiveness is considered as floating.

In this section loss of effectiveness and blockages simulated signals are presented along with the actual faults injections.

7.2.1 Loss of effectiveness

In order to simulate how an actual FDI module would estimate the loss of area in a control surface important delays, noises and 1st order responses were added.

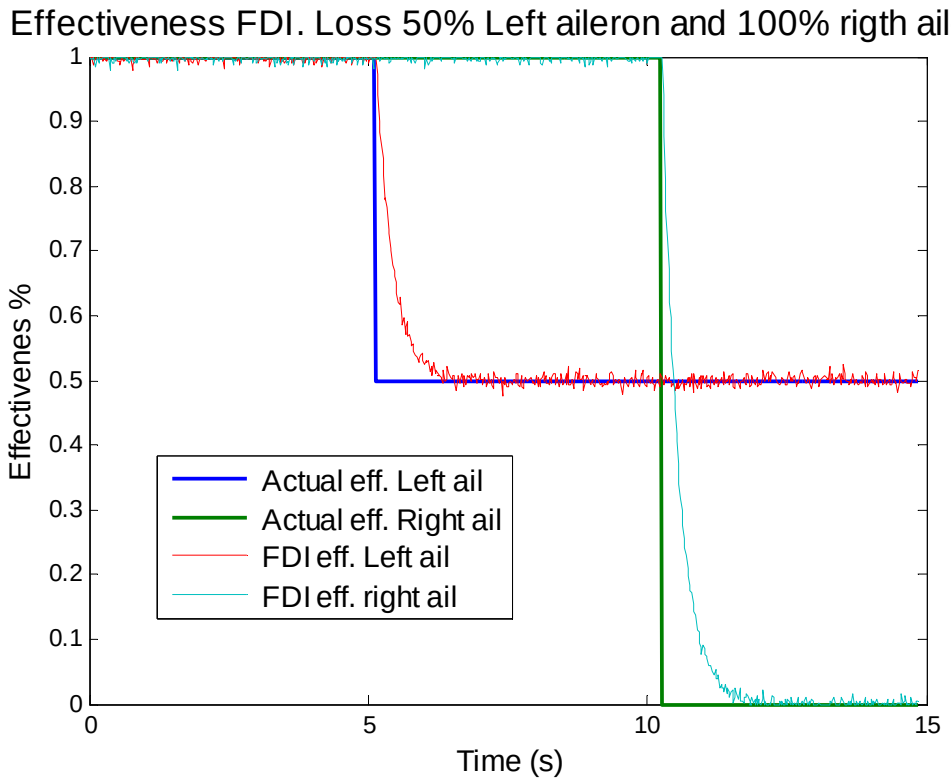


Figure 7-3 FDI effectiveness reduction detection

Figure 7-3 presents the cases in which left aileron losses 50% of area or effectiveness and right aileron losses 100% (floating). As seen, the simulated FDI module adds a certain amount of noise in order to represent uncertainty of the estimations and adds a 1st order response such that it takes around 1.5seconds to converge to the right value.

The design of these responses were based in [81] (page 788), where the authors were able to design several methods to detect loss of effectiveness, all of them below 1.5 seconds of time response, having a very similar response to the presented in Figure 7-3.

7.2.2 Blockages

In order to simulate how an FDI module would respond to an actuator blockage, actual signals form the fault injection panel are modified adding noises, delays and a 1st order responses as shown in Figure 7-4.

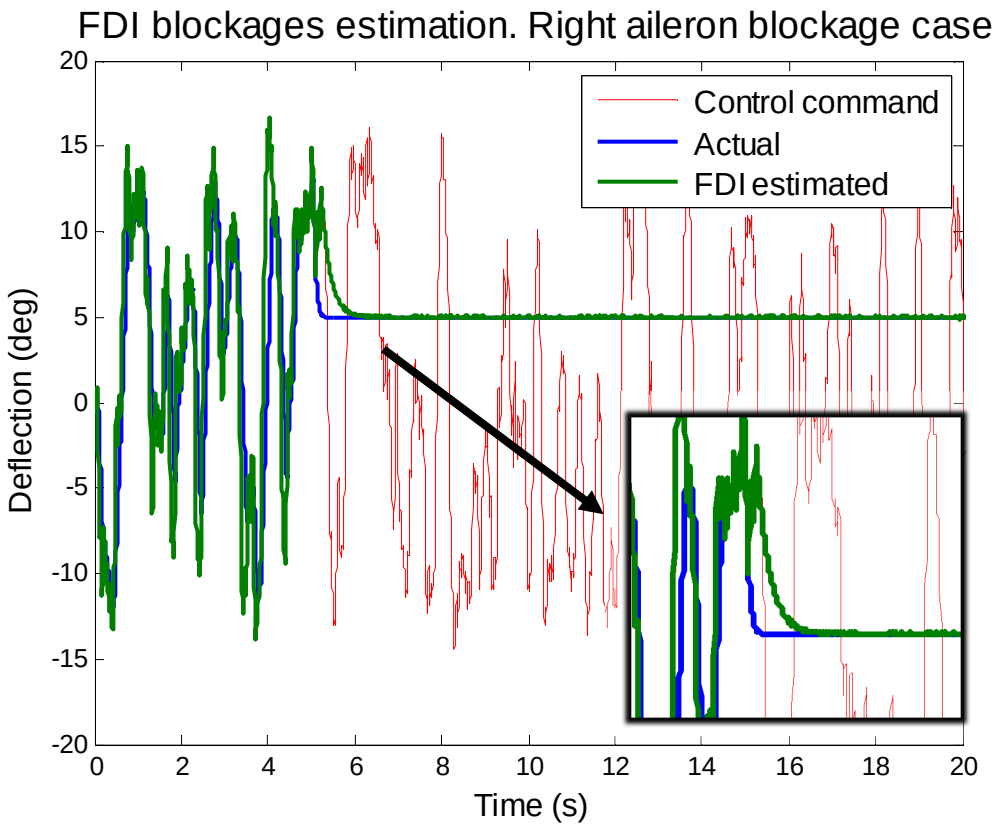


Figure 7-4 FDI Blockages behaviour

Figure 7-4 shows the case in which right aileron is blocked at ~5seconds at 5deg, while the control strategy is not fault-tolerant. In contrast, the control system was not informed of the estimated blockage (Adaptive concept) in order to make more evident the clear difference between the control command, the actual deflection and the estimation of FDI module. As well, IMU noise was incremented intentionally so that control system would be more variable and differences between the mentioned three signals become more evident.

Figure 7-4 shows how around ~5seconds actual deflection goes to 5deg and control signal does not, since control system was not informed of the fault (adaptive concept). FDI signal converges to the actual deflection after ~1second with a certain level of noise.

Authors in [31], [82], [45] and [44] present results of many different strategies to estimate faults in control surfaces. Results in these publications were taken into account when designing the proposed responses shown in Figure 7-4. In fact, [45] and [82] results show that their FDI estimations converges almost immediately to the actual deflection.

In the work here presented, a delay bigger than in [45] or [82] was implemented in order to be conservative and to simulate any decision making process that would work with the FDI information.

8 Validation and Analysis

Algorithms explained in sections 4 to 7 configure the Basic (Adaptive) and Advanced (Fault-Tolerant) Concepts, as explained in section 1. In particular, the Basic concept implements algorithms in sections 4 to 6, while the Advanced solution also includes a reconfigurable scheme including as algorithms from 4 to 7. Note that Advanced concept is an adaptive and reconfigurable controller at the same time.

In this section performance of these proposed architectures are tested.

Firstly, Basic concept design is validate by checking its capabilities and performance against specification. For this specification compliance, MIL F-8785C specification is used since it is the one that applies to the F-16 aircraft. With this specification compliance analysis, performances of the control system in different flight conditions are assessed, giving a good picture of the capabilities and goodness of the architecture and particular design.

Since the benefits of an adaptive controller are not only evidenced in off-design conditions performances and it provides stability under failures, performances of this Basic concept under different deviations, faults and damages of the aircraft are checked. This analysis reveals an upper limit of performance of the Basic concept under damage and, hence, the boundary where Fault-Tolerant controller is needed.

The deviations, faults and damages used for this purpose are intended to be realistic cases of configuration changes and real battle damage, for instance.

Lately, Advanced concept performance is checked under deviations, faults and damages. Since, this concept is based in the Basic concept, the handling qualities of this controller are the same as the Basic controller. Only faults and damages receiving cases are tested in this concept, since deviations (e.g. configuration changes) performance is the same as in Basic concept. Faults and damages used for testing this Concept include normal failures that can occur in a normal flight and severe faults/damages that can occur in a combat situation. The intention of inflict so severe damages is to discover the limits of the capabilities of the controller and/or the physical capabilities of the aircraft.

From this section on, despite the fact longitudinally the aircraft can be controlled either in AoA or load factor, only control in AoA is used. Note that work implemented include and switching system that allows control in load factor at high speeds and control in AoA at low speeds. This system is not tested in these sections.

Also, despite the gain scheduling designed in section 5.3.4, only the fixed design in section 5.3.3 is used. It can be though that if gain scheduling is used the performance in any point of the flight envelope is the best performance the aircraft can get with the given dynamic pressure.

In order to be realistic, it is assumed, as said in section 1, that the only information available for the control system is given by typical sensors on an aircraft and that this information contain noises, bias, delays, etc. These effects were added to the real signals in order to simulate typical sensors performance on an aircraft. Specifications of typical sensors (INS, IMU, GPS, Pitot tube, static probes, AoA and SS blades) from normal aeronautical manufacturers were used to build the simulated sensors.

As said at Introduction, a speed controller was implemented within the Concepts explained but in normal applications there not exist such control in speed. In contrast pilots control thrust level or power level. So, in this section and in section 9, speed controller is not used, being thrust level a variable to be manually controlled.

8.1 Basic concept

Here performance of the Basic concept is presented. Firstly MIL F-8785C compliance of the controller is presented in order to validate the design of the controller. As said a fixed design in section 5.3.3 is used controlling longitudinally in AoA.

Secondly performance of this concept under deviations, faults and damages is presented, where the intention is to see how the neuro-adaptive controller works performing adaptation and which its limits are.

8.1.1 Flight envelope performance: Specifications compliance

Here design of the Basic concept working with a given configuration of the aircraft in Table 10-1 is validate by assessing specification compliance.

This specification has several fundamental parts:

- Scope
- Requirements
    - General requirements
    - Manoeuvring and handling qualities sections

In the scope the airplane, levels of qualities and flight phases that specifications apply for are described.

In the first part of general requirements flight conditions, loading cases, configurations, mass properties and missions and manoeuvres are defined in order to deliver a precise specification for exact cases. In this section the aircraft and the presented flight cases are presented.

Regarding handling and manoeuvring qualities, most of the specifications are set for a given range of flight conditions given by the exact section in the specification or by the Category being examined.

In those cases there exists a range of flight conditions where exact specification is applied, a partition of the flight envelope is used. This partition is intended to collect the performance of the control system in different flight conditions in order to get a big picture of its capabilities in all flight envelope.

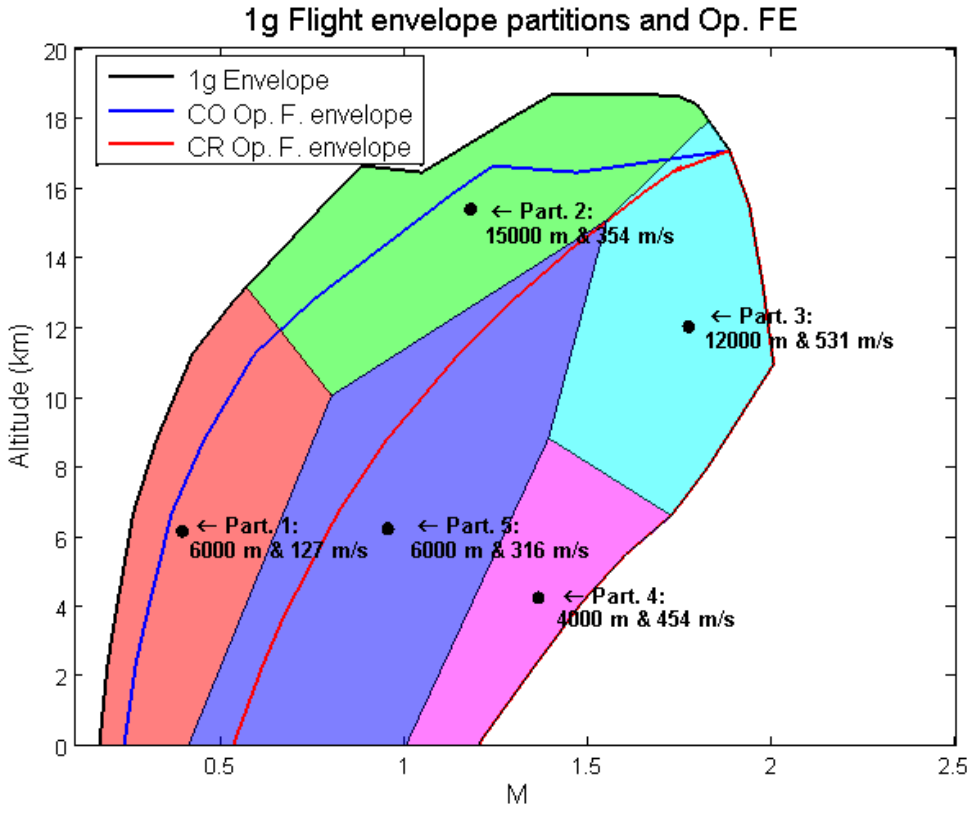


Figure 8-1 1g FE partitions

Five partitions of the flight envelope are presented in Figure 8-1 accordingly with the nature of each region of the Flight envelope. Partitions 1 and 2 are near-stall regions in

which dynamic pressure is particularly low. Partitions 3 and 4 are near maximum speed regions in which dynamic pressure is high but flight is limited because of power. Partition 5, central partition, does not present any particular issue. So, performance there should be optimal.

Note that the specification states several sections regarding different conditions and particular cases that may not apply for the case studied here. For instance, sections referring to stick forces, stick forces gradients, cross wind, wind disturbances and so on, are not taken into account.

8.1.1.1 Specification section 1: Scope

The flying qualities that are going to be checked belongs to the aircraft along with the Basic Flight Control system proposed without considering any other automatic control system such as sped controller, altitude holder, etc. The requirements are checked in terms of flight controls that produce AoA variations, roll rate variations and SS variations.

The first task this section states is to classify the aircraft by type, where the used model of F-16 belongs to Class IV for being a Fighter/Interceptor aircraft.

Regarding Flight Phase categories, an aircraft should comply with all flight phases it is intended to accomplish. Here flight phases of Cruise (CR) and Air-to-air Combat (CO) are considered for being the most representatives of the aircraft.

Finally, the Level of flying qualities chosen is Level 1, since the intention is to fully comply with requirements.

So, finally the aircraft would be evaluated for the cases:

Class	Category	Level
IV	A: Air to air combat B: Cruise	1

Table 8-1 Class, Categories and Level used in compliance

8.1.1.2 Specification section 3.1: General requirements

Operational missions taken into account for this compliance are stated in Table 8-1. Similarly, loading case, moments of inertia and mass configuration is chosen as a representative case presented in Table 10-1.

Configuration taken into account is a Clean configuration with no flaps neither undercarriage deployed. Note that only CR and CO Flight Phases are considered.

The estate of the aircraft considered is totally undamaged and functional.

Operational envelopes of chosen flight phases are determined by Table 8-2, where 1g flight envelope presented in Figure 8-1 is used to determine the physical limits. This 1g flight envelope collects only flight conditions that allows stable level 1g flight

Service Flight envelopes are considered as being reduced flight envelopes that depends on the flight phase.

Category	**Airspeed**		**Altitude**		**Load factor**	
	$V_{0_{MIN}}$	$V_{0_{MAX}}$	$h_{0_{MIN}}$	$h_{0_{MAX}}$	$n_{0_{MIN}}$	$n_{0_{MAX}}$
A: Air to air combat (CO)	$1.4 \cdot V_S$	V_{MAT}	MSL	Combat Ceiling	-1	n_L
B: Cruise (CR)	V_{range}	V_{NRT}	MSL	Cruise Ceiling	0.5	2

Table 8-2 Operation flight envelopes

Operational Flight envelopes are presented in Figure 8-1 considering that $V_{NRT} = V_{MAT} = V_{max1g}$ and that $V_{range} = \sqrt[4]{3} \cdot \frac{1}{C_{L_{opt}}^{1/2}} \cdot \sqrt{\frac{2 \cdot m \cdot g}{\rho \cdot S}}$.

Note that designing point of the control system lies out of the operational flight envelopes (see section 5.3.3).

Note:

- V_{range} speed of maximum range
- V_{NRT} max speed normal rated trhust
- V_{MAT}max speed maximum augmented thrust
- V_{MRT} Military rated thrust

Regarding load factors, Service load factors, permissible flight envelopes and structural maximum load factor are unknown for the aircraft presented.

Nevertheless, load factor (positive) flight envelope limits were found to be as the presented in Figure 8-2. So, these limits are assumed as the boundaries that limit the

rest of envelopes, if not define: For instance, operational load factor envelopes are bounded by Figure 8-2 and positive Service load factor can be defined as in Figure 8-2.

