INSTITUTO TECNOLÓGICO DE AERONÁUTICA

Gustavo Oliveira Violato

Review of Nonlinear Techniques Applied to Flight Control

Trabalho de Graduação
Ano 2009

Aeronáutica

CDU 629.73.017

Gustavo Oliveira Violato

Review of Nonlinear Techniques Applied to Flight Control

Orientador
Prof. Dr. Ing. Pedro Paglione

Co-Orientador
Prof. Dr. Félix Mora-Camino

Divisão de Engenharia Aeronáutica - IEA

São José dos Campos

Comando-Geral de Tecnologia Aeroespacial

Instituto Tecnológico de Aeronáutica

2009

Cataloging-in Publication Data

Division of Central Library of ITA/CTA

Oliveira Violato, Gustavo

Review of Nonlinear Techniques Applied to Aircraft Control / Gustavo Oliveira Violato.
São José dos Campos, 2009.
121f.

Work of Bachelor in Engineering – Course of Aeronautical Engineering. – Aeronautical Institute of Technology, 2009. Advisor: Prof. Dr. Ing. Pedro Paglione. Co-advisor: Prof. Dr. Félix Mora-Camino.

1. Nonlinear control. 2. Flight Mechanics. 3. Aircraft Control. 4. Adaptive Control. I. Aerospace Technical Center. Aeronautics Institute of Technology. Division of Aeronautical Engineering - Flight Mechanics. II. Title.

BIBLIOGRAPHIC REFERENCE

OLIVEIRA VIOLATO, Gustavo. **Review of Nonlinear Techniques Applied to Aircraft Control**. 2009. 121f. Work of Bachelor in Engineering – Technological Institute of Aeronautics, São José dos Campos.

CESSION OF RIGHTS

AUTHOR NAME: Gustavo Oliveira Violato
PUBLICATION TITLE: Review of Nonlinear Techniques Applied to Aircraft Control.
TYPE OF PUBLICATION/YEAR: Bachelor Work / 2009

Gustavo Oliveira Violato
Rua Manoel Correia de Freitas, 1300
CEP 82530-070 – Curitiba–PR–Brasil

Review of Nonlinear Techniques Applied to Flight Control

Essa publicação foi aceita como Relatório Final de Trabalho de Graduação

Gustavo Oliveira Violato
Autor

Prof. Dr. Ing. Pedro Paglione (ITA-IEA)
Orientador

Prof. Dr. Félix Mora-Camino (ENAC-LARA)
Co-orientador

Prof. Dr. Flávio Luiz de Silva Bussamra (ITA-IEA)
Coordenador do Curso de Engenharia Aeronáutica

São José dos Campos, 24 de novembro de 2009

Ao meu pai, Denis, minha mãe, Luci, meu irmão, Marcelo e minhas irmãs Juliana e Adriana.

Acknowledgements

À Deus, ou à coincidência, dependendo das crenças, pela chance de estar vivo e tatear a existência.

Ao meu maior mestre, orientador e motivador, exemplo de vida e conduta que me servirá pela eternidade: meu pai.

À todos meus outros verdadeiros mestres, que tiraram do seu tempo para mostrarem-me caminhos nos meandros do conhecimento. Em especial aos meus orientadores neste trabalho, professores Félix Mora-Camino e Pedro Paglione.

À minha mãe e minha família, pela minha criação, minha formação pessoal e simplesmente por sempre estarem lá para mim.

Aos amigos do "Iguana", por serem os irmãos que eu pude escolher, por participarem dos momentos mais importantes de minha vida e por terem contribuído para forjar meu caráter.

Aos amigos do ITA, por enfrentarem juntos as batalhas dos anos de formação superior, em especial àqueles do Aerodesign, por enfrentarem também outras batalhas junto comigo, as quais me motivam a aprender cada vez mais.

À todos os bons amigos que fiz na França, durante a confecção deste trabalho e após, em especial à Antoine Drouin, Marcella Drummond e Reyes Santamaria, pela boa companhia, ensinamentos de vida e por confiarem em minha capacidade antes de mim mesmo.

"A dúvida é o princípio da sabedoria."
"A educação tem raízes amargas, mas os seus frutos são doces."
— Aristóteles

Resumo

Este trabalho trata do estudo e aplicação de técnicas de controle não-linear ao controle de aeronaves de asa fixa. As técnicas estudadas são a inversão dinâmica, o backstepping, técnicas não-lineares adaptativas e o controle por modos deslizantes.O obetivo foi apresentar uma revisão dos principais métodos de controle não-linear introduzidos na literatura especilazada nas últimas décadas. Os fundamentos de cada teoria são explorados e seguidos de aplicações demonstrativas ao controle do vôo. Simulações são realizadas para vários casos de vôo e o desempenho de cada controlador é analisado criticamente.

Abstract

This work covers the study and application of nonlinear control techniques to flight control of fixed wing aircrafts. The techniques covered are dynamic inversion, backstepping, nonlinear adaptive control and sliding mode control. The objective is to present a review of the methods introduced in the literature in the last decades. The fundamentals of each theory are explored followed by demonstrative applications to the problem of flight control. Simulations are performed for various flight cases and the controller performance is critically analysed.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Why Nonlinear Control?

During the past century, linear control theory has reached a level of understanding in theory and application high enough to be considered a mature and well developed science and technology. The last decades have seen the advent of robust linear controllers, which can deal with plant nonlinearities, at least to a certain degree. Hence, natural questions that come to mind when studying nonlinear techniques is what does the nonlinear controllers can offer to enhance performance, how much improvement can they cause and how "costly" they will be. Based mainly in (SLOTINE; LI, 1991), some reasons for the study of nonlinear control can be pointed. It is noticed that nonlinear controllers can handle a large range of operation, since their design validity is not based on the assumption that the dynamics will remain close to the linearization point. Nonlinear control methods can also deal with hard nonlinearities, such as saturation, dead-zones and hysteresis, to cite a few, which simply rend the mathematical model non-linearizable.