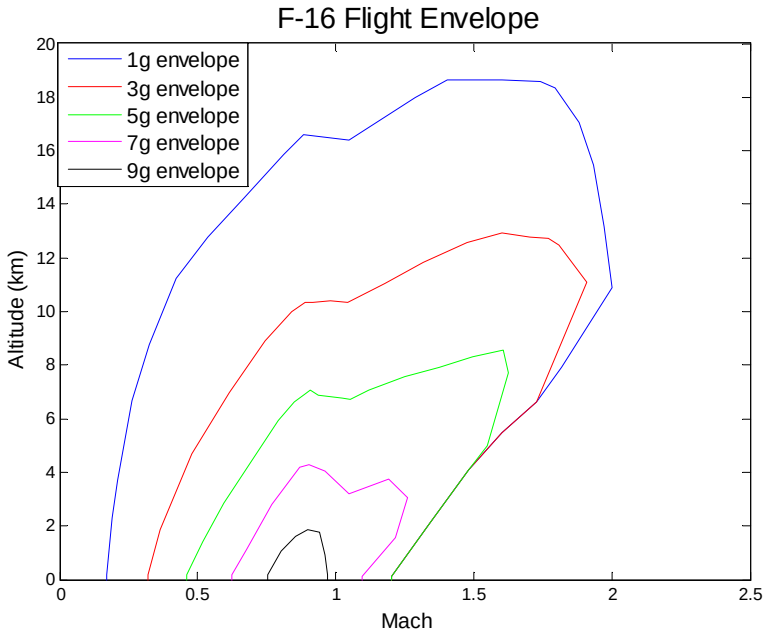


Figure 8-2 Load factor flight envelopes

Regarding service speeds, they are defined:

- V_{MAX} maximum permissible speed: right limit of 1g FE
- $V_{MIN} = \max(1.1 \cdot V_S, V_S + 5.14)$
- V_s is defined as the left limits of FE

8.1.1.3 Specification section 3.2: Longitudinal flying qualities

Regarding longitudinal flying qualities, the specification states three main points:

- 3.2.1. Longitudinal stability with respect to speed
- 3.2.2. Longitudinal manoeuvring characteristics
- 3.2.3. Longitudinal control

Section 3.2.1 in specification is not assessed since it is related with aircraft design indeed and not with control system capabilities. In fact, it was found that regardless the control parameters used the characteristics seen in this section remained unchanged, that is, the control system makes no effect in this characteristics.

Note that since no speed controller is being used, longitudinal stability with respect to speed characteristics can be assessed. If the design speed controller is used, no

Phugoid can be found since it belongs to the natural responses of the longitudinal dynamics.

Section 3.2.1 is found then related with design of the aircraft and speed/thrust control if available. Since these are not objectives of the work presented, assessment of this section is not performed.

8.1.1.3.1 Specification section 3.2.2: Longitudinal manoeuvring characteristics

In this section Longitudinal manoeuvring characteristics are assessed where the most important points belong to Short-period (SPPO in Section 3.2.2.1) mode and its parameters.

SPPO is a natural mode of typical aircraft but, since the proposed FCS provides an augmented control system to the pilot, the natural response of FCS+aircraft is no more the natural response of the aircraft. So, it can be thought that the artificial response on the longitudinal axis (e.g. step response) is the natural response of the system FCS+aircraft in that longitudinal axis. That means that the response of the control system is going to be assessed as if an SPPO mode of the system FCS+aircraft. So, by assessing this "natural" response, FCS performance is being evaluated.

Sections 3.2.2.1.1 and 3.2.2.1.2 are assessed with stick fixed that due to the nature of the FCS would be the same that stick free en neutral position. These sections states characteristics of SPPO in terms of undammed natural frequency, damping, and ratio of n/α as in asdsa in terms of AoA responses that occurs at constant speed when abrupt longitudinal control inputs are exerted. These abrupt commands are artificially added to the elevators signals as a perturbance.

The analysis of the perturbed responses is carried out using time ratio method, producing Table 8-3:

Set-point	ζ	$\omega_n \left(rad/s\right)$	n/α $(g's/rad)$
1	0.95	3.1	10
2	0.96	3.2	24
3	0.5	1.54	61
4	0.56	1.61	100
5	0.8	2.98	40

Table 8-3 SPPO parameters

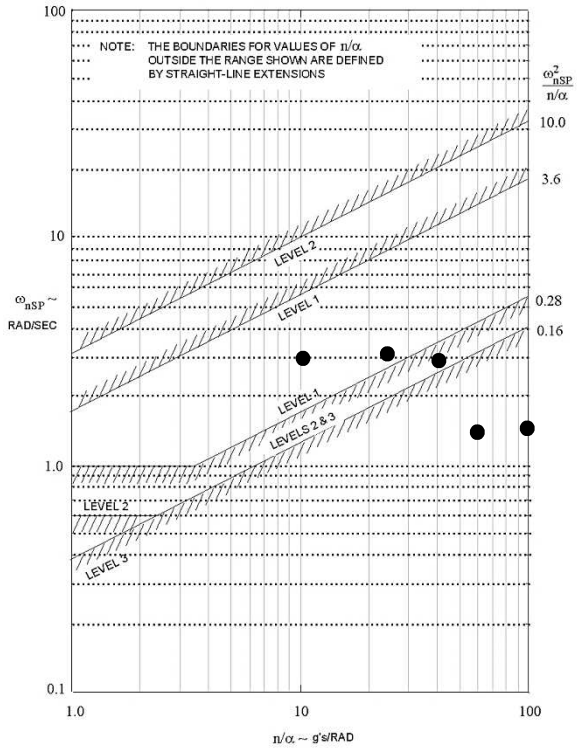


Figure 8-3 SPPO frequency requirement Cat A

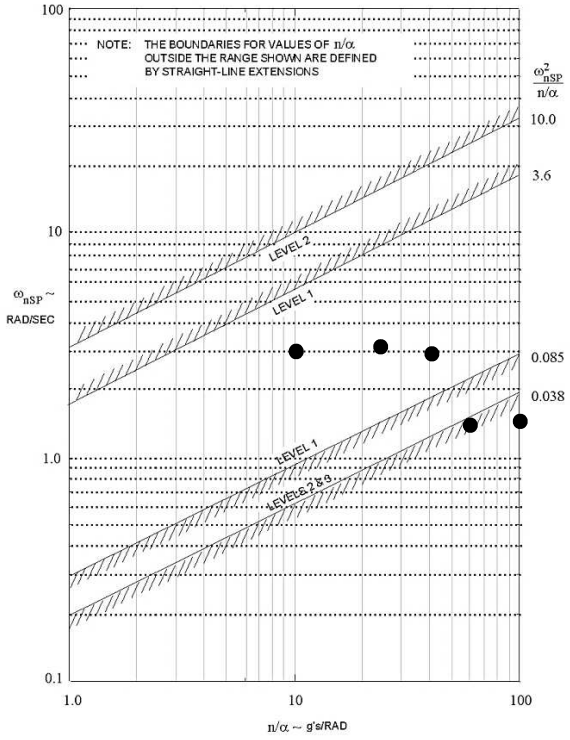


Figure 8-4 SPPO frequency requirement Cat B

Table 8-3 shows responses parameters when stated tests are performed and Figure 8-3 and Figure 8-4 shows same data against the specifications of section 3.2.2.1.1 of specification. As seen not all the partitions representative points (set-points) comply with the requirements for either Cat A and B. In fact, only the closest set-points to the designing point in section 5.3.3 comply with this specification. This is reasonable if analysis of section 5.3.4 is retaken.

Despite the fail of the design to meet requirements in all flight envelope, explicit model following philosophy implemented is an enough powerful tool to provide as good performance as available in the plant. So, the fail of design of section 5.3.3 analysed here is given by two facts:

- With explicit model following architecture, gain scheduling of time constants and gains is found to be needed if requirements must be complied in all flight envelope
- Constrain in equation (5-7) of the controller parameters has been found to strongly limit the performance

In order to demonstrate at least the first of these reasons, Appendix B.1 presents the same analysis of this section but using Gain scheduling of the longitudinal controller parameters as presented in section 5.3.4. These performances, shown in Table 8-4, are noticeably better than in Table 8-3 but compliance for all flight envelope is not yet achieved. This is likely to be because of the second reason.

Set-point	ζ	$\omega_n \left(rad/s\right)$	$n/\alpha \ (g's/rad)$
1	0.77	3.45	10
2	0.9	3.95	24
3	0.98	4.2	61
4	0.55	2.6	100
5	0.82	3.6	40

Table 8-4 SPPO parameters with Gain Scheduling

Regarding the second reason, it has been noticed that constrain in equation (5-7) is limiting the achieved performance. For instance, tuning the correspondent values for the most difficult case found, set-point 4, the design can comply with specifications. This design is shown in Appendix B.2 and in Table 8-5.

Set-point	ζ	$\omega_n \left(rad/s\right)$	$n/\alpha \ (g's/rad)$
4	0.44	6.84	100
AoA		***q with AoA***	
$\tau_{ref} = 0.35$		$\tau_{ref} = 0.06$	
$K = 10$		$K = 1$	

Table 8-5 SPPO parameters for set-point 4 without time constrain

Regarding specification section 3.2.2.1.2, where damping ratios are assessed, the requirements are collected by the following table, where obtained values are in Table 8-3:

Level	**Cat A and C**		**Cat B**	
	Minimum	**Maximum**	**Minimum**	**Maximum**
1	0.35	1.3	0.3	2

Table 8-6 SPPO damping ratio requirements

As seen, in terms of damping ratio, all designs comply widely.

It should be mentioned that when design reference model and controller which at high speeds was found very difficult to achieve.

8.1.1.3.2 Specification section 3.2.3: Longitudinal control

Section 3.2.3 in specification states requirements about longitudinal control capabilities. The most important points, which are treated here, are section 3.2.3.1 and 3.2.3.2 sections.

In section 3.2.3.1 of specification maximum and minimum speeds attainments are treated. The requirement is that the control system and aircraft allow the pilot to reach V_S and V_{max}. This requirement was found achievable with the design presented in section 5.3.3 since that design was carried out in a near stall situation, what ensures that the control system can manage to control the aircraft at near stall speeds.

This can be evidenced by studying gain scheduling tables in 5.3.4 where optimal and stable designs were obtained for all points in flight envelope, choosing as the most stable one the presented in section 5.3.3. That was verified by actual simulations near stall speeds.

Regarding V_{max} requirement, it is a condition that depends more in the power plant capabilities, although control system also plays an important role. Again, gain scheduling design is encouraged to see stability in all flight envelope of the chosen design in section 5.3.3.

Regarding section 3.2.3.2 of specification, V-n diagrams were unknown for the airplane used since most the key load factors are not known. But information about maximum (positive) load factors in all flight envelope is available as in Figure 8-2. This information is evaluated in the set-points of Figure 8-1 resulting in the Table 8-7.

Set-point	$\boldsymbol{n_{max}}$
1	2
2	1.7
3	3.25
4	4.04
5	5.4

Table 8-7 Max load factors at set-points

These maximums load factors were found possible to reach with the presented control system, where it was found that stall protection was needed in order to avoid the pilot stall the aircraft while maintaining maximum load factors at a given flight condition.

An example of how these load factors are achieved is in Appendix B.3 where an example of set-point 5 is shown. In this appendix, a pull-up manoeuvre is shown achieving +6g's, where the normal acceleration in the figure shows $(n-1)\cdot g\ \ m/s^2$.

Note that maximum load factor is limited by the stall protection system in the controller implemented.

8.1.1.4 Specification section 3.3: Lateral-directional flying qualities

Lateral-directional flying qualities requirements are presented in section 3.3 of the specification.

This section has three main parts that affects directly to the required performance of the control system as presented below. Rest of the sections are found to be related with stick forces and physical implementations issues.

8.1.1.4.1 Specification section 3.3.1: Lateral-directional modes characteristics

This section states requirements regarding Dutch-roll, roll and spiral modes mainly.

As in previous paragraphs, since the Control system is being tested with the aircraft, it is considered that the responses of the Controller plus Aircraft is the natural response of the system, assessing the responses of these two together against sub-sections of section 3.3.1 of specification. So, by assessing these "natural" responses, flight control systems performance is being evaluated.

The main sections inside this point are found to be:

- 3.3.1.1 Dutch-roll mode characteristics
- 3.3.1.2 Roll mode characteristics
- 3.3.1.3 Spiral mode oscillation

Regarding section 3.3.1.1. In specification about Dutch-roll mode, specification states that the performance in terms of natural frequency and damping ratio of the response should be evaluated when yaw disturbances are injected. For that reason, rudder perturbance was injected in the control signal, observing the response. Note that this test was performed at trim flight by observing pitching and yaw rates.

These requirements are shown in Table 8-8:

Level	Flight Phase Category	Class	Min ζ_d*	Min $\zeta_d\omega_{nd}$* rad/sec.	Min ω_{nd} rad/sec.
1	A (CO and GA)	IV	0.4	-	1.0
	A	I, IV	0.19	0.35	1.0
		II, III	0.19	0.35	0.4**
	B	All	0.08	0.15	0.4**
	C	I, II-C, IV	0.08	0.15	1.0
		II-L, III	0.08	0.10	0.4**
2	All	All	0.02	0.05	0.4**
3	All	All	0	0	0.4**

Table 8-8 Spec 3.3.1.1 section requirements

By observing the responses of the perturbed responses and using the logarithmic decrement method, Table 8-9 was obtained.

Set-point	$\boldsymbol{\zeta}$	$\boldsymbol{\omega_n}\left(\boldsymbol{rad/s}\right)$	$\boldsymbol{\zeta \cdot \omega_n}\left(\boldsymbol{rad/s}\right)$	**Rise time (s)**
1	0.44	10.81	4.75	0.24
2	0.23	29	6.67	0.16
3	0.84	49.6	40.18	0.25
4	0.39	26.35	10.28	0.78
5	0.9	57.61	51.85	0.27

Table 8-9 DR response parameters at set-points

Table 8-9 shows that all performance of all set-points complies with specifications for Category B but for Category A set-point 2 and 4 do not meet. So, if Category A requirements are intended to be complied, Gain scheduling technique would be needed.

Section 3.3.1.2 in specification states the requirements for roll mode in terms of time constant of the response of roll rate of a step input.

Set-point	**Rise time (s)**	$\boldsymbol{\tau_r}\ (\boldsymbol{s})$	**Overshoot (%)**	**Level 1 Class IV**	
1	0.38	0.177	1st order response	Maximum $\boldsymbol{\tau_r}\ (\boldsymbol{s})$	
2	0.21	0.095	1st order response	Cat A	Cat B
3	0.4136	0.1882	1st order response	1	1.4
4	0.54	0.245	0.77		
5	0.464	0.2114	0.36		

Table 8-10 Spec Section 3.3.1.2 compliance

Table 8-10 shows performances of roll responses for the different established set-points. It can be seen that the controller meet widely with the requirements. Example of these responses can be found in Appendix B.4.

A peculiar thing when observing Table 8-10 is that the best performances belongs to those points closer to the designing point in section 5.3.3, where these points are the flight conditions where dynamic pressure is lower than in set-points 3 or 4, where performance is worse. This is a non-intuitive effect, since set-points 3 and 4 were expected to get the best performance. This makes evident that the design is not optimal, regardless it meets the requirements. So, gain scheduling may be needed to reduce the performance in those points it is excessive and improve it in those points it is not good enough.

Regarding sections 3.3.1.3 and 3.3.1.4 of specification there were not observations of spiral mode or coupling between any spiral behaviour and roll axis. So, these sections cannot be assessed.

8.1.1.4.2 Specification section 3.3.2: Lateral-directional responses characteristics

Section 3.3.2 of specifications states requirements of the lateral-directional response in terms of responses to atmospheric disturbances and in terms of allowable roll rate and bank oscillations, sideslip excursions, roll control forces and yaw control forces.

The most important point there are section 3.3.2.2 minimum values of the second peaks in % compared with the first peak of the roll rate response. Given the fact the responses given by the proposed FCS do not present these peaks this point cannot be assed. So, this section is complied. See Table 8-10 and Appendix B.4.

A similar situation is presented for section 3.3.2.3 where roll rate and bank angle responses oscillations are bounded. See page 25 of MIL F8785C.

Regarding roll rate oscillations most of the responses presented no oscillations and those that presented, their overshoots were less than any maximum value stated in this section. So, this section is complied. See Table 8-10, Appendix B.4 and Appendix B.5.

In terms of bank angle, due to the nature of roll rate responses, there was no observation in bank angle responses of overshoots or any other oscillation. So, in terms of bank angle oscillations, the specification is met. See Table 8-10, Appendix B.4 and Appendix B.5

8.1.1.4.3 Specification section 3.3.4: Roll control effectiveness

In section 3.3.4. of specification roll performance requirements are presented in terms of bank angle change in a given time.

Section 3.3.4.1 of specification states maximum roll times to roll a certain bank angle for different flight conditions for Class IV airplanes, as shown in Appendix B.5.

The chosen speeds to evaluate these conditions are presented in Table 8-11 and in Figure 8-5.

Cat	VL ($^{m}/_{s}$)	L ($^{m}/_{s}$)	M ($^{m}/_{s}$)	H ($^{m}/_{s}$)	Altitude
A	98	120	180	420	4000m
B	250	280	363	480	6000m

Table 8-11 Spec section3.3.4.1 speeds

Note that in specification only speed ranges are stated, so altitudes of 4000m and 6000m were chosen as representatives as shown in Table 8-11.

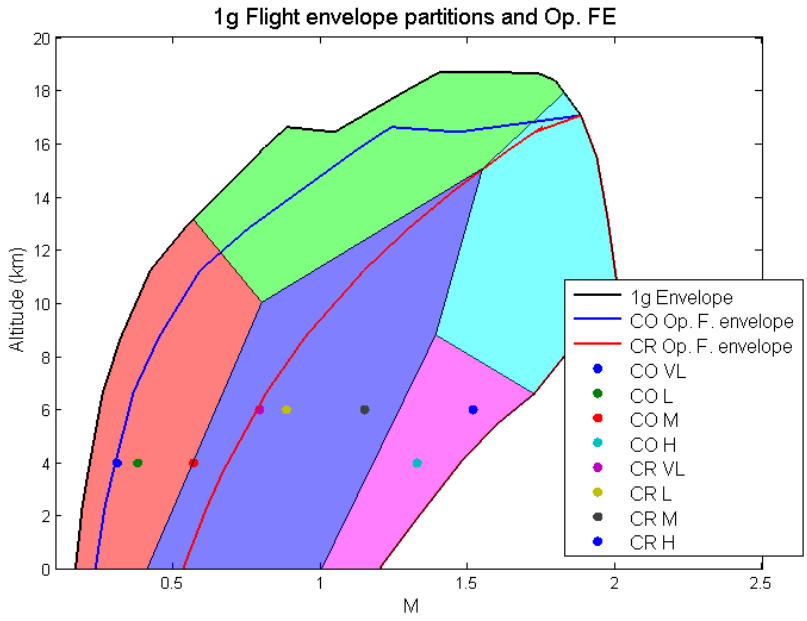


Figure 8-5 Roll bank angle speeds in FE

Performing the required set of tests at flight conditions of Table 8-11, the performances found are presented in Table 8-12:

Speed range	Cat A			Cat B
	30º	50º	90º	90º
VL	0.95			1.7
L	0.9			1.63
M			1.3	1.64
H		0.89		1.28

Table 8-12 Bank angles roll performance

Comparing Table 8-12 performance times with table IXb (Cat A and Cat B) in section 3.3.4.1 of specification, it is found that the control system complies with this specification section (see Appendix B.5).

Section 3.3.4.1.1 of specification states requirements for Class IV airplanes Category A in CO phase, which is being assessed. These requirements are also in terms of maximum rolling times, as found in Appendix B.7: Table IXb in specification.

Design presented in section 5.3.3 directly does not comply with these requirements mainly because the limitations in the pilot inputs imposed. Relaxing these constrains in pilot inputs up to $\pm 250\ ^{\circ}/_{s}\ and\ \ 1000\ ^{\circ}/_{s^2}$ in the mentioned roll channel the performance found is presented in Table 8-13:

Speed range	CO phase			
	30º	90º	180º	360º
VL	0.95			
L		0.95	1.72	3.88
M		0.56	1	2.1
H		0.38	0.81	1.71

Table 8-13 Bank angle roll specs for CO phase

So, comparing Table 8-13 with Appendix B.7 it is found that the design also comply with these requirements. Note that these specifications are not only about the FCS design but about aircraft design and capabilities.

That relaxation of limits to the pilot is not recommended at all but it might be needed for CO phase.

8.1.2 Adaptation performance

In this section, performance of the proposed controller when adapting deviations, fault and damages is assessed.

The main attractive point of adaptive control systems for a control systems designer is that not modelled effects in the model can be compensated by the adaptive NN. These effects can be seen as deviations that can occur in flight or even off-flight, for instance mass properties variations.

In this section mass property variation case is taken into account to present how the Adaptive controller adapts this kind of deviations.

Also, a reduction of the performance through control surfaces faults case is presented, where, once again, the adaptation will be valuable when accommodating the fault.

The scenario in which results in this section are presented is at 4000m and 200m/s level flight starting with the base-line aircraft without any deviation, fault or damage.

8.1.2.1 Deviations accommodation

The case here presented tries to simulate a not modelled loss of fuel tank.

This case can be seen as the situation where the airplane is needed to incorporate new fuel tanks what were not taken into account when modelling the plant. This situation can occur either on-flight or off-flight, where adaptive control can provide the adaptation needed.

These new fuel tanks incorporation is going to be carried out in mid-flight, something not realistic but it will make evident how adaptation is carried out, in the same way it could be done off-flight.

New fuel tanks mass is set as 500kg each located at 3.5m of the symmetry plane of the aircraft, provoking changes in mass, moments of inertia, drag, pitching moments and CG location. Estimations of these changes are presented in Table 8-14.

$\Delta J_x(\%)$	ΔJ_z (%)	Δmass (%)
$\frac{2 \cdot 500 \cdot 3.5\text{^}2}{J_x} = 27.2\%$	$\frac{2 \cdot 500 \cdot 3.5\text{^}2}{J_z} = 14.3\%$	10.8%
ΔC_D	ΔC_{M_0}	x_{cg}
0.02 (Note usually $C_{D_0} \cong 0.013$)	-0.03 (Note usually $C_{M_0} \sim -0.054$)	0.35

Table 8-14 Estimated new fuel tanks changes

Note that these changes are conservatives in order to provide a good margin of change, which will mean bigger performance degradation than a real case. Note as well, that normal CG position is 0.3 of CMA which means that CG is displaced a 5% of CMA backward. This parameter is very sensitive in aircraft, so 5% change means a great change.

The scenario to test this case is the stated flight condition when manoeuvring with steps in AoA and roll, showing how the performance of the FCS without deviation is. Then, deviation in injected, where the NN will adapt and there will be a response in terms of AoA and load factor. Then, again the same steps commands are repeated where difference of performance should be visible.

Figure 8-6 shows the responses to these commands, where the injections were placed about 13 seconds. It can be seen that there are changes in AoA and normal acceleration, being the first one immediately compensated. Regarding yaw and roll channels there is no response since the change is completely symmetric.

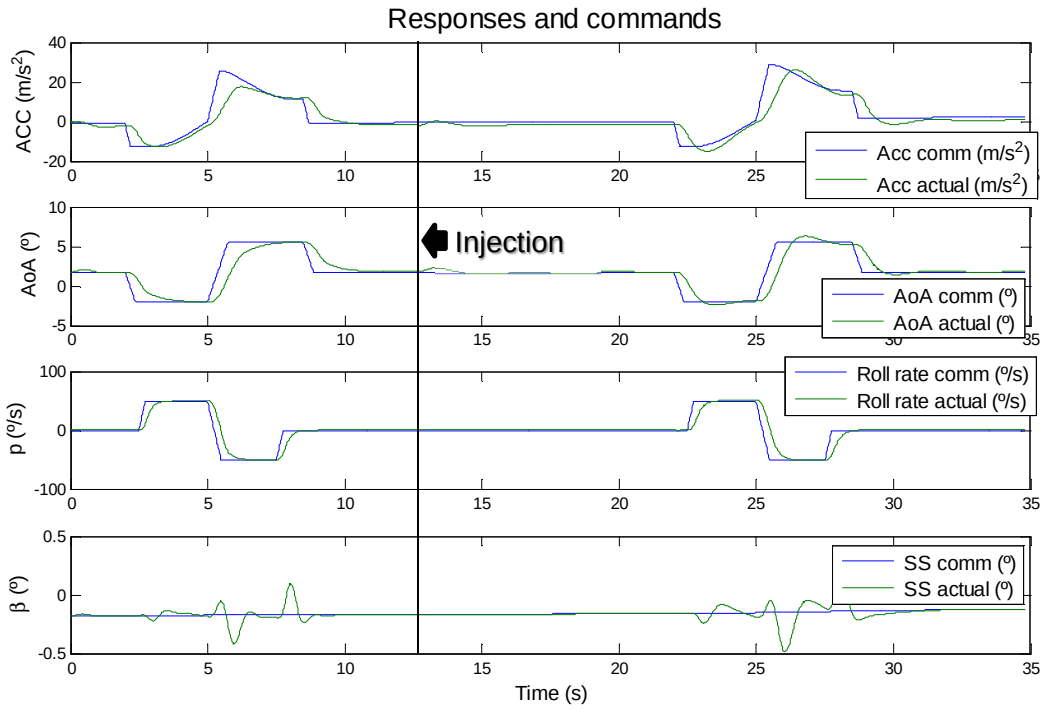


Figure 8-6 Mass properties changes responses

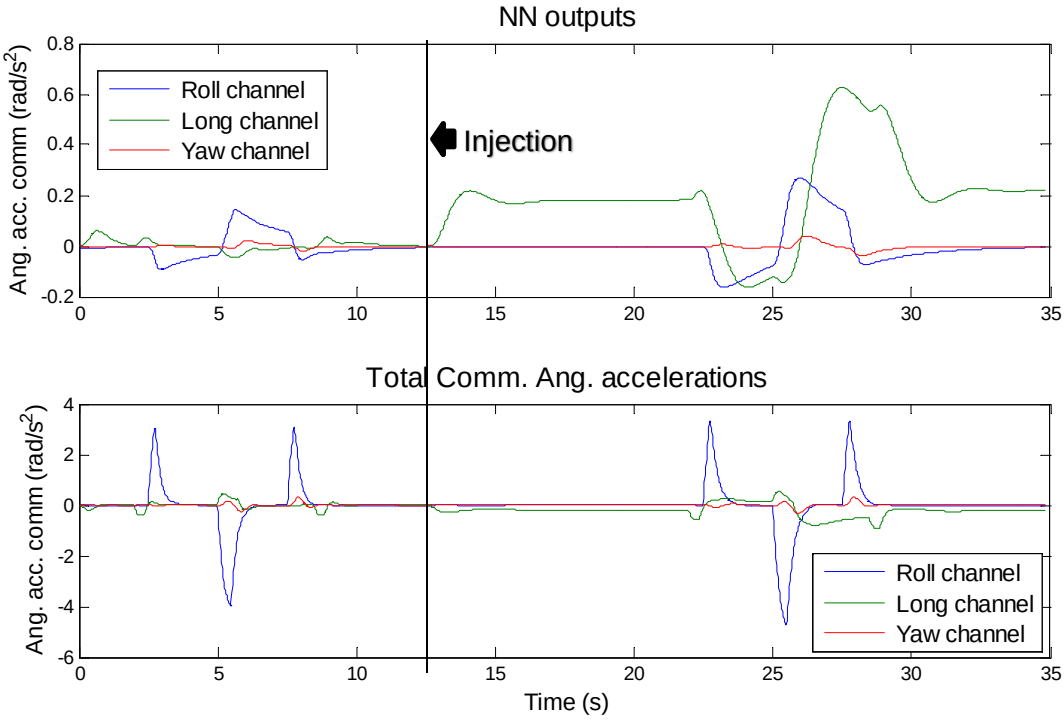


Figure 8-7 Mass properties changes commands adaptation

It can be seen in Figure 8-6 as well that after the injection, performance of responses of AoA and roll commands are degraded and how they get better in successive manoeuvres and time since adaptation takes place.

Figure 8-7 shows the outputs of the NN v_{ad} and the total commands v. Adaptation here is more evident and the real effect of NN is made evident. Right after the injection, due to the displacement of the CG and pitching moment change, NN starts compensating the longitudinal channel providing a bias to the total angular accelerations commanded. As well, NN command in longitudinal channel increases when responding to step command, helping to the reference model and controller to deal with the CG displacement.

Roll channel command of the NN in Figure 8-7 does not show any change when injection happens, due to the symmetry of the changes, but when manoeuvring in roll, the outputs rapidly change and increase. Even, there is change in the response from step to step responses after the injection. That is the learning process of the NN in action, compensating the fact of a bigger angular moment and the need of higher angular acceleration.

8.1.2.2 Damage/fault receiving

The most typical case in adaptive control system designing is the case of loss of a control surface or loss of effectiveness of it. The idea is to see how the adaptive NN can compensate the loss of a control surface while manoeuvring for example.

Since something similar is performed in section 9 here only response of the NN is observed and presented.

The case treated here is the loss of an aileron while performing roll commands so that the manoeuvrability is reduced.

Figure 8-8 shows the response of the roll steps with and without fault where the injection takes place at second 5. It can be seen how the performance is reduced since the steady errors take more to be corrected. The effect of adaptation is precisely to eliminate these steady errors that are caused by the error that NDI is producing, since in this basic concept no FDI is used.

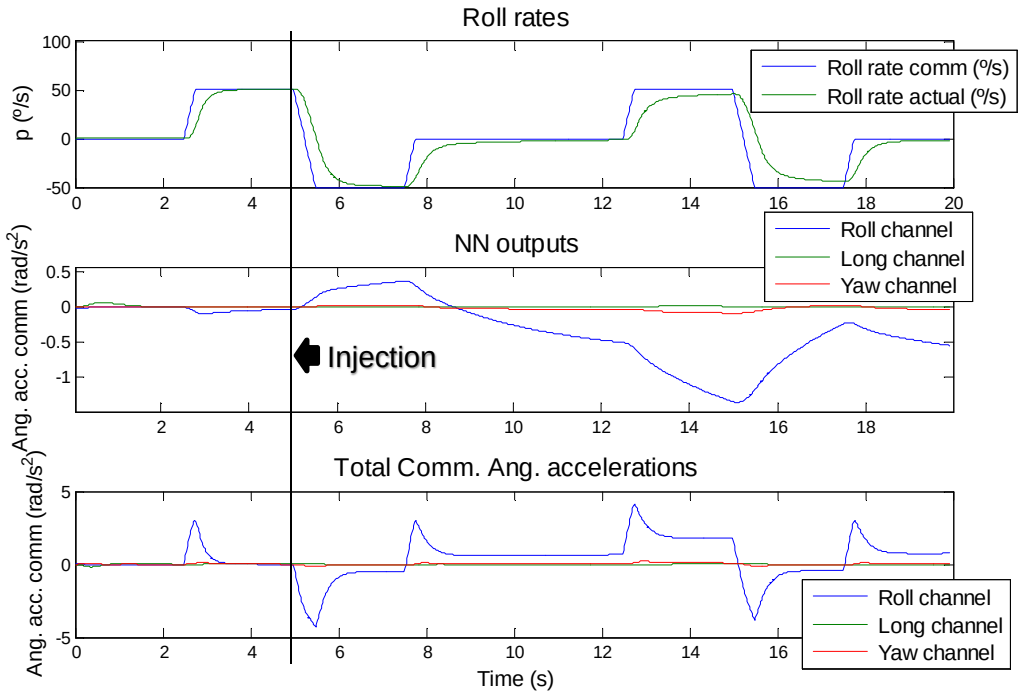


Figure 8-8 Aileron loss Adaptation

It can be seen that adaptation provides a variable bias in the total angular acceleration command that eliminates the steady error.

So, finally it can be confirmed, as said in section 6.3, that the effect of adaptation is to correct steady errors regardless their cause: “adaptive integral action”.

8.2 Advanced concept

Advanced concept performance is presented here where only reconfiguration and adaptation features are shown, leaving the performance for section 9 where performances are compared. The intention is to show how the reconfigurable and adaptive controller reconfigures the control effort and accommodates faults.

The scenario contemplated is at 4000m and 200m/s level flight starting with the base-line aircraft without any deviation, fault or damage.

The case here presented involves several manoeuvres in roll, where it can be seen the performance after a blockage of both ailerons while performing a roll manoeuvre, which means the ailerons are blocked in a non-zero deflection.

Figure 8-9 shows the results of these cases. Firstly a positive roll input is commanded when, right before start roll negatively, the blockages are injected, letting the ailerons

blocked at that positions, as it can be seen in Figure 8-9. After the injection FDI system starts noticing of the blockages and sends this information to NDI and CA, who reconfigure their laws to achieve complete reconfiguration. This fact can be noticed by observing elevators deflections that right after the fault injection adopt an asymmetrical configuration, compensating the blockages of the ailerons.

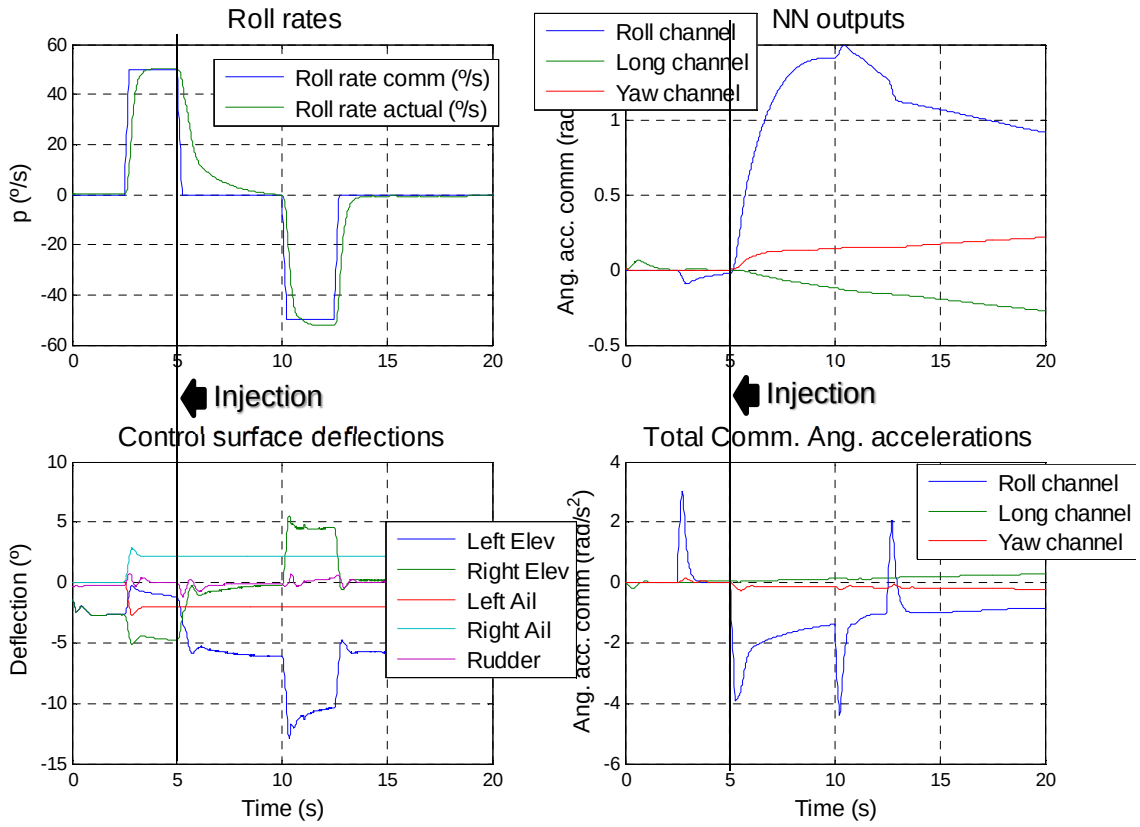


Figure 8-9 Advanced concept reconfiguration

Regarding the roll performance, it can be seen that right after the failure it is noticeably degraded, but this roll manoeuvre would help to detect the faults and, in consequence, to reconfigure NDI and CA laws. After a period of time it can be seen that performance is almost completely recovered, being as big as the airplane can provide with the given faults.