Perhaps the greatest advantage of nonlinear control methods to aeronautical applications is the ability to deal with model uncertainties. The aerodynamic coefficients values are hard to precisely estimate, and some may be sensible to flight conditions, not to cite for instance the possibility of aircraft damage in flight. Techniques such as adaptive control and backstepping design can deal with those problems, respectively by continuously estimating the parameters in flight or by using the physical characteristics of the nonlinearities when designing the controller.

Finally, from a more practical point of view, the development of electronic technology has made possible to use increasingly powerful low cost microprocessors in control applications, allowing for more calculations in the control loop. Nonlinear control can also deal with sensors of somewhat degradable quality, since these do not have to be linear in the first place. With some luck, this allows a cheaper design with enhanced performance.

1.2 Objectives

The objective of this work is to apply some of the existing nonlinear methodologies to the design of stabilization and guiding systems to aircraft control, focusing specially in the application for UAVs (Unmanned Aircraft Vehicles). A critical analysis following each application will take into account its advantages an disadvantages, trying to reach a better understanding of the methods. It will be always kept in mind the final objective of applying the discussed theory, addressing which would be the most practical way of quickly putting these techniques into use.

1.3 Report Structure

The next chapter briefly discusses the fundamental theory behind each technique further applied, noticing its assumptions and presenting the most important results in which the later development will rely on. The third chapter is a development of the flight dynamics equations, showing the hypothesis behind the used models and arriving at a mathematical description suited to the application of control techniques. Chapters 4 to 6 show an increasingly comprehensive application of the methodologies previously discussed to aircraft control and consist the main part of this work. The last chapter presents the concluding remarks.

2 Flight Dynamics

2.1 Modelling the Aircraft Dynamics

To *model* a dynamic system is to derive a set of differential equations that describe the evolution of this system in time. Of course, it is desired that the model under study represents reality in the best possible way and that is why the laws of physics are always the starting point.

The aeroplane case is no different. Hence, classical mechanics (Newton's laws of motion) can be used to accurately describe the aeroplane flight. Following (TENENBAUM, 2003) (a great reference in Dynamics, from which we also borrow the notation), we start from the equations of motion for the aircraft of the first and second type, respectively:

$$\mathbf{F}^{\mathcal{F}_\mathrm{e}/C^*} = \mathbf{F}^{\mathcal{F}_\mathrm{a}/C^*} + \mathbf{F}^{\mathcal{F}_\mathrm{p}/C^*} + \mathbf{F}^{\mathcal{F}_\mathrm{o}/C^*} = m\left({}^{A}\frac{\mathrm{d}}{\mathrm{d}t}{}^{R}\mathbf{v}^{C^*} +^{R} \omega^{A} \times^{R} \mathbf{v}^{C^*}\right) \tag{2.1a}$$

$$\mathbf{M}^{\mathcal{F}_\mathrm{e}/C^*} = \mathbf{M}^{\mathcal{F}_\mathrm{a}/C^*} + \mathbf{M}^{\mathcal{F}_\mathrm{p}/C^*} = \mathbb{I}^{A/C^*} \cdot^{R} \alpha^{A} +^{R} \omega^{A} \times \mathbb{I}^{A/C^*} \cdot^{R} \omega^{A} \tag{2.1b}$$

Where:

$\mathcal{F}_\mathrm{e}$ stands for *External* forces.

$\mathcal{F}_\mathrm{a}$ stands for *Aerodynamic* forces.

$\mathcal{F}_\mathrm{p}$ stands for *Propulsion* forces.

$\mathcal{F}_\mathrm{o}$ stands for *Other* forces.

Here, the *other forces* term is simply the weight of the aircraft, that is why there is no external moment component related to it, as can be seen from 2.1b. Also, as noted from equations 2.1a and 2.1b, C^* refers to the centre of gravity of the aircraft. The great challenge of correctly modelling the aircraft dynamics is therefore to determine the forces and moments components generated by the aerodynamic and propulsion external forces. Also, it is worth noticing that it was considered that the aircraft's mass and inertia matrix are both kept constant during flight, which, although not true for regular aircrafts, are sufficiently slowly varying to render the above equation precise enough for all application means.

2.1.1 Referentials

The vectors in the equations of motion for the aircraft can be written in any base. As we will see just below, depending on the nature of the vector (i.e. aerodynamic resultant force or body angular velocity) it is more suitable to write it in one ore another base, which will always be associated with a given reference frame. This is why different vector bases are defined for the description of a plane in flight. The most used ones are defined below:

The inertial reference frame R: Considering the "flat earth" referential as a inertial frame (STEVENS; LEWIS, 2003), we can define the *inertial reference* base (see figure 2.1) $\mathbf{R} = \{\mathbf{x_0}, \mathbf{y_0}, \mathbf{z_0}\}$ with $\mathbf{x_0}$and$\mathbf{y_0}$ defining the horizontal plane, $\mathbf{x_0}$ pointing a given direction (i.e. local north), $\mathbf{z_0}$ pointing downwards and finally $\mathbf{y_0}$ in such a way that $\mathbf{R}$ is right-hand positive.

The body reference frame B: As the name suggests, the *body reference* base $\mathbf{B} = \{\mathbf{x_B}, \mathbf{y_B}, \mathbf{z_B}\}$ (see figure 2.1) is aligned with the aircraft geometry, being $\mathbf{x_B}$ the aircraft central axis, $\mathbf{y_B}$ pointing to the right wing and $\mathbf{z_B}$ rendering $\mathbf{B}$ a right-hand positive base.