Regarding the NN and control commands, it can be seen that right after the injection NN starts compensating in roll channel the deviation and, due to the reconfiguration, this adaptation starts to be not necessary inducing a bias in the response. So that, NN start reducing its compensation in roll channel. It is interesting how the NN compensates in

yaw and longitudinal channels due to the coupled effect of aileron deflections in those channels.

An important point to clarify here is that adaptive NN does not play a key role when reconfiguring, although it provides a valuable stabilization, where SI, FDI, NDI and CA do play a key role, reallocating the commanded control effort among the remaining surfaces and/or, in the appropriate case, adapting their control laws to any damaged airframe.

9 Performance comparisons

This section presents performances of the Basic and Advanced controller in comparison with classical approaches which usually are implemented in FCS. Also a comparison between Basic and Advanced concepts is presented in order to present fault-tolerant capabilities of each concept.

9.1 Classical controller vs. basic concept

Basic concept controller presented in this work presents two features that distinguish it from classical approaches.

The first one lies on the fact the it is based on Explicit model following philosophy through an Reference model or desired dynamics and NDI and CA. This means that the controller will follow, as long as it is possible, the desired performance. That is, the designer implements a required performance in terms of time or frequency, and the controller will follow it.

Other important characteristic of this first feature is that, since Dynamic Inversion is being carried out, the controller takes into account Mach and Altitude effects in the plant, providing the appropriate control signals for the given flight condition. This is something that in classical control approaches must be done by hand, designing all parameters of the control system for every flight condition, what means lots of effort.

The second most important feature of the presented concept is that it implements, along with the Explicit model following scheme, an Adaptive Neural Network that provides an adaptive framework to perform control and provides stability when the aircraft suffer from any unforeseen or not modelled deviation, fault or damage, that might prevent a classical approach or even the Explicit model following approach from performing correctly. This unforeseen deviation, connecting with the previous point, can be seen as a miss-modelling of the plant in terms of Mach and Altitude, so that Adaptive NN adapts these deviation what a classical approach would not be able to do.

So, this adaptive feature allows to the designer to reduce the effort in plant modelling as well as the effort when designing and tuning the controller, since any miss-modelling or not efficient design would be compensated by the NN.

In this section comparisons between a classical design based in PID controllers and the Basic concept in terms of off-design point performance and performance under deviations and faults are presented

9.1.1 Off-design performances

In this section a classical controller approach and the Basic controller are compared in terms of performance in roll responses. The classical controller used is based in multiple PID controllers. This controller also have the option of perform Gain scheduling of the PID controller gains over all flight envelope.

The purpose of the comparison is to make evident how a classical approach performance degrades out of the design point whereas Basic concept presented maintain a certain performance. Also, the Classical controller can perform Gain scheduling, what helps to improve the responses along different points in flight envelope, but designing of Gain scheduling of so many parameters means a lot of effort and time.

The comparison is carried out performing roll manoeuvres since it was found the most comparable manoeuvre given the performance in longitudinal axis of the classical controller. The comparison consists on:

a. Design both Classical and Basic controllers in a design point: $H = 4000m \; and \; M = 0.85$ evaluating their performances
b. Evaluate those designs out of design point: $H = 4000m \; and \; M = 0.37$ evaluating their performances

Regarding the design of controller in design point (a), classical controller was design so that it offers the best response at that point but also provides stability in point (b). Basic concept controller design of roll channel was chosen as the in section 5.3.3 since they were found to give same performance in almost all flight envelope.

Table 9-1 presents performances in terms of rise time found and Figure 9-1 shows the responses of both controllers. Classical controller responses shown are on-design point and off-design point both using no gain scheduling and using it. The purpose of shown performance of gain scheduling in (b) is to let to be compared this performance with the obtained if no gain scheduling is used in (b) as well.

Note that, since performances of both controllers are very different, comparison is not carried out between controllers but between performances of same controller in different flight conditions (a) and (b).

	Flight condition	Rise time(s)	Degradation (%)	Steady error (º)
Classical controller	On-design	2.44		0.44
	Off-design	NaN	NaN	
	Off-design + Gain scheduling	3.15		
Basic concept	On-design	0.48		
	Off-design	0.40	$-16.6\% \rightarrow 0\%$	

Table 9-1 Performances in rise time comparison

It can be seen in Figure 9-1 that design point response presents a good performance. Evaluating classical controller using Gain scheduling at point (b) the performance obtained is very similar, although there is a degradation of the rise time due to the lower dynamic pressure. The fixed design evaluation off-design provides a worse response since it does not even converge to the commanded step and there exist a non-corrected steady error.

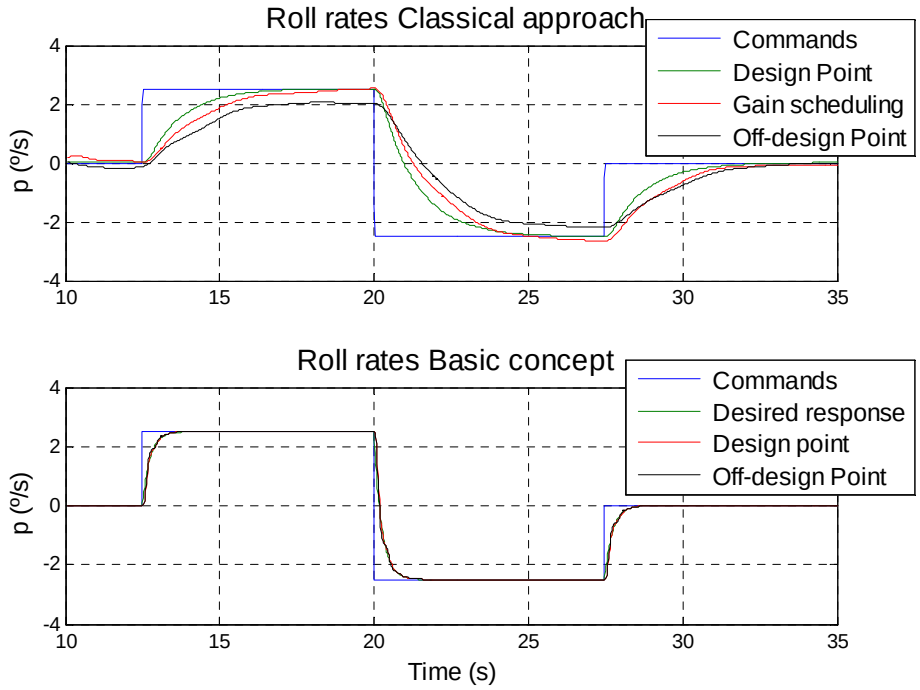


Figure 9-1 Classical and Basic controllers on/off-design performances

Figure 9-1 also shows the performance of the Basic controller, where responses on and off the design point are shown, along with the desired 1st order response implemented in the desired dynamics ($\tau = 0.25$). It can be seen that they three are very similar. Detail of the responses can be seen in Figure 9-2. Note that although off-design response rise time seems to be better, in fact, if close look is taken to Figure 9-2, it tracks worse the desired performance than the on-design response.

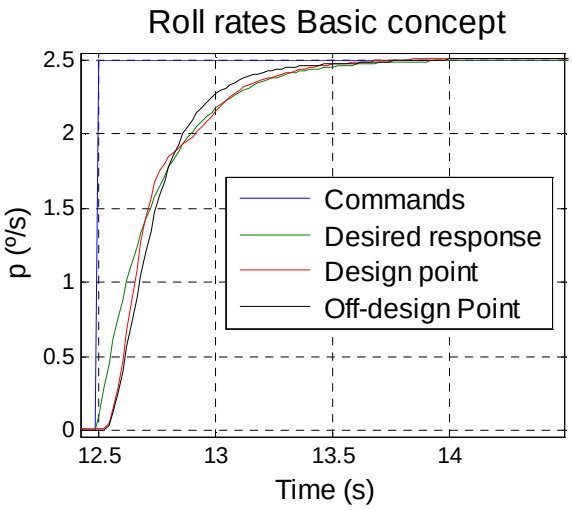


Figure 9-2 Detail basic controller responses

The main reason of the similarity of performances of the Basic approach is that, since it performs Explicit model following through a non-linear inversion, it already takes into account the flight condition to command the adequate control signals that best track the desired performance.

Adaptation is not visible on the presented cases here since there is not a steady error or deviation in the plant responses. The effect of Adaptation might be more visible in following sections.

9.1.2 Performances under deviations, faults or damages

The purpose of these section is to compare again a classical controller and the Basic controller outlining which is the effect of adaptation and it benefits in comparison with the classical approach.

Classical controllers are based on a design model of the plant. Once these controllers are design performance of them are given by the similarity of the actual aircraft and the mathematical representation that was used for designing the controller. So, not only the classical controller has to offer good performance but a robustness margin as well.

The most typical problem that occurs in reality is that the mathematical model of the aircraft shows different variations of behaviour with Mach and Altitude that the real aircraft, what may bring to the controller to perform worse. That is, the aerodynamic model used in the model is not accurate enough.

The case here presented tries to capture this issue by comparing performances at two flight conditions, trying to emulate a deviation in the mathematical model used in design:

a. On-design point: $H = 4000m \; and \; M = 0.85$
b. Off-design point: $H = 4000m \; and \; M = 0.37$ where ailerons roll moment coefficient is 60% less than modelled

Note that at point (b) the aircraft shows a 60% less ailerons roll moment coefficients than modelled what may be caused by a bad modelling. Note also that in reality these errors in aerodynamic models are usually smaller but here the error is exaggerated, so that it can be seen a noticeable difference.

The first results presented are shown in Figure 9-3 where the classical controller with and without gain scheduling and without deviation and with deviation are evaluated in point (b). Also performance at point (a) is presented for comparison.

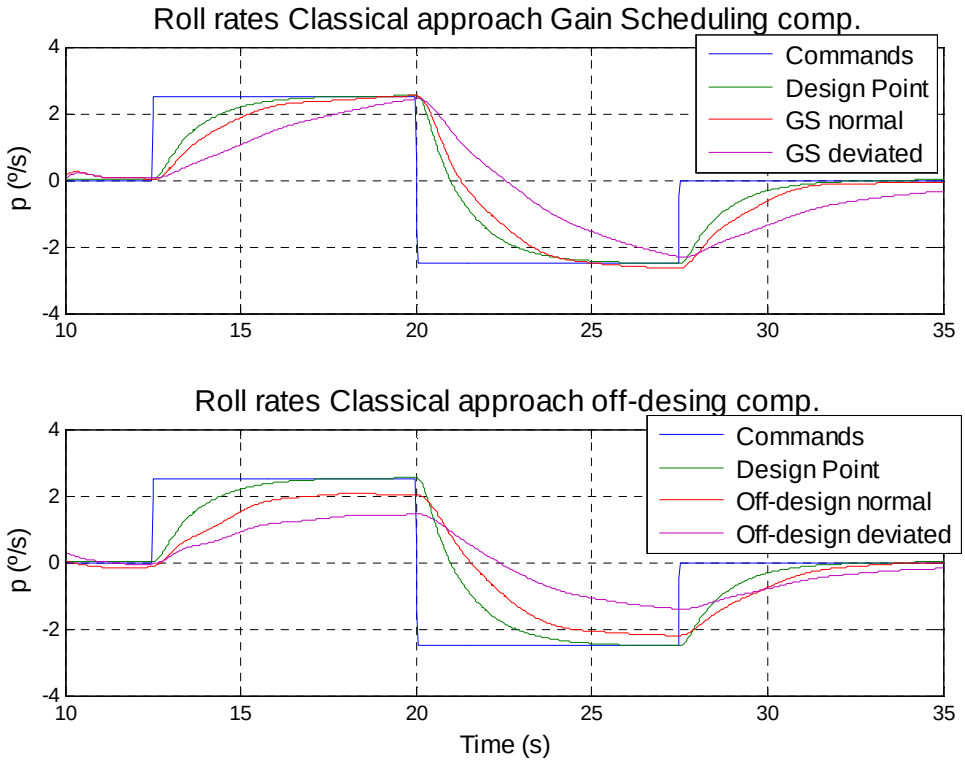


Figure 9-3 Classical controller under deviation

About the classical controller performing gain scheduling can be said that performance gets worse under deviation, although integral action should be bigger for the given case and, that is why it does not perform correctly. Comparing now controller without gain scheduling, evaluation at point (b) reveals that it does not converges but under deviation

the performances is even worse. That makes evident that the classical controller without gain scheduling has not enough integral action. Note rise times comparisons and degradations are shown in Table 9-2.

Regarding the Basic controller Figure 9-4 shows the ideal response and responses at point (b) of the Basic controller and the same but removing the adaptive action. As well, adaptive term given by the NN is presented so that the difference between adaptive and non-adaptive can be better evidenced.

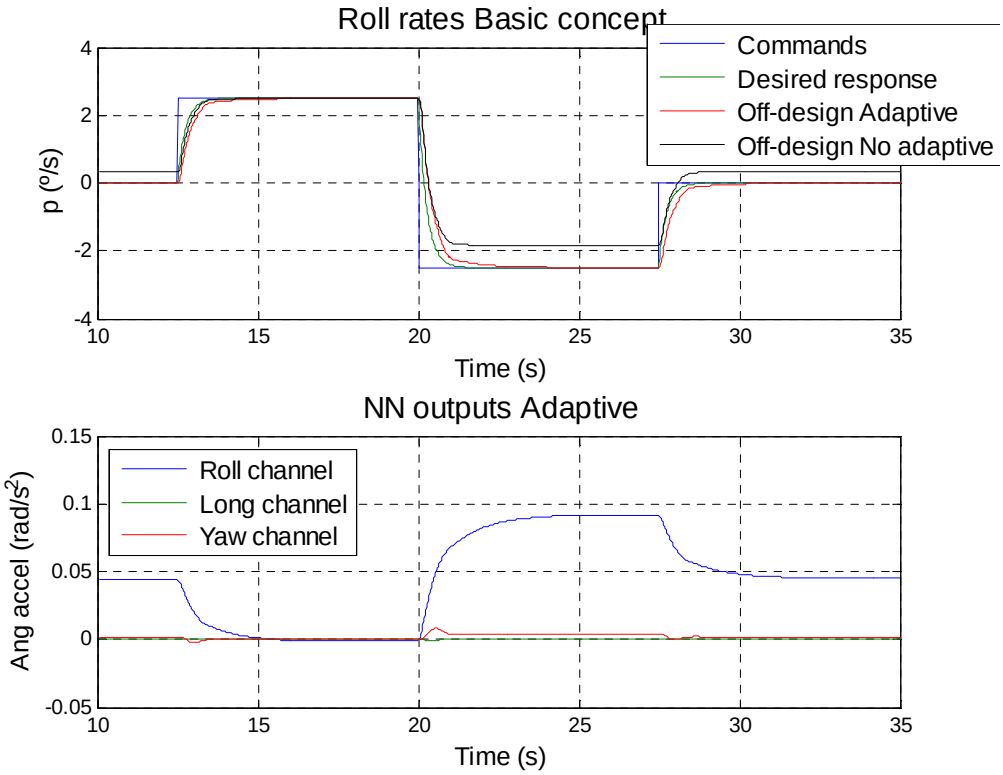


Figure 9-4 Basic controller under deviation

It can be seen in Figure 9-4 that response of adaptive controller under deviation is worse that it should be in terms of rise time and settling time, comparing it with the expected or ideal response. It can be seen in section 9.1.1 that the performance of the controller without deviation is closed enough to the ideal response so that they are considered as equal.

Comparing now adaptive and non-adaptive responses in Figure 9-4 (detail given in Figure 9-5), the effect of NN can be better evidenced. It can be seen that NN acts as an integral action correcting the deviation, something the non-adaptive controller is unable to do since no integral action is used. Taking a look to the NN response it can be seen

that the speed of its response is giving the speed with which roll response converges to the command. Hence, the learning speed or learning rate of the NN determines how fast the errors and deviations are corrected.

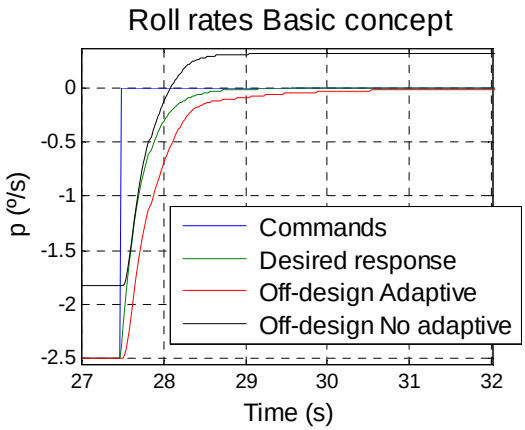


Figure 9-5 Detail Basic controller under deviation

This fact is one of the main advantages of the adaptive controllers since their performance does not depends on flight condition, given deviation or any other variable. Their performance only depends on a single parameter, learning rate. So that, increasing the learning rate, the learning speed is incremented and the convergence time of the responses of the controller under deviations, faults or damages is improved.

Quantitative comparisons are carried out in Table 9-2:

	Flight condition	**State**	**Reached steady value (º)**	**Degradation (%)**	**Steady error (º)**
Classical controller	Off-design	Normal	2.062		0.44
		Deviated	1.43	30.6%	1.07
	Flight condition	**State**	**Rise time (s)**	**Degradation (%)**	
	Off-design + Gain scheduling	Normal	3.15		
		Deviated	5.59	77.46%	
Basic concept	Off-design	Ideal	0.55		
		Deviated Adapt	0.71	29.1%	
		Deviated No adapt	NaN	NaN	

Table 9-2 Quantitative comparison of Classical under deviation

Note that Table 9-2 degradations are obtained, which may not be comparable between Classical and Basic controller due to the fact their rise times are completely different.

Note as well that degradation of the Basic concept is a 29.1% in terms of rise time when compared with ideal response. This degradation can be reduced up to a 9.1% if increasing the learning rate of the NN to 80, making evident the fact that an increase of

learning rate produces and improvement of the adaptive response, regardless the flight condition or deviation given.

With a learning rate of 80 it was found that degradation can be up to 9.1% in responses of the basic concept under damage.

Finally, effects of adaptive NN are emphasized, stating that the adaptation would be equivalent to an integral action that adapts to the flight condition or deviation/fault given, being its learning rate the unique parameter to tune which determines the speed of the adaptation. That means that if any error in modelling exist, any deviation/fault/damage happens or any change in the aircraft is made, adaptation takes care of it through adaptation, not having to really on robustness.

9.2 Classical and Basic controllers vs. Advanced concept: Reconfiguration

In this section Adaptive concept is compared with both Classical and Basic controller in order to show how a fault affects to the performance of the controllers and how the Advanced concept solves it performing reconfiguration.

In order to carry out this comparison the controller are evaluated a same flight condition: $H = 4000m\,, V_{TAS} = 275\,{}^{m}/_{S}\ and\ M = 0.85.$

At this flight condition roll manoeuvres are performed, as in previous sections, and then a fault is injected: right aileron blockage at current position. That is, if the airplane is rolling, the right aileron get blocked at the deflection it was.

Figure 9-6 shows the performances of controllers at the stated flight condition for fault and no fault condition. Classical controller gets degraded it performance in terms of rise time but it still shows convergence. That is due the integral action although, for the given fault, it is not enough and degradation is noticeable. Nevertheless its shows a good relative convergence that enhance the benefits of integral action in classical control.

Regarding Basic controller, responses of the controller with and without adaptation are shown in order to remark once again the adaptation effects and performance. It can be seen in Figure 9-6 and Figure 9-7 that Basic controller successfully compensates the steady error, being the speed of the convergence determined by the NN learning rate. Note that Basic controller without adaptation does not converges.

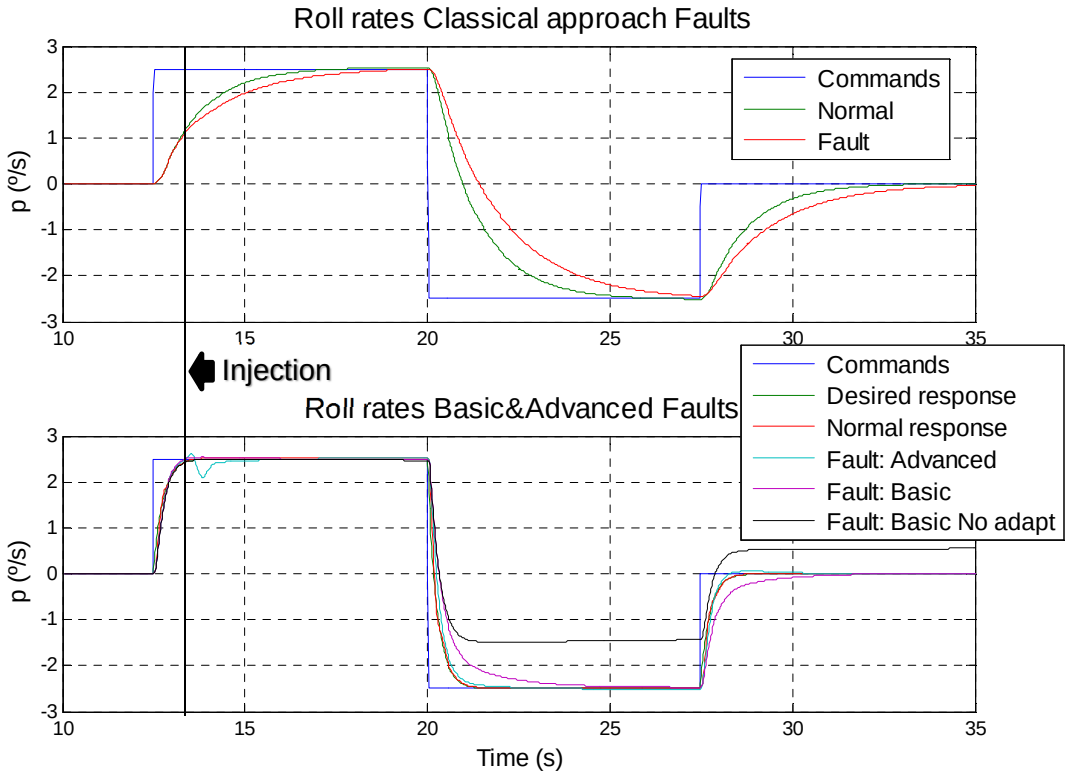


Figure 9-6 Fault responses Classical, Basic and Advanced

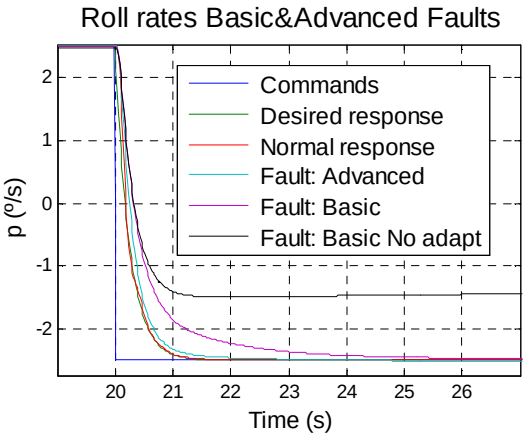


Figure 9-7 Detail Fault responses Classical, Basic and Advanced

The important response in Figure 9-6 and Figure 9-7 is the response of the Advanced controller. It can be seen that when fault is injected FDI system starts identifying it, providing at the beginning wrong estimations that lately converge to the actual values. It is then when Advanced controller is capable of perform control of the aircraft using the remaining control surfaces taking into account the blockage. Hence, as it can be seen in Figure 9-7 the performance is similar to the desired one.

Note that the performance of the Advanced concept would be unchanged regardless the fault or deviation in control surfaces, as long as FDI provides correct information and enough control authority still remains.

A quantitative comparison is carried out in Table 9-3 where rise times as compared:

	State	Rise time (s)	Degradation (%)
Classical controller	Normal	2.54	
	Fault	4.1	61.4%
Basic & Advanced	Normal or Ideal	0.55	
Advanced controller	Fault	0.56	1.8%
Basic Controller	Fault	1.13	103.6%
Basic Controller No adaptation	Fault	NaN	NaN

Table 9-3 Fault, Quantitative comparison Basic, Classical and Advanced

It can be seen that Basic controller degradation is bigger than the Classical controller but it should take into account the actual degradation in terms of seconds. Note that the Basic controller, regardless its degradation it would still comply with some specifications (see Table 8-10).

9.3 Basic concept vs. advanced concept: Adaptation capabilities limits

In this section both of the controllers proposed are compared. The purpose of these comparison is to know the relative capabilities of both controllers and know the limits of the Basic concept.

Basic controller provides stability under uncertainties such as deviations, faults or damages, but something it cannot provide is manoeuvrability under those cases. Advanced concept is intended to provide this remaining manoeuvrability so that the aircraft can be operate using all the remaining control authority.

The case contemplated here is an extreme case such the loss of 80% of right wing surface, loss of right LEF and loss of right aileron. This case can represent perfectly a foreign object impact.

The flight condition contemplated is level flight at an altitude of 4000m and a speed of 170m/s, at which the three damages are injected (5^{th} second in figures). Then, in order to test manoeuvrability, roll manoeuvres and AoA manoeuvres are performed in separated cases.

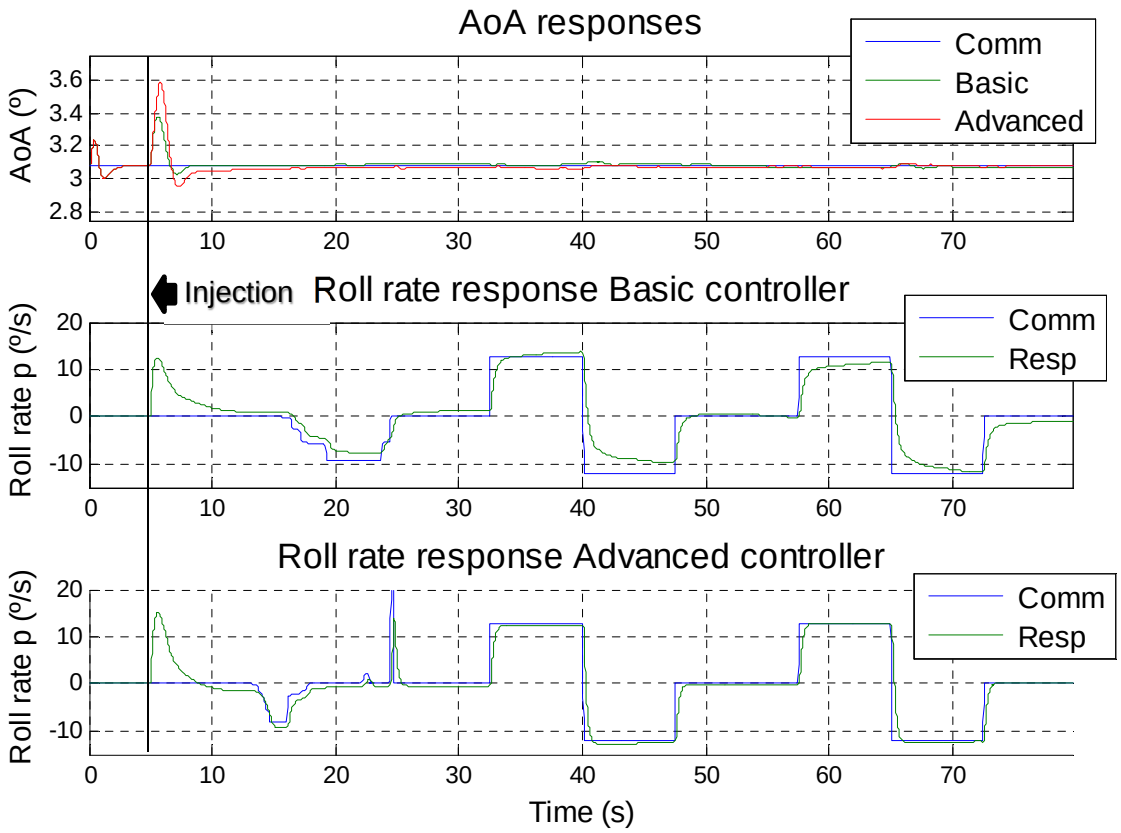


Figure 9-8 Basic and Advanced comparison. Roll manoeuvres

Figure 9-8 shows the first case in which roll manoeuvres are performed after injection. It can be seen that both controller correctly overcome the damage and faults compensating correctly. After injection, the aircraft is levelled so that is why negative roll rate commands appear. After levelling roll commands are performed.

It can be see that basic controller correctly compensates and let to perform roll manoeuvres reaching acceptable roll rates, but due to the severity of the damages, convergence to the step commands takes more time than in sections 9.2 or 9.1.2. It cannot be said that there is no convergence but it can be said settling time maybe too high for a required performance.

Advanced controller in contrast, reallocate remaining control authority so that it can perform roll manoeuvres with a better performance than the Basic controller. As well, SI identifies the current asymmetry so that the control is even more precise. Note that due to the slow convergence of SI and due to the adaptation as well, performance gets better as manoeuvres are executed. As well, due to SI, convergence after injection is better than in Basic concept.

Note as well, that no cross effect is made evident here so that the main difference in performance between Advanced and Basic controller is due to FDI system and reallocation of control effort.

The second case, in Figure 9-9, is the most important comparison since it makes evident the difference in capabilities between the controllers.

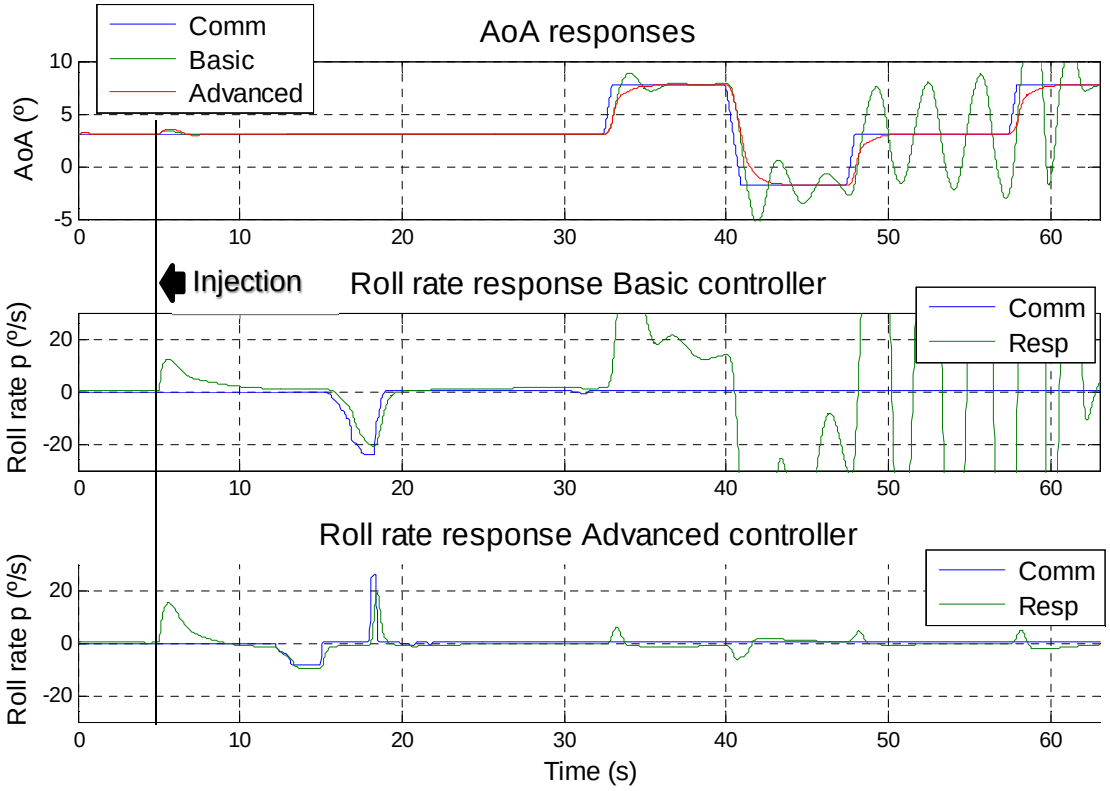


Figure 9-9 Basic and Advanced comparison AoA manoeuvres

In Figure 9-9 AoA commands are performed after the injection where cross response between longitudinal and lateral-directional axes is made evident. When a position increment of AoA is made, due to the asymmetry, a roll response appears, something that makes the aircraft uncontrollable if no SI is used. As seen, Basic controller, that does not implement SI, has cross responses that makes the aircraft uncontrollable in roll axis and unable to perform correctly a symmetric longitudinal manoeuvre.

In contrast, advanced controller uses an accurate prediction from SI system and is able to reduced significantly the cross response, although it is still visible. Note that, in moments after the injection in the Advanced controller, cross responses help the SI system to better identify the asymmetry and converge faster.

10 Conclusions

10.1 Summary

In section 2 exploration of the current state of art in the field of Adaptive and Fault-Tolerant control is carried out, performing an examination of the most important techniques already explored. Also, a selection of those techniques that best fit into the proposed problems was performed, doing a deeper research on those these techniques.

An explicit model following direct adaptive control strategy was chosen as a bottom-line for the construction of the controllers, choosing NDI, CA, Adaptive NN, SI and FDI as the key components of the final solutions.

In section 3 the model of an F-16 aircraft is shown, using this model to test, develop and validate the control systems built. In order to achieve an overactuated model of F-16, the model introduce in section 3.1 is modified, proposing significant changes on the dynamics model and allowing to the control system to control the aircraft using 7 control inputs: left/right elevator, right/left aileron, rudder, left/right LEF.

In order to test and develop the controllers proposed, this last modified model was altered again so that reconfiguration parameters can be set to change most of the parameters of the aircraft dynamics. These parameters are designed so that most of the cases of faults, deviations and damages contemplated in section 1.4 can be represented with different combinations of these parameters.

In section 4 NDI technique is explained and its implementation outlined. Also, for the case of perfect dynamic inversion, stability is proven and model reference (desired dynamics) and controller general law is found. As well, uncertainties of dynamic inversion are considered, introducing the formulation used in section 6 to demonstrate stability of the entire controller.

As well, in section 4, CA techniques are introduced, outlining the most important methods and those used for the actual implementation in this work. Also, implementation of CA and formulation used is introduced.

Finally in section 4, in order to reduce errors in the inversion of non-linear models using CA, a method called “Incremental Control Allocation” is introduced.

In section 5 different techniques to construct a reference model are explained. In this section, two proposals of reference model and controller regarding aircraft control are presented. In this work, non-linear multiple reference model and P controller proposal in section 5.3 is used. Also, in each proposal, formulation of the tracking error dynamics needed and used in section 6 to perform stability proofs are outlined.

Also in this section, a deigning point for the model reference and controller is chosen, exploring the nature of the phenomena that intervenes in the parameters selection. This analysis is also used to justify the eventual necessity of gain scheduling. As well, a gain scheduling design of the longitudinal channel of the controller proposed is presented in section 5.3.4.

In section 6 Adaptive NN are introduced, performing a general introduction to the theory of NN and to the concept of Adaptive NN. As well in this section, several architectures of NN of different number of layers and different features (e.g. regularization feature) are presented along stability analysis. As a generalization of those, a general architecture adaptive NN algorithm for any number of layers and neurons is proposed in section 6.2.3 along with its stability analysis.

Finally in section 6, an anti-windup implementation is implemented to work along with the NN, where two methods are proposed and working principle shown.