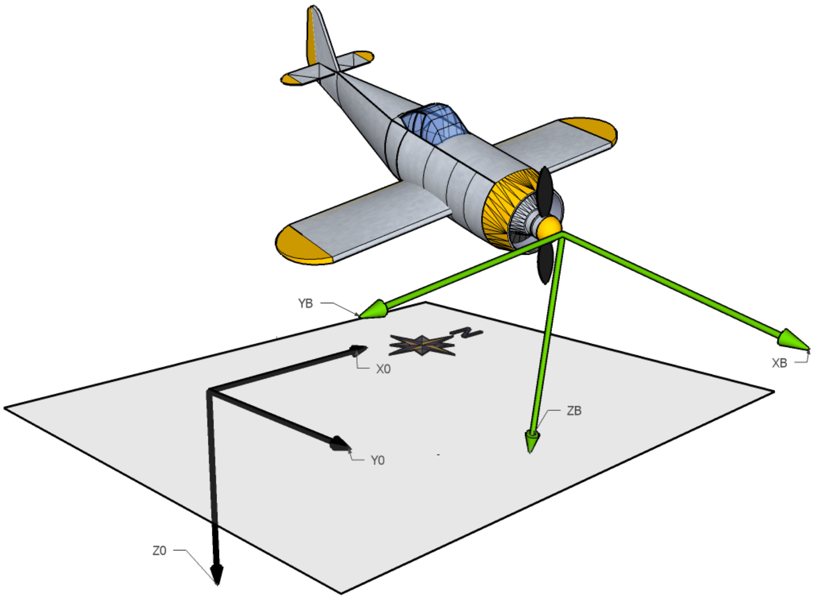


FIGURE 2.1 – Reference frames **R** and **B**

The wind reference frame W: This is the frame aligned with the true airspeed velocity vector (see figure 2.2). The *wind reference* base $\mathbf{W} = \{\mathbf{x_W}, \mathbf{y_W}, \mathbf{z_W}\}$ is defined with $\mathbf{x_W}$ in the same direction as the true airspeed, pointing away from the aircraft. The other axis are defined in such a way that, when rotated an angle $-\beta$ around the $\mathbf{z_W}$ axis, and then an angle α around the $\mathbf{y_W}$ axis, the resulting frame coincides with body frame **B**.

The propulsion reference frame P: Following the same thinking as for the wind reference frame, we define the *propulsion reference* frame $\mathbf{P} = \{\mathbf{x_P}, \mathbf{y_P}, \mathbf{z_P}\}$, as being aligned with the resultant propulsion force vector acting on the aircraft in a given moment. Hence, to transform **P** into **B**, the rotations are analogous to the described above for the frame **W**, but now the angles are α_f and $-\beta_f$

2.1.2 Aerodynamic Forces and Moments

For a mathematical model of the aerodynamic forces and moments we closely follow what is presented in (NELSON; SMITH, 1989; STEVENS; LEWIS, 2003). We start by

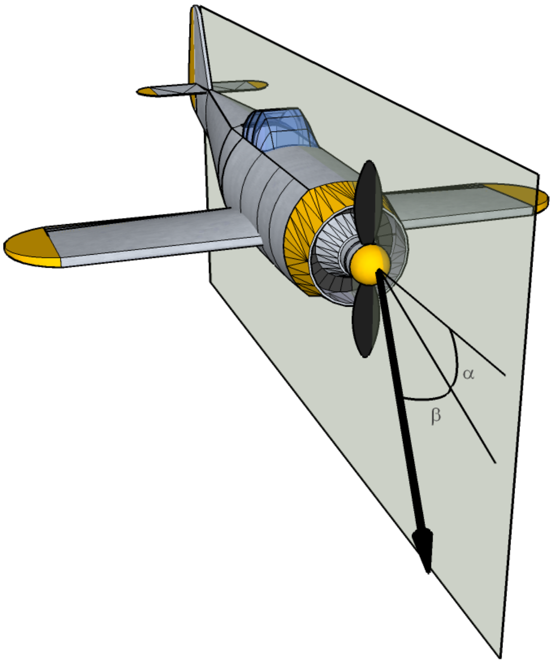


FIGURE 2.2 – Reference frame **W**

taking the centre of gravity of the aircraft as a reference point. The practical objective of the study of aircraft aerodynamics is to give the resultant force and moment acting in this reference point for all the expected flight conditions (SCHLICHTING; TRUCKENBRODT, 1979). Suppose then that we have this resultants. Lets decompose the aerodynamic resultant force in the directions given by the wind frame **W**. We will then call the set of resultant forces: $\mathcal{F}_{\mathrm{a}} = [-D, Y, -L]^T_{\mathbf{W}}$. In a similar manner, the total aerodynamic moment is decomposed in the body frame **B** forming the set of moments: $\mathcal{M}^{\mathcal{F}_{\mathrm{a}}} = [\mathcal{L}, \mathcal{M}, \mathcal{N}]^T_{\mathbf{B}}$.

Now, what the study of aerodynamics say (SCHLICHTING; TRUCKENBRODT, 1979) is that, under certain conditions, the aerodynamic forces and moments $(\mathcal{F}_{\mathrm{a}}, \mathcal{M}^{\mathcal{F}_{\mathrm{a}}})$ and the flight variables (i.e the angles α and β, the angular velocity components and the control surface deflections) hold a linear relation to each other. The conditions just cited above are that α and β remain relatively small and that the angular velocity rates also remain bounded to certain values (when this conditions are not satisfied, there must be an explicit warning). The only exception of this rules is the drag force (D), which approximately holds a parabolic relation with the lift force.

Also, all forces and moments are adimensionalized in the aerodynamic calculations, so that these relations described above are given in terms of the aerodynamic coefficients.[1] Again, the reader is asked to refer to (SCHLICHTING; TRUCKENBRODT, 1979) for a complete treatment of the subject. The important result to retain from the above discussion is that it is possible to write each of the aerodynamic coefficients as showed below:

$$C_L = C_{L0} + C_{L\alpha}\alpha + C_{Lq}q\frac{cma}{2V_{ref}} + C_{L\mathrm{dp}}\mathrm{d}p \tag{2.2a}$$

$$C_Y = C_{Y\beta}\beta + C_{Yr}r\frac{b}{2V_{ref}} + C_{Y\mathrm{dr}}\mathrm{d}r + C_{Y\mathrm{da}}\mathrm{d}a \tag{2.2b}$$