In section 7 a SI general theory is outlined, exploring the most common techniques and methods. As well, an algorithm to perform SI by using an UKF algorithm is presented. Selection of an UKF algorithm to perform this task was motivated due to the fact it is possible to assess the quality of the estimations.

Also in section 7 explanation of the methods and resources used to simulate a FDI module are explained.

In section 8 a validation and analysis of the two controllers proposed, Basic and Advanced controllers, is carried out. Firstly, a specification compliance assessment is perform using the specification MIL F-8785C. It was found that the controllers implemented comply with specification in all the sections of the specification that was possible to evaluate the solutions.

Note that, due to the restriction in the reference model implemented in equation (5-7), the fixed design of section 5.3.3 did not comply completely with SPPO requirements. It was demonstrated in section 8.1.1 that by relaxing the condition in equation (5-7) it is

possible to fulfil all the requirements of the specification. Also, in section 8.1.2 adaptation performance of the Basic controller is shown by two test:

- Deviations accommodation: A test of a hypothetical case of sudden incorporation of two new fuel tanks was evaluated
- Damages/faults receiving: Loss of an aileron while performing roll manoeuvres

Results of these tests show that Basic controller performs adaptation correctly, successfully stabilizes the aircraft and provides the required manoeuvrability.

Finally in this section, Advanced controller is subjected to the case of loss of ailerons while performing roll manoeuvres, showing an adequate reconfiguration of control surfaces.

Finally, in section 9 comparisons between classical control approaches, such as PDI basic control systems, and Basic controller are carried out. Comparisons conducted include an off-design capabilities test and a performance under deviations/faults evaluation. These comparisons show the superior capabilities of the Basic controller.

Also, Basic and classical controllers are compared with the Advanced one in order to show the advantages of the Advanced controller under faults or deviations.

Finally in this section, a comparison between Basic controller and Advanced one is carried out under the case of loss of 80% of right wing and loss of right aileron and LEF. This test is intended to delimitate the capabilities of the adaptation feature under deviations, faults and/or damages, motivating the adequacy of the Advanced controller in such extreme cases.

10.2 Contributions

It is considered that contributions of the following points have been innovative in the field of Adaptive and Fault-Tolerant control:

- Modification of the model to achieve 7dof model
- Modification of that model to obtained reconfigurable model
- Incremental CA
- General architecture NN and proof
- Anti-windup implementation along with adaptive NN
- SI with quality estimation: usage of UKF

10.3 Potential applications

Solutions proposed in this work can be applied in a big variety of platforms.

Regardless this work was oriented to manned aircrafts, the solutions proposed can be perfectly applied to Unmanned Aerial Vehicles in a similar way they were implemented in this work.

Applicability to unmanned systems can be attractive since the fact the controllers proposed can provide the capabilities to UAVs to pursue the mission despite the fact of receiving damages. As well, other interesting capability is the possibility of recover the aircraft even if it is damaged.

Regarding manned aviation, the solutions proposed have a clear advantage in comparison with actual systems. Adaptive and Fault-Tolerant controllers can provide to manned aircrafts not only not carry out with their missions, (military aircrafts) but to perform a safe landing, eventually saving lives.

Benefits of the proposed architectures are their flexibility and ability to reduce costs of development and operation. As well, they provide the capabilities to accommodate changes, faults and damages on ground or in mid-air, letting the aircraft enough manoeuvrability as physically possible and, eventually, a safe landing.

The possible applications of the methods used and developed in this work are virtually unlimited inside the field of Adaptive and Fault-Tolerant control field. Generally, they are applicable to any application where control methods are applied and cost, effort and complexity of the development are needed to be reduced.

Also, they are applicable to any application where there exist uncertainties on the plant and failure of the control system is critical. As well, in applications where there may be faults or deviations in plant and failure of the control system is critical.

10.4 Future Work

The presented work includes many algorithms and techniques but there are still many improvements, changes and developments that may be carried out:

- Improve the model modifications to achieve a greater level of reality
- Test new algorithms to perform CA, achieving a simplification of the current implementation and improving its efficiency

- Relax equation (5-7) constrain in the implemented design of parameters of the reference model
- Design a gain scheduling without equation (5-7) constrain
- Design a gain scheduling technique for all channels and not only in longitudinal axis
- Study and implement new and more efficient ways of perform anti-windup working along with the NN
- Implement a SI exciter in order to perform a control of estimations quality
- Implement a logic or decision making system to manage the information from SI to NDI and CA considering quality of information
- Implement an FDI system that allows identification of blockages, loss of effectiveness and total loss (or floating) of control surfaces.
- As well, implement a system that evaluate the degradation of performance or estimates the remaining control authority under cases of deviations, faults or damages receiving. This is needed to implement a limiter system that limits the inputs to NDI and prevent an inappropriate performance of anti-windup system
- Develop an information management system that decides when to pass information from FDI to NDI and CA systems, based on quality of FDI estimations

Generally, it is considered that, any further development of the techniques here presented would be approached, exploring different techniques such as Backstepping.

Finally, the most challenging future aspiration of this work is: implement and apply controllers here presented or their improvements to real platforms.

REFERENCES

[1] S. Aloni, Israeli F-15 Eagle Units in combat, Osprey, 2006.

[2] K. J. Aström and B. Wittenmark, Adaptive Control, Addison-Wesley, 1989.

[3] K. S. Narendra and L. S. Valvani, “Direct and Indirect Model Reference Adaptive Control,” *Automatica,* vol. 15, pp. 653-664, 1979.

[4] I. Hwang, S. Kim, Y. Kim and c. Eng Seah, “A survey of Fault Detection, Isolation and Reconfiguration Methods,” *IEEE transactions on control systems technology,* vol. 18, no. 3, 2010.

[5] M. Bodson and J. E. Groszkiewicz, “Multivariable Adaptive Algorithms for,” *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 5, no. 2, 1997.

[6] M. Steinberg, “A Historical Overview of Research in Reconfigurable Flight Control,” Patuxent River.

[7] R. Montoya, W. Howell, W. Bundick, A. Ostroff, R. Hueshen and C. Belcastro, “Restructurable Controls,” 1982.

[8] C. Huang and R. Stengel, “Restructurable Control Using Proportional-Integral Implicit Model,” *Guidance, Control, and Dynamics,* vol. 13, no. 2, 1990.

[9] W. Morse and K. Ossman, “Model Following Reconfigurable Flight Control System for the AFTI/F-16,” *JGCD,* vol. 16, no. 6, pp. 969-976, 1990.

[10] P. Maybeck, “Application of Multiple Model Adaptive Algorithms to Reconfigurable Flight Control,” *IEEE Transactions on Aerospace Electronic Systems,* vol. 27, no. 3, pp. 470-480, 1991.

[11] F. Ahmed-Zaid, P. Ioannoun, K. Gousman and R. Rooney, “Accomodation of Failures on the F-16 Aircraft using Adaptive Control,” *IEEE Control Systems Magazine,* vol. 11, no. 1, pp. 73-8, 1991.

[12] Z. Gao and P. J. Antsaklis, “Reconfigurable Control System Design via Perfect Model-Following,” in *AIAA Guidance, Navigation a nd Control Conference*, 1991.

[13] P. Chandler, M. Pachter and M. Mears, “System Identification of Adaptive and Reconfigurable Control,” *JGCD,* vol. 18, no. 3, pp. 516-24, 1995.

[14] J.-E. Bobrow and W. Murray, “An algorithm for RLS Identification of Parameters that Vary Quickly,” *IEEE Transactions on Automatic Control,* vol. 38, no. 2, 1993.

[15] R. Eberhardt and D. Ward, “Indirect adaptive flight control of a tailless fighter aircraft,” 1999.

[16] D. Ward, R. Barro, M. Carley and T. Curtis, “Real-Time Parameter Identification for Self-Designing Flight Control,” *NAECON,,* no. 1994.

[17] B. Kim and A. Calise, “Nonlinear Flight Control using Neural Networks,” 1997.

[18] M. McFarland and A. Calise, “Multilayer neural networks and adaptive nonlinear control of agile anti-air missiles,” in *AIAA Guidance, Navigation, and Control Conference*, 1997.

[19] A. Calise and R. Rysdyk, “Adaptive model inversion flight control for tiltrotor aircraft,” 1997.

[20] J. Wrinkers and K. Wise, “Reconfigurable flight control for a tailless advanced fighter aircraft,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA*, 1998.

[21] J. Brinker and K. Wise, “Flight Testing of Reconfigurable Control Law on the X-36 Tailless,” *AIAA Journal of Guidance, Control, and Dynamics,,* vol. 24, no. 5, pp. 903-909, 2001.

[22] A. Calise, “Development of a reconfigurable flight control law for the X-36 tailless fighter aircraft,” in *AIAA Guidance, Navigation, and Control Conference*, 2000.

[23] K. Wise, J. Brinker, A. Calise, D. Enns, M. Elgerma and P. Voulgaris, “Direct adaptive reconfigurable flight control for a tailles advanced figther aircraft,” *International journal of robust and nonlinear control,* vol. 9, pp. 999-1012, 1999.

[24] J. Boskovic, S.-M. L and R. Mehra, “Study of an adaptive reconfigurable control scheme for tailless advanced fighter aircraft (TAFA) in the presence of wing damage,” in *Position Location and Navigation Symposium*, 2000.

[25] J. Urnes, J. Davison and R. Jacobson, “A Damage Adaptive Flight Control System using Neural Network Technology,” in *Proceedings of the American Control Conference,*, 2001.

[26] A. Page and M. Steinberg, “Effects of control allocation algorithms on a nonlinear adaptive,” in *AIAA Guidance, Navigation, and Control Conference*, 1999.

[27] J. Anderson, C. Clark, P. Madsen and F. Unfried, “Reconfigurable Flight Control System Simulation,” *Aerospace Simulation,* pp. 89-110, 1988.

[28] 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,* vol. 27, pp. 755-770, 2013.

[29] E. V. KAMPEN, Q. P. CHU and J. A. MULDER, “ONLINE ADAPTIVE CRITIC FLIGHT CONTROL USING APPROXIMATED PLANT DYNAMICS,” in *IEEE* , 2006.

[30] L. Sonneveldt, E. van OOrt, Q. Chu and J. Mulder, “Nonlinear Adaptive Trajectory Control Applied to an F-16 Model,” *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS,* vol. 32, no. 2, 2009.

[31] G. J. Ducard, Fault-tolerant Flight Control and Guidance Systems, Springer, 2009.

[32] J. R. Fisher, AIRCRAFT CONTROL USING NONLINEAR DYNAMIC INVERSION IN CONJUNCTION WITH ADAPTIVE ROBUST CONTROL, 2004.

[33] B. White, L. Bruyere and A. Tsourdos, “Dynamic inversion for missile lateral velocity control via polynomial eigenstructure assignment,” in *Guidance, navigation and control Conference and Exhibit*, Texas, 2003.

[34] G. Chowdhary, E. C. Johnson, R. Chandramohan, M. Kimbrell and J. Calise, “Autonomous guidance and control of Airplanes under Actuator Failures and

Severe structural Damage," *American Institute of Aeronautics and Astronautics,* 2010.

[35] S. Sieberling, Q.-P.-. Chu and J. Mulder, "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction," *Journal of Guidance Control and Dynamics,* vol. 33, no. 6, pp. 1732-1742, 2010.

[36] M. Bodson and S. A. Frost, "Load Balacing in Control allocation," *Journal of sguidance Control and dynamics,* vol. 34, no. 2, p. 380, 2011.

[37] J. M. BUFFINGTON, "MODULAR CONTROL LAW DESIGN FOR THE INNOVATIVE CONTROL EFFECTORS (ICE) TAILLESS FIGHTER AIRCRAFT CONFIGURATION 101-3," 1999.

[38] F. Salama, "Fault tolerant systems through reconfigurable control allocation," 2014.

[39] O. Härkeg˚ard and s. T. Glad, "Resolving actuator redundancy- Optimal control vs. control allocation," *Automatica ELSEVIER,* vol. 41, pp. 137-144, 2004.

[40] O. Härkegard, "Dynamic control allocation using constrained quadratic programming," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey California, 2002.

[41] A. Various, The control handbook: Control systems advanced methods, CRC press, 2011.

[42] H. Ning, X. Jing and L. Cheng, "Online Identification of nonlinear spatiotemporal systems using learning approach," *IEEE Transactions on neural networks,* vol. 22, no. 9, p. 1381, 2011.

[43] W. Xie, Y. Zhu, Z. Zhao and Y. Wong, "Nonlinear system identification using optimazed dynamic neural networks," *Neurocomputing ELSEVIER,* vol. 72, no. 72, p. 3277, 2009.

[44] Y. Xiong and M. Saif, "Unknown disturbance inputs estimation based on a state functional observer design," *Automatica ELSEVIER,* vol. 39, no. 39, p. 1389, 2003.

[45] F. Bateman, H. Noura and M. Ouladsine, “Fault diagnosis and fault.tolerant control strategy for the aerosonde UAV,” *IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS,* vol. 47, no. 3, p. 2119, 2011.

[46] A. J. Calise and R. T. Rysdyk, “Nonlinear Adaptive Flight Control using Neural Networks. Georgia Institute of Technology,” Georgia Institute of Technology, Georgia Institute of Technology.

[47] M. B. McFarland and A. J. Calise, “Adaptive Nonlinear Control of Agile Antiair Missiles Using Neural Networks,” *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 8, no. 5, p. 749, 2000.

[48] M. Salem, M. Shahi Ashtiani and S. Sadati, “Designing an Adaptive Neural Network Based Controller for Automatic Landing Maneuver,” *International Journal of Scientific & Engineering Research,* vol. 4, no. 9, p. 696, 2013.

[49] M. Bahita and K. Belarbi, “Neural stable adaptive control for a class of nonlinear systems without use of a supervisory term in the control law,” *Journal of Engineering Science and Technology,* vol. 7, no. 1, pp. 97-118, 2012.

[50] L. Chen and K. S. Narendra, “Nonlinear adaptive control using neural networks and multiple models,” *Automatica,* vol. 37, p. 1245}1255, 2001.

[51] S. Ferrari and M. Jensenius, “Robust and Reconfigurable Flight Control by Neural Networks,” in *AIAA*, Virginia, 2005.

[52] M. R. Khosravani, “Application of Neural Network on Flight Control,” *International Journal of Machine Learning and Computing,* vol. 2, no. 6, 2012.

[53] P. J. Werbos, “Neural Networks for Flight Control: A Strategic and Scientific Assessment,” in *Neural Networks for Flight Control*, 2014.

[54] L. Yu, S. Fei, F. Long, M. Zhang and J. Yu, “Multilayer neural networks-based directad aptive control for switched nonlinear systems,” *Neurocomputing,* vol. 74, no. 74, pp. 481 - 486, 2010.

[55] D.-H. Shin and Y. Kim, “Reconfiguralble Flgiht Control System Desing using adaptive neural networks,” *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 12, no. 1, p. 87, 2004.

[56] Y. Diao and K. M. Passino, “Intelligent fault-tolerant control using adaptive and learning methods,” *Control Engineering Practice,* vol. 10, pp. 801 - 817, 2002.

[57] X. Zhang , T. Parisini and M. M. Polycarpou, “Adaptive Fault-Tolerant Control of Nonlinear Uncertain Systems: An Information-Based Diagnostic Approach”.

[58] M. Battipede, P. Gili, M. Napolitano, M. Perhinschi, L. Massotti and M. Lando, “Implementation of an Adaptive Predictor-Corrector Neural Cotnroller withing the NASA IFCS F-15 WVU Simulator,” in *Proceedings of the American Control Conference*, 2003.

[59] X.-L. Li, C. Jia, D.-X. Liu and D.-W. Ding, “Nonlinear adaptive control using multiple models and dynamics neural networks,” *Neurocpmputing ELSEVIER,* vol. 136, pp. 190-200, 2014.

[60] E. N. Johnson and G. Chowdhary, “Adaptive Neural Network Flight Control Using both Current and Recorded Data,” in *AIAA Guidance, Navigation and Control Conference and Exhibit*, South Carolina, 2007.

[61] J.-S. Wang and C. G. Lee, “Self-Adaptive recurrent neuro-Fuzzy controll of an autonomous underwater vehicle,” *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION,* vol. 19, no. 2, p. 283, 2003.

[62] B. Yao, Adaptive Robust Control of Nonlinear Systems with Application to Control of Mechanical Systems, 1996.

[63] D. Kufoalor and T. Johansen, “Reconfigurable Fault Tolerant Flight Control based on Nonlinear Model Predictive Control,” in *American Control Conference*, 2013.

[64] D. K. Kufoalor, Reconfigurable Autopilot Design using Nonlinear Model Predictive Control, 2012.

[65] V. Gavrilets, “Damage Tolerant Flight Control Systems for Unmanned Aircraft,” 2008.

[66] T. Wang, Sliding Mode Fault Tolerant Reconfigurable Control against Aircraft Control Surface Failures, Montreal, 2012.

[67] Y. Shtessel, J. Buffington and S. Banda, “Tailless Aircraft Flight Control Using Multiple Time Scale Reconfigurable Sliding Modes,” *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 10, no. 2, 2002.

[68] D. Ye and G.-H. Yang, “Adaptive Fault-tolerant tracking Control against actuators faults with appilcation to Flight Control,” *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY,* vol. 14, no. 6, 2006.

[69] R. S. Russell, “Non-linear F-16 Simulation using Simulink and Matlab,” *University if Minnesota,* 2003.

[70] L. Sonnoeveldt, “Nonlinear F-16 Model Description,” *Control and Simulation Division at Faculty of Aerospace Engineering. Delft University,* 2006.

[71] E. A. Morelli, “Global nonlinear parametric modeling with application to F-16 aerodynamics,” NASA Langley Research Center, Hampton, Virginia.

[72] M. C. Fox and D. K. Forrest, “Supersonic Aerodynamic Characteristics of an advnaced F-16 derivative Aircraft Configuration,” NASA Tehcnical Report 3355, Virginia, 1993.

[73] S. Thomas, H. G. Kwatny and B.-C. Chang, “Nonlinear reconfiguration for asymetric failures in a six dregrees of freedom F-16,” in *American Control Conference*, Boston, 2004.

[74] M. Cook, Flight Dynamics Principles, 2nd edition, Butterworth-Heinemann, 2007.

[75] S. O. Madwik, “An efficient orientation filter for inertial and inertial/magnetic sensor arrays,” 2010.

[76] O. Nguyen and K. Gilbert, “Simulator study f stall/Post-stall characteristics of a fighter airplane with relaxed longitudinal static stability,” NASA technical report, 1979.

[77] B. Stevens and F. Lewis, Aircraft control and simulation, New York: Wiley Inter-science, 1992.

[78] a. Several, calculo de aviones, ETSIA, Madrid.

[79] T. A. Johansen and T. I. Fossen, “Control Allocation - A survey,” *Autmatica ELSEVIER,* vol. 49, no. 49, p. 1087 1103, 2013.

[80] O. Härkegard, “Backsteping and Cotnrol allocation qith Applciation to flight control,” in *Backsteping and Cotnrol allocation qith Applciation to flight control*, Linköping Sweden, 2003.

[81] N. E. Wu, Y. Zhang and K. Zhou, “Detection, estimation and accomodation of loss of control effectiveness,” *INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING,* vol. 14, no. 14, pp. 775-795, 2000.

[82] M. J. Sanchez Garcia, “Active Aircraft Fault Detection, accomodation and Evaluation of Performance Reduction,” Cranfield University, 2014.

[83] J. Brinker and K. Wise, “Reconfigurable flight control for a tailless advanced fighter aircraft,” in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 1998.

[84] N. Andrew, “Machine Learning course notes. Coursera. Standford Unevirsity,” Coursera. Standford Unevirsity, 2013.

[85] H.-S. Shin, Autonomous Systems Control Notes.Cranfield, 2014.

[86] E. S. D. Unit, “67003 The equations of motion of a rigid aircraft,” 2003.

APPENDICES

Appendix A Models

A.1 Appendix Mass and geometric data

The following table shows the data of the baseline aircraft:

Parameter	Symbol	Value
$Mass\ of\ aircraft\ (kg)$	m	9295
$Reference\ wing\ surface\ (m^2)$	S	27.89
$Reference\ fin\ surface\ (m^2)$	S_{fin}	6.578
$Reference\ wing\ span\ (m)$	b	9.144
$Mean\ Aerodynamic\ chord\ (m)$	$\bar{c}\ or\ MAC$	3.45
$Root\ chord\ (m)$	c_{root}	5.04
$Wing\ taper\ ratio$	λ	0.21
$CG\ location\ (m)$	x_{cg}	$0.3 \cdot \bar{c}$
$Reference\ CG\ location\ (m)$	x_{cg_r}	$0.35 \cdot \bar{c}$
$Engine\ angular\ momentum\ (kg \cdot \frac{m^2}{s})$	h_E	216.9
$X-axis\ moment\ of\ inertia\ (kg \cdot m^2)$	I_x	12874.8
$Y-axis\ moment\ of\ inertia\ (kg \cdot m^2)$	I_y	75673.6
$Z-axis\ moment\ of\ inertia\ (kg \cdot m^2)$	I_z	85552.1
$XZ\ product\ of\ inertia\ (kg \cdot m^2)$	I_{xz}	1331.4
$XY\ product\ of\ inertia\ (kg \cdot m^2)$	I_{xy}	0
$YZ\ product\ of\ inertia\ (kg \cdot m^2)$	I_{yz}	0
$Distance\ to\ the\ elevator\ (m)$	l_e	1.69
$Distance\ to\ the\ aileron\ (m)$	l_a	3.82
$Distance\ to\ the\ LEF\ (m)$	l_{LEF}	2.54

Table 10-1 Mass and geometric properties baseline aircraft

A.2 Appendix Engine model

$$P_c(\delta_{th}) = \begin{cases} 64.94 \cdot \delta_{th} & if\ \ \delta_{th} \leq 0.77 \\ 2017.38 \cdot \delta_{th} - 117.38 & if\ \ \delta_{th} > 0.77 \end{cases}$$

$$\dot{P}_a = \frac{1}{\tau_{eng}} \cdot (P_c - P_a)$$

$$P_c = \begin{cases} P_c \;\; if \;\; P_c \geq 50 \;\; and \;\; P_a \geq 50 \\ 60 \;\; if \;\; P_c \geq 50 \;\; and \; P_a < 50 \\ 40 \;\; if \;\; P_c < 50 \;\; and \; P_a \geq 50 \\ P_c \;\; if \; P_c < 50 \;\; and \;\; P_a < 50 \end{cases}$$

$$\frac{1}{\tau_{eng}} = \begin{cases} 5 \;\; if \;\; P_c \geq 50 \;\; and \;\; P_a \geq 50 \\ \frac{1}{\tau_{eng}}^{*} \;\; if \;\; P_c \geq 50 \;\; and \; P_a < 50 \\ 5 \;\; if \;\; P_c < 50 \;\; and \; P_a \geq 50 \\ \frac{1}{\tau_{eng}}^{*} \;\; if \; P_c < 50 \;\; and \;\; P_a < 50 \end{cases}$$

$$\frac{1}{\tau_{eng}}^{*} = \begin{cases} 1 \;\; if \;\; (P_c - P_a) \leq 25 \\ 0.1 \;\; if \;\; (P_c - P_a) > 50 \\ 0.19 - 0.036 \cdot (P_c - P_a) \;\; if \;\; 25 \leq (P_c - P_a) < 50 \end{cases}$$

And the Thrust is given:

$$F_T = \begin{cases} T_{idle} + (T_{mil} - T_{idle}) \cdot \left(\frac{P_a}{50}\right) \;\; if \; P_a < 50 \\ T_{mil} + (T_{max} - T_{mil}) \cdot \left(\frac{P_a}{50} - 1\right) \;\; if \; P_a \geq 50 \end{cases}$$

A.3 Complete aerodynamic model

Here, the complete aerodynamic model is presented. Take the notation of section 3.1.7.1.

Total force coefficient in x-axis:

$$C_{X_T} = C_X(\alpha, \beta, \delta_e) + \Delta C_{X_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}} \cdot \left(C_{X_q}(\alpha) + \Delta C_{X_q}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right)$$

$$\Delta C_{X_{LEF}} = C_{X_{LEF}}(\alpha, \beta) - C_X(\alpha, \beta, \delta_e = 0)$$

Total force coefficient in y-axis:

$$C_{Y_T} = C_Y(\alpha, \beta) + \Delta C_{Y_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \left(\Delta C_{Y_{\delta_a}} + \Delta C_{Y_{\delta_{a_{LEF}}}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) \cdot \left(\frac{\delta_a}{20}\right) + \Delta C_{Y_{\delta_r}} \cdot \frac{\delta_r}{30}$$
$$+ \frac{r \cdot b}{2 \cdot V_{TAS}} \cdot \left(C_{Y_r}(\alpha) + \Delta C_{Y_{r_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) \quad + \quad \frac{p \cdot b}{2 \cdot V_{TAS}}$$
$$\cdot \left(C_{Y_p}(\alpha) + \Delta C_{Y_{p_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right)$$

$$\Delta C_{Y_{LEF}} = C_{Y_{LEF}}(\alpha,\beta) - C_Y(\alpha,\beta)$$

$$\Delta C_{Y_{\delta_a}} = C_{Y_{\delta_a}}(\alpha,\beta) - C_Y(\alpha,\beta)$$

$$\Delta C_{Y_{\delta a_{LEF}}} = C_{Y_{\delta a_{LEF}}}(\alpha,\beta) - C_{Y_{LEF}}(\alpha,\beta) - \Delta C_{Y_{\delta_a}}$$

$$\Delta C_{Y_{\delta_r}} = C_{Y_{\delta_r}}(\alpha,\beta) - C_Y(\alpha,\beta)$$

Total force coefficient in z-axis:

$$C_{Z_T} = C_Z(\alpha,\beta,\delta_e) + \Delta C_{Z_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}} \cdot \left(C_{Z_q}(\alpha) + \Delta C_{Z_{q_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right)$$

$$\Delta C_{Z_{LEF}} = C_{Z_{LEF}}(\alpha,\beta) - C_Z(\alpha,\beta,\delta_e = 0)$$

The total rolling moment coefficient:

$$C_{l_T} = C_l(\alpha,\beta,\delta_e) + \Delta C_{l_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \left(\Delta C_{l_{\delta_a}} + \Delta C_{l_{\delta a_{LEF}}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) \cdot \left(\frac{\delta_a}{20}\right) + \Delta C_{l_{\delta_r}} \cdot \frac{\delta_r}{30}$$
$$+ \frac{r \cdot b}{2 \cdot V_{TAS}} \cdot \left(C_{l_r}(\alpha) + \Delta C_{l_{r_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) + \frac{p \cdot b}{2 \cdot V_{TAS}}$$
$$\cdot \left(C_{l_p}(\alpha) + \Delta C_{l_{p_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) + \Delta C_{l_\beta}(\alpha) \cdot \beta$$

$$\Delta C_{l_{LEF}} = C_{l_{LEF}}(\alpha,\beta) - C_l(\alpha,\beta,\delta_e = 0)$$

$$\Delta C_{l_{\delta_a}} = C_{l_{\delta_a}}(\alpha,\beta) - C_l(\alpha,\beta,\delta_e = 0)$$

$$\Delta C_{l_{\delta a_{LEF}}} = C_{l_{\delta a_{LEF}}}(\alpha,\beta) - C_{l_{LEF}}(\alpha,\beta) - \Delta C_{l_{\delta_a}}$$

$$\Delta C_{l_{\delta_r}} = C_{l_{\delta_r}}(\alpha,\beta) - C_l(\alpha,\beta,\delta_e = 0)$$

The total pitching moment coefficient:

$$C_{m_T} = C_m(\alpha,\beta,\delta_e) + C_{Z_T} \cdot (x_{CG_r} - x_{CG}) + \Delta C_{m_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}}$$
$$\cdot \left(C_{m_q}(\alpha) + \Delta C_{m_{q_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) + \Delta C_m(\alpha) + \Delta C_{m_{ds}}(\alpha,\delta_e)$$

$$\Delta C_{m_{LEF}} = C_{m_{LEF}}(\alpha,\beta) - C_m(\alpha,\beta,\delta_e = 0)$$

The total yawing moment coefficient:

$$C_{n_T} = C_n(\alpha, \beta, \delta_e) + \Delta C_{n_{LEF}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right) + \left(\Delta C_{n_{\delta_a}} + \Delta C_{n_{\delta_{a_{LEF}}}} \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) \cdot \left(\frac{\delta_a}{20}\right) + \Delta C_{n_{\delta_r}} \cdot \frac{\delta_r}{30}$$
$$+ \frac{r \cdot b}{2 \cdot V_{TAS}} \cdot \left(C_{n_r}(\alpha) + \Delta C_{n_{r_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) + \frac{p \cdot b}{2 \cdot V_{TAS}}$$
$$\cdot \left(C_{n_p}(\alpha) + \Delta C_{n_{p_{LEF}}}(\alpha) \cdot \left(1 - \frac{\delta_{LEF}}{25}\right)\right) + \Delta C_{n_\beta}(\alpha) \cdot \beta - C_{Y_T} \cdot \left(x_{cg_r} - x_{cg}\right) \cdot \frac{\bar{c}}{b}$$

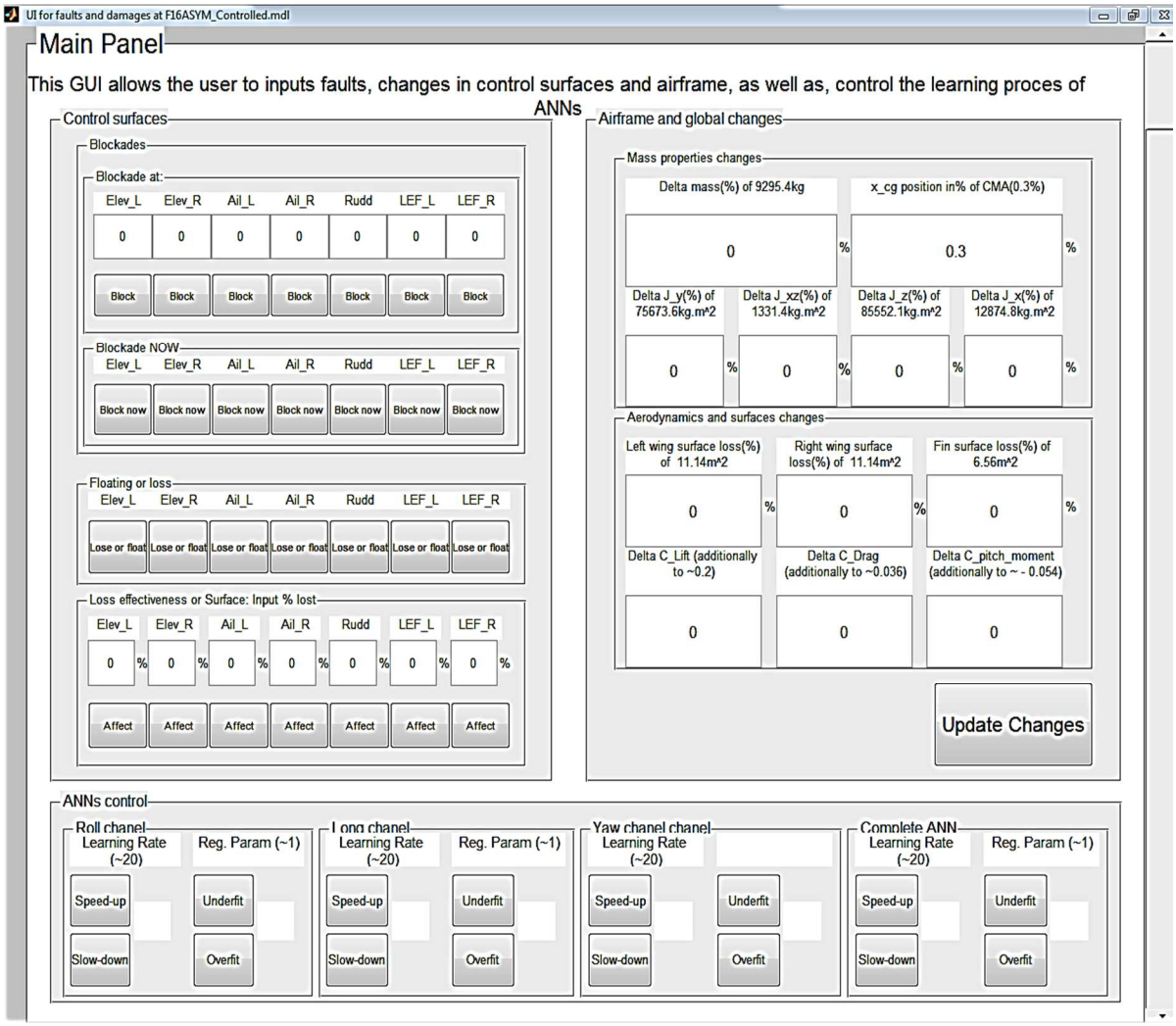
$$\Delta C_{n_{LEF}} = C_{n_{LEF}}(\alpha, \beta) - C_l(\alpha, \beta, \delta_e = 0)$$

$$\Delta C_{n_{\delta_a}} = C_{n_{\delta_a}}(\alpha, \beta) - C_n(\alpha, \beta, \delta_e = 0)$$

$$\Delta C_{n_{\delta_{a_{LEF}}}} = C_{n_{\delta_{a_{LEF}}}}(\alpha, \beta) - C_{n_{LEF}}(\alpha, \beta) - \Delta C_{n_{\delta_a}}$$

$$\Delta C_{n_{\delta_r}} = C_{n_{\delta_r}}(\alpha, \beta) - C_n(\alpha, \beta, \delta_e = 0)$$

A.4 Main panel GUI



Appendix B Performance analysis and validation

B.1 SPPO spec with Gain scheduling

Set-point	ζ	$\omega_n \left(rad/_s\right)$	$^n/_\alpha$ $(g's/rad)$
1	0.77	3.45	10
2	0.9	3.95	24
3	0.98	4.2	61
4	0.55	2.6	100
5	0.82	3.6	40