$$C_D = C_{D0} + k_1 C_L + kC_L^2 \tag{2.2c}$$

$$C_{\mathcal{L}} = C_{\mathcal{L}\beta}\beta + C_{\mathcal{L}p}p\frac{b}{2V_{ref}} + C_{\mathcal{L}r}r\frac{b}{2V_{ref}} + C_{\mathcal{L}\mathrm{da}}\mathrm{d}a + C_{\mathcal{L}\mathrm{dr}}\mathrm{d}r \tag{2.3a}$$

$$C_{\mathcal{M}} = C_{\mathcal{M}0} + C_{\mathcal{M}\alpha}\alpha + C_{\mathcal{M}\dot{\alpha}}\dot{\alpha}\frac{cma}{2V_{ref}} + C_{\mathcal{M}q}q\frac{cma}{2V_{ref}} + C_{\mathcal{M}\mathrm{dp}}\mathrm{d}p \tag{2.3b}$$

$$C_{\mathcal{N}} = C_{\mathcal{N}\beta}\beta + C_{\mathcal{N}p}p\frac{b}{2V_{ref}} + C_{\mathcal{N}r}r\frac{b}{2V_{ref}} + C_{\mathcal{N}\mathrm{da}}\mathrm{d}a + C_{\mathcal{N}\mathrm{dr}}\mathrm{d}r \tag{2.3c}$$

Notice that in (2.2a) to (2.3c) each coefficient depends on some specific state variables and not all of them.

2.1.3 Propulsion Forces and Moments

The total thrust force produced by the aircraft power system normally varies with flight regime. The two principal quantities on which this force depends are the airspeed velocity and air density. Depending on the type of engine used, there are different cor-

[1] That is, the aerodynamic force coefficients have to be multiplied by the force $\frac{1}{2}\rho V^2 S_{ref}$, where ρ is the air density, V is the total airspeed velocity and S_{ref} is a reference area to give the final forces. The moment coefficients have to be multiplied by $\frac{1}{2}\rho V^2 S_{ref} c_{ref}$ where c_{ref} is a reference length. The values of S_{ref} and c_{ref} for *each* coefficient **have** to be specified in the model description.

relations for these variations, but those can be approximately summarized in a general equation of the form:

$$F = n_{eng}\pi F_{max}\left(\frac{\rho}{\rho_i}\right)^{n_\rho}\left(\frac{V}{V_i}\right)^{n_V} \tag{2.4}$$

Where:

$\mathbf{n}_{eng}$ is the number of engines

π is the throttle $(0 < \pi < 1)$

$\mathbf{F}_{max}$ is each engine's maximum force

ρ is the current air density

$\mathbf{V}$ is the current airspeed

ρ_i is a reference air density

$\mathbf{V}_i$ is a reference airspeed

$\mathbf{n}_\rho$ is a correlation exponent for density

$\mathbf{n}_V$ is a correlation exponent for airspeed

Once a suitable set of measurements (or a model) are done with a engine, (2.4) can be used to fit the data and give the thrust force for any desired flight regime. In the case of a fixed pitch propeller and a electric engine, as for the simple UAV described in appendix A for example, the values of n_ρ and n_V are respectively 0.75 and -1.

The propulsion forces will only produce moment if the axis of action of the thrust force does not pass through the aircraft's center of gravity. In that case, naturally the moment produced will equal the cross product between the position vector of the engine with respect to C* ($\mathbf{p}^{eng/C^*}$) and the thrust vector:

$$\mathcal{M}^{\mathcal{F}_\mathrm{p}} = \mathbf{p}^{eng/C^*} \times \mathbf{F} \tag{2.5}$$

2.2 Equations of Motion for an Aircraft

The set of parameters necessary to describe the forces acting in the aircraft for each flight condition is called an *aircraft model*. Once we have this model (as in appendix A) we can right the equations of motion for the aircraft, that is, replace $\mathbf{F}^{\mathcal{F}_\mathrm{a}/C^*}, \mathbf{F}^{\mathcal{F}_\mathrm{p}/C^*}, \mathbf{F}^{\mathcal{F}_\mathrm{o}/C^*}$

and $\mathbf{M}^{\mathcal{F}_\mathrm{a}/C^*}, \mathbf{M}^{\mathcal{F}_\mathrm{p}/C^*}$ in (2.1a) and (2.1b) for the expressions derived in the precedent sections. Before doing this, we note that each force studied above is written in its own referential, so it is necessary to chose one base in which to "merge" all the vector values. For that reason, it is useful to discuss a little the different ways in which this coordinates transformations can be done.

2.2.1 Attitude representations and Transformation Matrices

Our objective here is only to present two different forms of performing a coordinate transformation, the Euler angles and the quaternions parameters. The reader is refered to (PHILLIPS; HAILEY; GEBERT, 2001; ROBINSON, 1958) for more complete discussions.

Euler Angles Due to their physical sense, the Euler Angles are the most used set of parameters to perform coordinate transformations. If we have two frames completely independent of each other, as in figure 2.1, we can describe a transformation as three consecutive rotations, each about a different axis.

Following figure 2.3, we first have to coincident frames **0** and **B0** 2.3(a). Rotating **B0** around the $\mathbf{z_{B0}}$ axis an angle ψ, we get the intermediate frame **B1** 2.3(b). Considering only the plane defined by $[\mathbf{x_{B0}}, \mathbf{y_{B0}}]$ we have the following equation relating the coordinates of a vector in **B1** with its coordinates in **B0**:

$$\begin{bmatrix} x_{B1} \\ y_{B1} \\ z_{B1} \end{bmatrix} = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x_{B0} \\ y_{B0} \\ z_{B0} \end{bmatrix} \equiv \mathbf{T}_\psi \mathbf{p}_{B0}$$