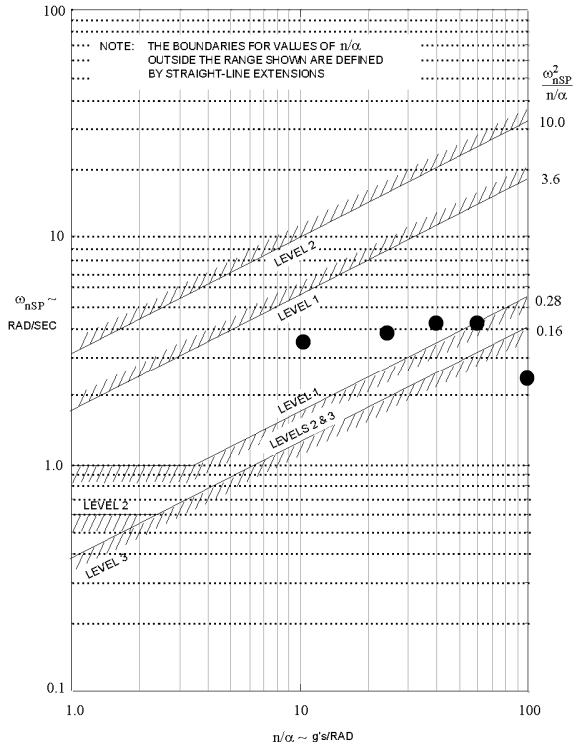


Figure 10-1 SPPO frequency requirement Cat A

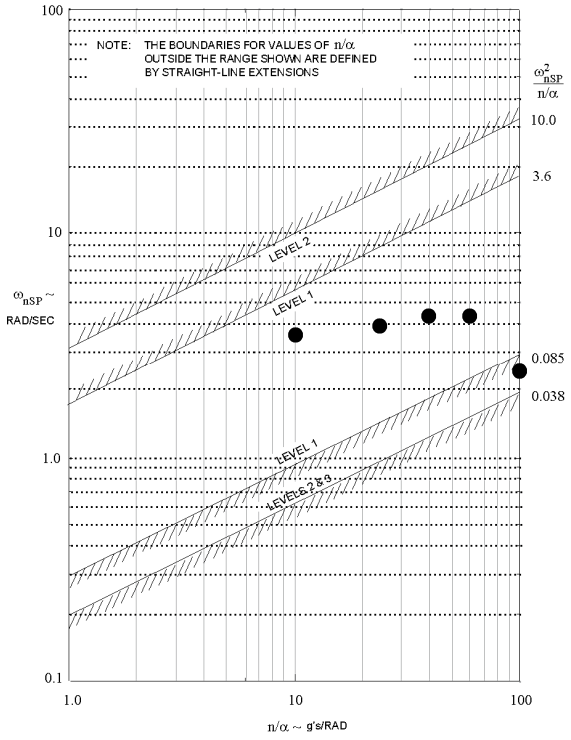
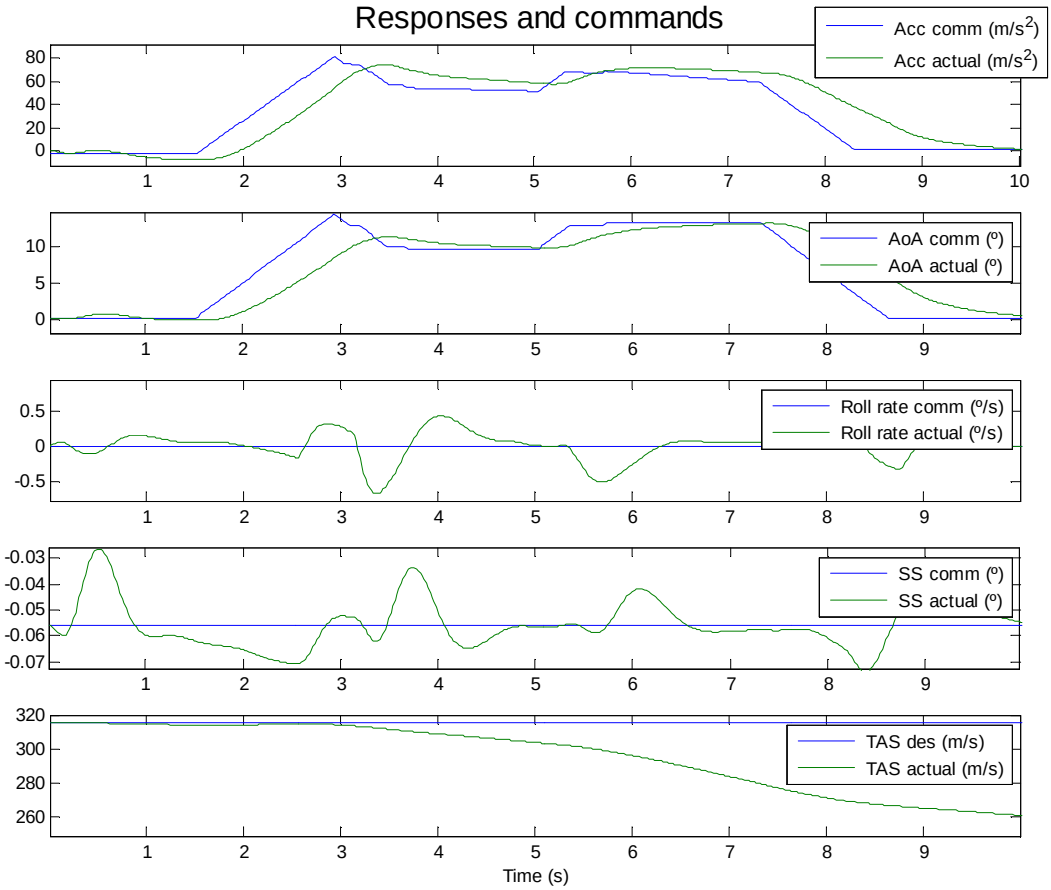


Figure 10-2 SPPO frequency requirement Cat B

B.2 SPPO specifications set-point 4 design

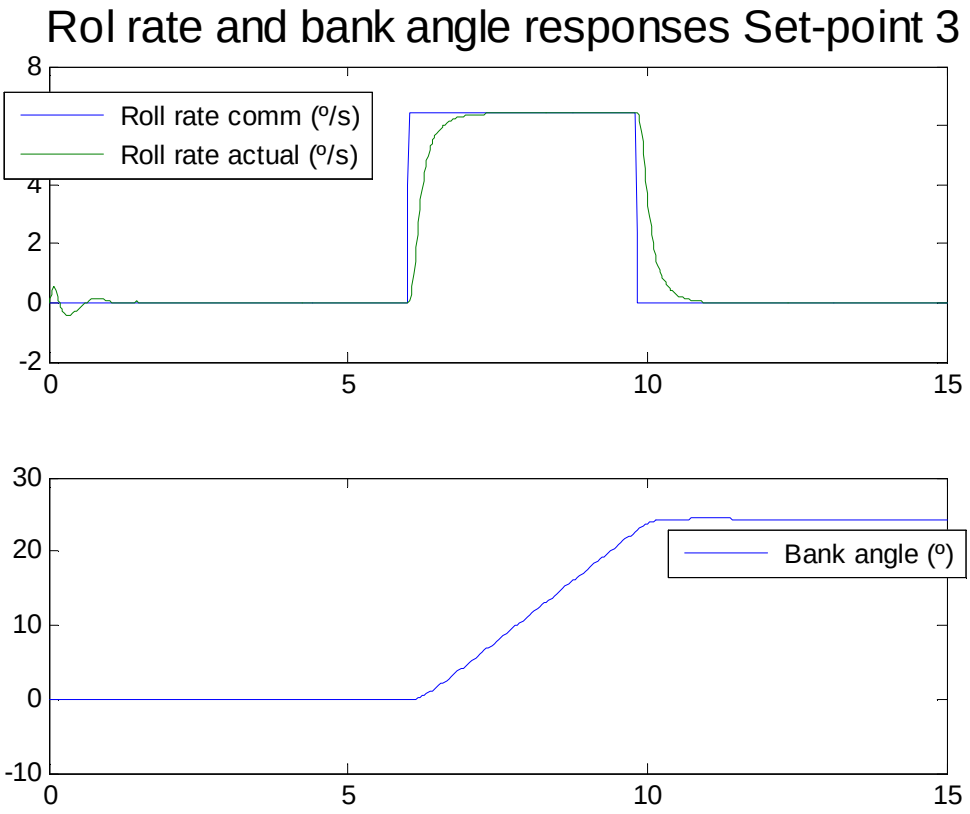
Set-point	ζ	$\omega_n \left(rad/s\right)$	$n/\alpha \ (g's/rad)$
4	0.44	6.84	100
AoA		***q with AoA***	
$\tau_{ref} = 0.35$		$\tau_{ref} = 0.06$	
$K = 10$		$K = 1$	

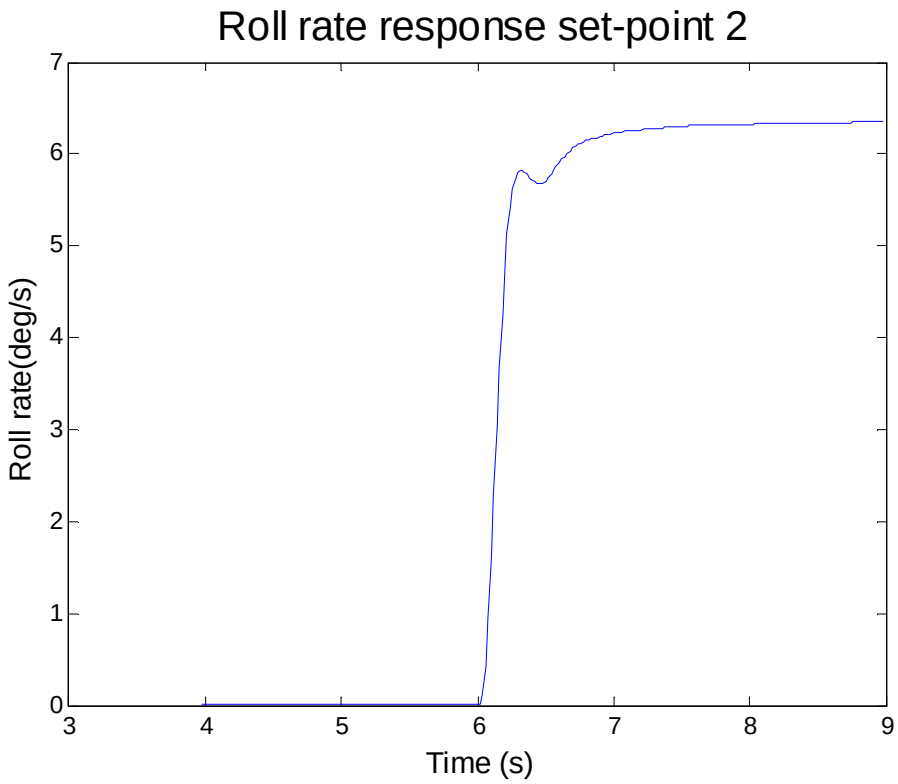
B.3 Long control maximum load factor of set-point 5

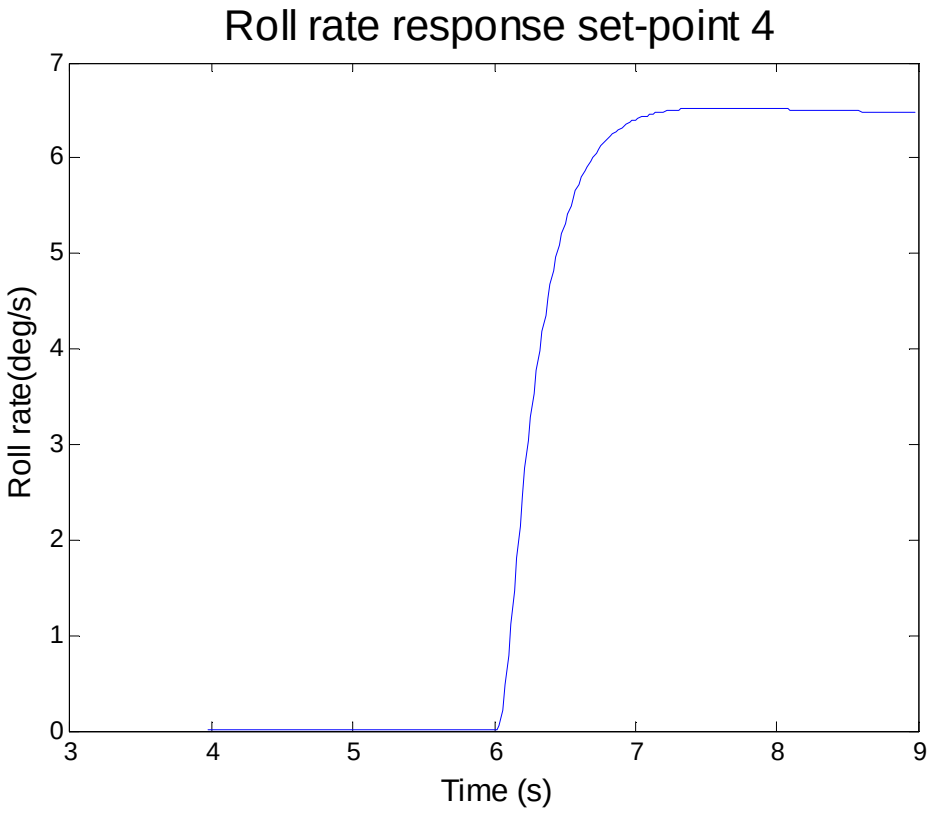


B.4 Roll rate response set-point 2

Note that this response is the response to a step input starting at second 6 with the same steady value as the response.

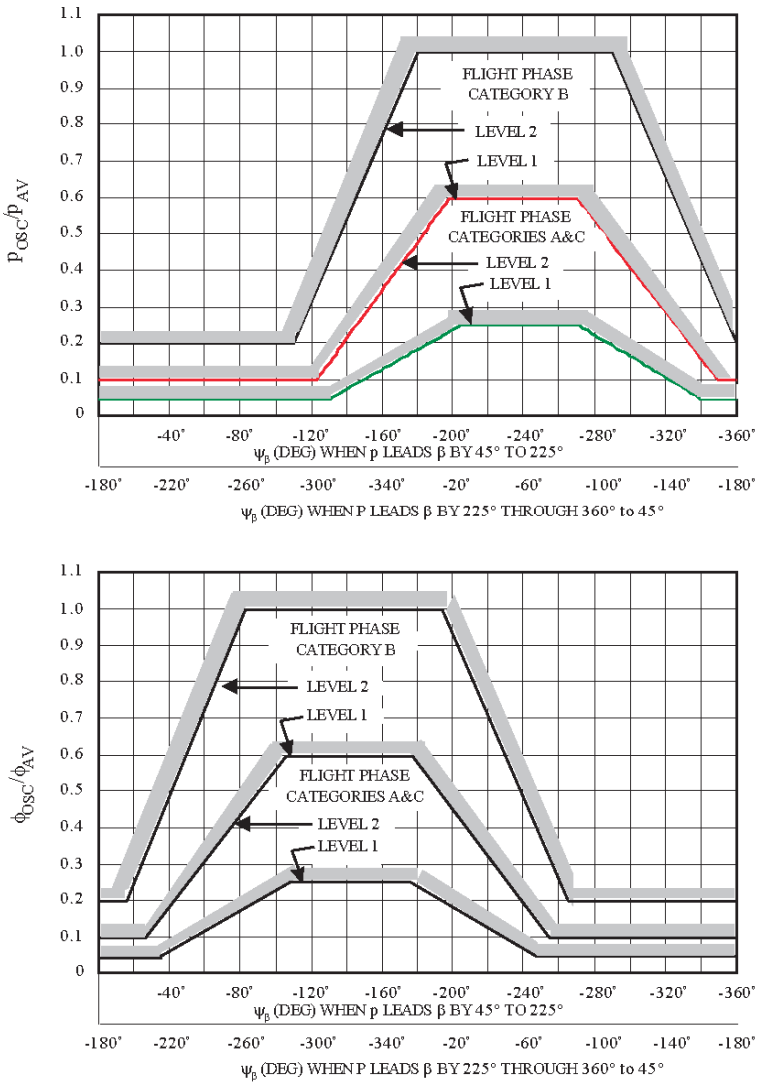



Roll rate response set-point 2
Roll rate(deg/s)
Time (s)
0
1
2
3
4
5
6
7
3
4
5
6
7
8
9


Roll rate response set-point 4
Roll rate(deg/s)
Time (s)
0
1
2
3
4
5
6
7
3
4
5
6
7
8
9

B.5 Roll and bank angle oscillations specs

MIL-F-8785C



B.6 Bank angle times roll performance Specs

3.3.4.1 Roll performance for Class IV airplanes. Roll performance in terms of ϕ_t for Class IV airplanes is specified in table IXb. Additional or alternate roll performance requirements are specified in 3.3.4.1.1 and 3.3.4.1.2; these requirements take precedence over table IXb. Roll performance for Class IV airplanes is specified over the following ranges of airspeeds:

Speed Range Symbol	Equivalent Airspeed Range For Level 1	For Levels 2 & 3
VL	$V_{o_{min}} \le V \le V_{min} + 20$ KTS	$V_{min} \le V \le V_{min} + 20$ KTS
L	$V_{min} + 20$ KTS$^{(1)} \le V < 1.4\ V_{min}$	$V_{min} + 20$ KTS $\le V < 1.4\ V_{min}$
M	$1.4\ V_{o_{min}} \le V < .7\ V_{max}^{(2)}$	$1.4\ V_{min} \le V < .7\ V_{max}$
H	$.7\ V_{max}^{(2)} \le V \le V_{o_{max}}$	$.7\ V_{max} \le V \le V_{max}$

(1) or $V_{o_{min}}$ whichever is greater

(2) or $V_{o_{max}}$ whichever is less

TABLE IXb. Roll performance for Class IV airplanes.

Time to Achieve the Following Bank Angle Change (Seconds)

Level	Speed Range	Category A 30˚	Category A 50˚	Category A 90˚	Category B 90˚	Category C 30˚
1	VL	1.1			2.0	1.1
	L	1.1			1.7	1.1
	M			1.3	1.7	1.1
	H		1.1		1.7	1.1
2	VL	1.6			2.8	1.3
	L	1.5			2.5	1.3
	M			1.7	2.5	1.3
	H		1.3		2.5	1.3
3	VL	2.6			3.7	2.0
	L	2.0			3.4	2.0
	M			2.6	3.4	2.0
	H		2.6		3.4	2.0

3.3.4.1.1 Roll performance in Flight Phase CO. Roll performance for Class IV airplanes in Flight Phase CO is specified in table IXc in terms of ϕ_t for 360˚ rolls initiated at 1g, and in table IXd for rolls initiated at load factors between $.8n_o$(-) and $.8n_o$(+).

B.7 Bank angle times roll performance Specs Cat A CO

TABLE IXc. Flight Phase CO roll performance in 360° rolls.

Time to Achieve the Following Bank Angle Change (Seconds)

Level	Speed Range	30°	90°	180°	360°
1	VL	1.0			
	L		1.4	2.3	4.1
	M		1.0	1.6	2.8
	H		1.4	2.3	4.1
2	VL	1.6			
	L	1.3			
	M		1.3	2.0	3.4
	H		1.7	2.6	4.4
3	VL	2.5			
	L	2.0			
	M		1.7	3.0	
	H		2.1		