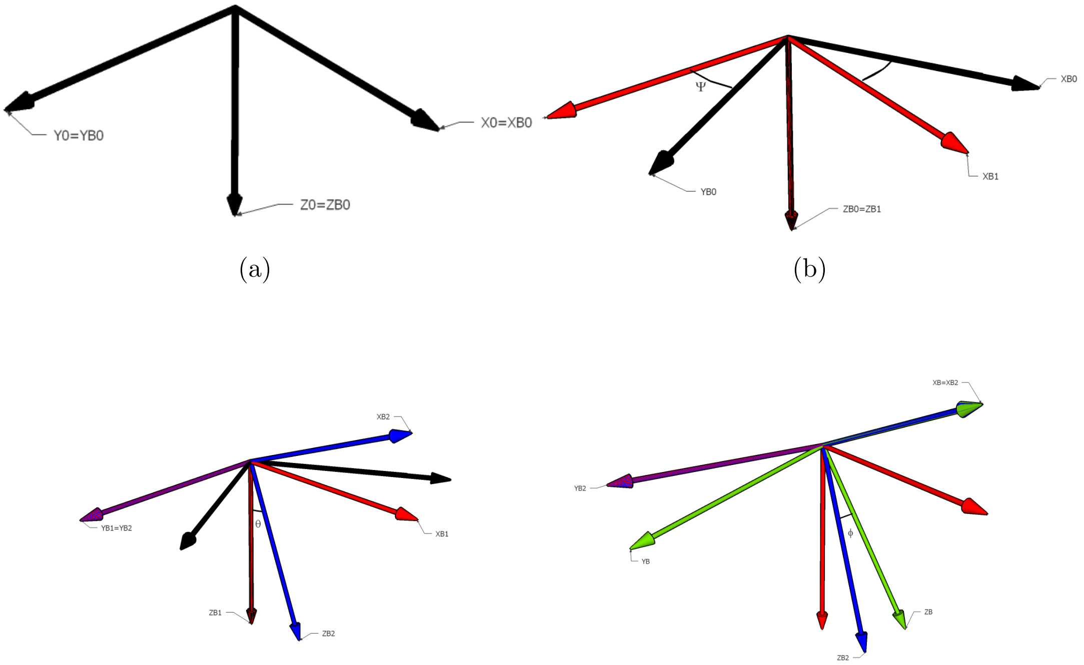
Next, rotating **B1** around $\mathbf{y_{B1}}$ an angle θ, we arrive at the second intermediate frame **B2**

2.3(c). Again, considering the plane $[\mathbf{x_{B1}}, \mathbf{z_{B1}}]$ we write:

$$\begin{bmatrix} x_{B2} \\ y_{B2} \\ z_{B2} \end{bmatrix} = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{bmatrix} x_{B1} \\ y_{B1} \\ z_{B1} \end{bmatrix} \equiv \mathbf{T}_\theta \mathbf{p}_{B1}$$

Finally, rotating **B2** around $\mathbf{x_{B2}}$ an angle ϕ, we get the resulting frame **B** 2.3(d). In this case, clearly:

$$\begin{bmatrix} x_{B} \\ y_{B} \\ z_{B} \end{bmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{bmatrix} x_{B2} \\ y_{B2} \\ z_{B2} \end{bmatrix} \equiv \mathbf{T}_\phi \mathbf{p}_{B2}$$



(a)

(b)

(c)

(d)

FIGURE 2.3 – Complete set of rotations using euler angles.

Now, for a complete transformation, from **0** to **B**, we can use the defined matrices $\mathbf{T}_\psi$, $\mathbf{T}_\theta$ and $\mathbf{T}_\phi$, which multiplied together can give the coordinates of a vector **p**, originally written in $\mathbf{0}$, in the coordinate frame $\mathbf{B}$, that is:

$$\mathbf{p}_B = \mathbf{T}_\phi \mathbf{T}_\theta \mathbf{T}_\psi \mathbf{p}_0 \equiv \mathbf{T}_{\phi\theta\psi}\mathbf{p}_0 \tag{2.6}$$

Note that inverting the matrix $\mathbf{T}_{\phi\theta\psi}$ we can as well pass from $\mathbf{B}$ to $\mathbf{0}$. Verify also (ROBINSON, 1958) that the transformation matrices are always orthogonal (i.e. $T^{-1} = T^T$). For our interest here, we can see that the Euler Angles can help us pass from any reference frame defined in subsection 2.1.1 to another, an thus they allow us to write the equations of motion of the aircraft in terms of the angles which define each transformation.

To finish this brief treatment of coordinate transformations via Euler Angles, it is necessary to cover the kinematics of frames that are moving in relation to each other. That is, suppose they are given two reference frames, one fixed (i.e. $\mathbf{0}$) and another arbitrarily varying in time (i.e. $\mathbf{B}$). We want to relate the angular velocity of $\mathbf{B}$ with the rates of change in time of the euler angles which define the transformation from $\mathbf{0}$ to $\mathbf{B}$. To derive this relation, note that for each rotation just defined ($\mathbf{T}_\psi, \mathbf{T}_\theta$ and $\mathbf{T}_\phi$), we can associate a angular velocity vector ($\dot\psi, \dot\theta$ and $\dot\phi$). Naturally, once we have these vectors, we can use the transformations matrices derived to write them in the body reference frame. The result then *has* to be the body angular velocity vector ω.

Also, verify that each of these angular velocity vectors are always aligned with the axis around which the transformation is done (i.e. $\dot\psi$ is always aligned with $\mathbf{z_0}$), so they can be easily written in their original reference frames, allowing us to write:

$$\begin{gathered}\omega = \begin{bmatrix}\dot\phi\\0\\0\end{bmatrix} + \mathbf{T}_\phi\begin{bmatrix}0\\\dot\theta\\0\end{bmatrix} + \mathbf{T}_\phi\mathbf{T}_\theta\begin{bmatrix}0\\0\\\dot\psi\end{bmatrix}\\ \begin{bmatrix}p\\q\\r\end{bmatrix} = \begin{bmatrix}\dot\phi - \sin\theta\dot\psi\\ \cos\phi\dot\theta + \sin\phi\cos\theta\dot\psi\\ -\sin\phi\dot\theta + \cos\phi\cos\theta\dot\psi\end{bmatrix}\end{gathered} \tag{2.7}$$

Finally, inverting (2.7):

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} p + (r\cos\phi + q\sin\phi)\tan\theta \\ q\cos\phi - r\sin\phi \\ (r\cos\phi + q\sin\phi)\sec\theta \end{bmatrix} \tag{2.8}$$

2.2.2 Flight Simulation

We are now ready to write the flight equations of motion. We will choose to write all the vectors and the inertia matrix in frame **B**. Returning to the definitions of auxiliary frames and remembering how the forces are written in their frames, we rewrite (2.1) as:

$$\mathbf{T}_{\alpha\beta} \begin{bmatrix} -D \\ Y \\ -L \end{bmatrix} + \mathbf{T}_{\alpha_f\beta_f} \begin{bmatrix} F \\ 0 \\ 0 \end{bmatrix} + \mathbf{T}_{\phi\theta\psi} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = m\left(\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} \right) \tag{2.9a}$$

$$\mathbf{p}^{eng/C^*} \times \mathbf{T}_{\alpha_f\beta_f} \begin{bmatrix} F \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \end{bmatrix} = \mathbb{I}^{A/C^*} \cdot \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \mathbb{I}^{A/C^*} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.9b}$$

With the described above it is possible to simulate aircraft flight. Starting from a initial condition defined by a given attitude (ϕ, θ, ψ), a ground velocity (u, v, w) and a angular velocity (p, q, r) we are able to calculate the aerodynamic forces and moments from (2.2) and (2.3) then the thrust from (2.4). Next, the Euler Angles rates can be calculated from (2.7) and (2.9) can be solved for $(\dot{u}, \dot{v}, \dot{w})$ and $(\dot{p}, \dot{q}, \dot{r})$.

For the sake of completeness, it will be noted that before calculating the aerodynamic coefficients it is necessary to derive the angles α and β from the ground velocity. Lets calculate those for the more general case when there is wind acting in the surroundings of the aircraft.

2.2.2.1 Including the Wind

The total airspeed velocity vector in figure 2.2 is actually a vectorial difference of the ground velocity with the local wind velocity. Hence, in order to calculate α and β it is first necessary to write the total airspeed in the aircraft reference frame **B**. As we already have ${}^{R}\mathbf{v}^{C^*}$ written in **B**, we will consider the wind velocity also written in **B** as ${}^{R}\mathbf{v}^{W} = [\mathbf{W_{xB}}, \mathbf{W_{yB}}, \mathbf{W_{zB}}]_B^T$.[2]

Again, by referring to figure 2.2 and considering the total airspeed we have:

$$\alpha = \arcsin\left(\frac{\mathbf{w} - \mathbf{W_{zB}}}{\sqrt{(\mathbf{u} - \mathbf{W_{xB}})^2 + (\mathbf{w} - \mathbf{W_{xB}})^2}}\right) \tag{2.10a}$$

$$\beta = \arcsin\left(\frac{\mathbf{v} - \mathbf{W_{yB}}}{\sqrt{(\mathbf{u} - \mathbf{W_{xB}})^2 + (\mathbf{v} - \mathbf{W_{yB}})^2 + (\mathbf{w} - \mathbf{W_{zB}})^2}}\right) \tag{2.10b}$$

The angles α adn β in(2.10) are the ones that must be replaced in equations (2.2) and (2.3) to calculate the aerodynamic forces acting on the aircraft. However, there are other sets of angles that can be defined to simplify the form of the dynamic equations for the aircraft: The *inertial wind angles* and the *inertial wind velocity* defined such as:

$$u = V_i \cos\alpha_i \cos\beta_i \tag{2.11a}$$

$$v = V_i \sin\beta_i \tag{2.11b}$$

$$w = V_i \sin\alpha_i \cos\beta_i \tag{2.11c}$$

As the total airspeed V and the aerodynamic angles α and β are more closely related to the forces and moments acting on the aircraft (as seen in (2.2), (2.3)) we will choose to write the equations of motion with respect to the inertial wind angles and inertial velocity instead of (u, v, w). Since (2.9a) is normally solved for $(\dot{u}, \dot{v}, \dot{w})$, it is necessary to derivate

[2] *Normally, the wind velocity components are given as a vectorial function of position and time in the inertial reference frame.This should pose no additional difficulty for us, since its just necessary to multiply this wind vector by the transformation matrix* $\mathbf{T}_{\phi\theta\psi}$ *to obtain the wind components in the body frame.*

(2.11) and solve for $(\dot{V}_i, \dot{\alpha}_i, \dot{\beta}_i)$, obtaining:

$$\begin{aligned}
\dot{V}_i &= \mathbf{C}_{\alpha_\mathbf{i}}\mathbf{C}_{\beta_\mathbf{i}}\dot{u} + \mathbf{S}_{\beta_\mathbf{i}}\dot{v} + \mathbf{S}_{\alpha_\mathbf{i}}\mathbf{C}_{\beta_\mathbf{i}}\dot{w} \\
\dot{\alpha}_i &= -\frac{\mathbf{S}_{\alpha_\mathbf{i}}}{\mathbf{C}_{\beta_\mathbf{i}}V_i}\dot{u} + \frac{\mathbf{C}_{\alpha_\mathbf{i}}}{\mathbf{C}_{\beta_\mathbf{i}}V_i}\dot{w} \\
\dot{\beta}_i &= \frac{1}{V_i}\left(-\mathbf{C}_{\alpha_\mathbf{i}}\mathbf{S}_{\beta_\mathbf{i}}\dot{u} + \mathbf{C}_{\beta_\mathbf{i}}\dot{v} - \mathbf{S}_{\alpha_\mathbf{i}}\mathbf{S}_{\beta_\mathbf{i}}\dot{w}\right)
\end{aligned} \tag{2.12}$$

2.2.3 Same equations, different presentation

To deal with (2.9) from a dynamic system point of view (3.1) it is necessary to present them in a state-space form. To do this, we have considered the state $X = [V_i, \alpha_i, \beta_i, \phi, \theta, \psi, p, q, r]^T$ and the control $U = [F, \mathrm{d}_p, \mathrm{d}_a, \mathrm{d}_r]^T$.

The system is then obtained part by part: The (V_i, α_i, β_i) dynamics by solving (2.9a) and substituting the values for $(\dot{u}, \dot{v}, \dot{w})$ in (2.12); the euler angles dynamics by (2.8) and the angular rates dynamics by solving (2.9b) for $(\dot{p}, \dot{q}, \dot{r})$.

After a fair bit of algebra, we arrive at the equations described below:

$$\begin{bmatrix} \dot{V}_i \\ \dot{\alpha}_i \\ \dot{\beta}_i \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} -\frac{D}{m} + g_1 \\ q - tan\beta_i(p\mathbf{C}_{\alpha_\mathbf{i}} + r\mathbf{S}_{\alpha_\mathbf{i}}) + \frac{1}{V_i\mathbf{C}_{\beta_\mathbf{i}}}\left(\frac{-L_0}{m} + g_2\right) \\ p\mathbf{S}_{\alpha_\mathbf{i}} - r\mathbf{C}_{\alpha_\mathbf{i}} + \frac{1}{V_i}\left(\frac{Y_0}{m} + g_3\right) \\ p + (r\cos\phi + q\sin\phi)\tan\theta \\ q\cos\phi - r\sin\phi \\ (r\cos\phi + q\sin\phi)\sec\theta \\ i_1pq + i_2qr + i_{p\mathcal{L}}\mathcal{L}_0 + i_{p\mathcal{N}}\mathcal{N}_0 \\ i_3pr + i_4(-p^2 + r^2) + i_{\mathcal{M}}\mathcal{M}_0 \\ i_5pq + i_6qr + i_{r\mathcal{L}}\mathcal{L}_0 + i_{r\mathcal{N}}\mathcal{N}_0 \end{bmatrix} + \begin{bmatrix} \frac{T_{fV}}{m}F \\ \frac{1}{mV_i\mathbf{C}_{\beta_\mathbf{i}}}\left(T_{f\alpha}F + L_{\mathrm{d}p}\mathrm{d}p\right) \\ \frac{1}{mV_i}\left(T_{f\beta}F + Y_{\mathrm{d}r}\mathrm{d}r\right) \\ 0 \\ 0 \\ 0 \\ K_{p\mathrm{d}a}\mathrm{d}a + K_{p\mathrm{d}r}\mathrm{d}r \\ K_{q\mathrm{d}p}\mathrm{d}p \\ K_{r\mathrm{d}a}\mathrm{d}a + K_{r\mathrm{d}r}\mathrm{d}r \end{bmatrix} \tag{2.13}$$

Where:

$$\begin{aligned} g_1 &= g(-\mathbf{S}_\theta \mathbf{C}_{\alpha_\mathrm{i}} \mathbf{C}_{\beta_\mathrm{i}} + \mathbf{C}_\theta \mathbf{S}_\theta \mathbf{S}_{\beta_\mathrm{i}} + \mathbf{C}_\theta \mathbf{C}_\phi \mathbf{S}_{\alpha_\mathrm{i}} \mathbf{C}_{\beta_\mathrm{i}}) \\ g_2 &= g(\mathbf{C}_\theta \mathbf{C}_\phi \mathbf{C}_{\alpha_\mathrm{i}} + \mathbf{S}_\theta \mathbf{S}_{\alpha_\mathrm{i}}) \\ g_3 &= g(\mathbf{C}_\theta \mathbf{S}_\phi \mathbf{C}_{\beta_\mathrm{i}} + \mathbf{S}_\theta \mathbf{C}_{\alpha_\mathrm{i}} \mathbf{S}_{\beta_\mathrm{i}} - \mathbf{C}_\theta \mathbf{C}_\phi \mathbf{S}_{\alpha_\mathrm{i}} \mathbf{S}_{\beta_\mathrm{i}}) \end{aligned} \tag{2.14}$$

and:

$$\begin{aligned} T_{fV} &= \mathbf{C}_{\alpha_\mathrm{i}} \mathbf{C}_{\beta_\mathrm{i}} \mathbf{C}_{\alpha_\mathrm{f}} + \mathbf{S}_{\alpha_\mathrm{i}} \mathbf{C}_{\beta_\mathrm{i}} \mathbf{S}_{\alpha_\mathrm{f}} \\ T_{f\alpha} &= \mathbf{S}_{\alpha_\mathrm{f}} \mathbf{C}_{\alpha_\mathrm{i}} - \mathbf{C}_{\alpha_\mathrm{f}} \mathbf{S}_{\alpha_\mathrm{i}} \\ T_{f\beta} &= \mathbf{C}_{\alpha_\mathrm{i}} \mathbf{C}_{\alpha_\mathrm{f}} \mathbf{S}_{\beta_\mathrm{i}} + \mathbf{S}_{\alpha_\mathrm{i}} \mathbf{S}_{\beta_\mathrm{i}} \mathbf{S}_{\alpha_\mathrm{f}} \end{aligned}$$

$D = F_{ad} C_d$	$F_{ad} = \frac{1}{2} \rho V^2 S$
$L_0 = F_{ad}(C_{L0} + C_{L\alpha}\alpha + C_{Lq} q)$	$L_{\mathrm{dp}} = F_{ad} C_{l\mathrm{dp}}$
$Y_0 = F_{ad} C_{y\beta} \beta$	$Y_{\mathrm{dr}} = F_{ad} C_{Y\mathrm{dr}}$
$\mathcal{L}_0 = F_{ad} b(C_{\mathcal{L}\beta}\beta + C_{\mathcal{L}p} p + C_{\mathcal{L}r} r)$	$K_{\mathrm{pda}} = F_{ad} b(i_{p\mathcal{L}} C_{\mathcal{L}\mathrm{da}} + i_{p\mathcal{N}} C_{\mathcal{N}\mathrm{da}})$ $K_{\mathrm{pdr}} = F_{ad} b(i_{p\mathcal{L}} C_{\mathcal{L}\mathrm{dr}} + i_{p\mathcal{N}} C_{\mathcal{N}\mathrm{dr}})$
$\mathcal{M}_0 = F_{ad} cma(C_{\mathcal{M}0} + C_{\mathcal{M}\alpha}\alpha + C_{\mathcal{M}q} q)$	$K_{q\mathrm{dp}} = F_{ad} cma C_{\mathcal{M}\mathrm{dp}}$
$\mathcal{N}_0 = F_{ad} b(C_{\mathcal{N}\beta}\beta + C_{\mathcal{N}p} p + C_{\mathcal{N}r} r)$	$K_{\mathrm{rda}} = F_{ad} b(i_{r\mathcal{L}} C_{\mathcal{L}\mathrm{da}} + i_{r\mathcal{N}} C_{\mathcal{N}\mathrm{da}})$ $K_{\mathrm{rdr}} = F_{ad} b(i_{r\mathcal{L}} C_{\mathcal{L}\mathrm{dr}} + i_{r\mathcal{N}} C_{\mathcal{N}\mathrm{dr}})$

Observations In the equations above, the terms $i_{p\mathcal{L}}, i_{p\mathcal{N}}, i_{\mathcal{M}}, i_{r\mathcal{L}}, i_{r\mathcal{N}}$ and $i_1, \ldots, i_6$ are functions of the therms of the inertia matrix. They will not be presented here not to further augment the already busy notation, but are straightforward to obtain.

Note also that the second term of (2.13) is affine in the control arguments, hence, (2.13) is the aeroplane dynamical system put in the form (3.1), and from there we can work our way in controlling the airplane using the theory developed in the following chapters.

3 Feedback Linearization

Feedback linearization was one the first nonlinear control methods to be applied in aircraft control and for some time remained the paradigm of nonlinear controllers applied to aeronautical problems. We will find in the aeronautical literature the name of *Dynamic Inversion* for this method. The idea behind feedback linearization (SLOTINE; LI, 1991; ISIDORI, 1995) is to change the state variables through a coordinate transformation $\mathbf{Z} = \mathbf{Z}(X)$ (a diffeomorphism in a certain domain Ω of $\mathbb{R}^n$) and to use a control $u = u(X, v)$ so that the new system $[\mathbf{Z}, v]$ is linear or at least partially linear in the new set of coordinates, and that the new control variable v can drive the system to a desired trajectory.

The application of the theory of feedback linearization to the control of aircraft's attitude will be explored. By aircraft piloting, we will be referring to the process of keeping a desired orientation, which are reflected in the equations by the euler angles. This is also called the aircraft *fast dynamics* as in opposite to the usually slower dynamics of the aircraft's velocity.

The advantage of choosing the euler angles as output variables, as it will be seen, is that a relatively easy implementation of dynamic inversion can be performed. On the other hand, one has to keep in mind that the wind angles (α,β) cannot fall out of a certain domain to keep the validity of the equations used.

3.1 Definition of the control problem

Before starting the development of the theory of this chapter and also as a base framework for the development of the chapters to follow, let us now define our control problem.

The problem which we will deal is that of controlling the systems of the form:

$$\begin{cases} \dot{X} = f(X) + g(X)u & X \in \mathbb{R}^n \quad u \in \mathbb{R}^m \\ y = h(X) & y \in \mathbb{R}^m \end{cases} \tag{3.1}$$

Where $f : \mathbb{R}^n \to \mathbb{R}^n$; $g : \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^m$ and $h : \mathbb{R}^n \to \mathbb{R}^m$ are nonlinear vector fields. It is important to notice that we are limiting ourselves to the case of systems affine in the control u and to the cases where $dim(u) = dim(y) = m$, the so called *square systems*.

The objective of *controlling* a system of the form (3.1) is to find a control $u(t)$ that can drive the output $y(t)$ to follow a desired trajectory $y_d(t)$. Desired characteristics for this "tracking" is that it be stable (i.e. any perturbations of limited modulus will not drive the airplane to leave its desired trajectory) and with a small error (i.e. $y(t) - y_d(t) \approx 0$).

Instead of linearizing (3.1) around an operating point and using the well developed linear control techniques, we will try to hold the consideration of nonlinearities and develop control laws that can cover a wider set of states to be controlled without changing the controller parameters.

3.2 The SISO Case

Let us begin the development by considering a system of the form:

$$\begin{cases} \dot{X} = f(X) + g(X)u & X \in \mathbb{R}^n \quad u \in \mathbb{R} \\ y = h(X) & y \in \mathbb{R} \end{cases} \tag{3.2}$$

And let $\Omega \subset \mathbb{R}^n$ be the domain of interest. This is a system with a **S**ingle **I**nput u and a **S**ingle **O**utput y. Lets take the derivative of y with respect to time to obtain:

$$\dot{y} = \nabla h \cdot \dot{X} = \nabla h \cdot (f(X) + g(X)u) \tag{3.3}$$

Now, if $\nabla h \cdot g(X)$ is zero for all $X \in \Omega$, $\Omega \subset \mathbb{R}^n$, we have to continue to derivate (3.3) in order for the control u to appear. To simplify the notation, before we continue, lets introduce the so called *Lie Derivatives* (ISIDORI, 1995).

Definition 1 *Let h: $\mathbb{R}^n \to \mathbb{R}$ be a smooth scalar function and $\mathbf{f}$: $\mathbb{R}^n \to \mathbb{R}^n$ be a smooth vector field on $\mathbb{R}^n$. Then the* $\underline{\textit{Lie Derivative of } h \textit{ with respect to } \mathbf{f}}$ *is a scalar function defined by: $L_f h \equiv \nabla h \cdot \mathbf{f}$*

Repeated Lie Derivatives can be defined recursively as follows:

$$\begin{aligned} &L_f^0 h = h \\ &L_f^i h = L_f L_f^{i-1} h = \nabla \left(L_f^{i-1} h \right) \cdot \mathbf{f} \quad i = 1,2,\dots \end{aligned}$$

With the definition above we can write (3.3) as:

$$\dot{y} = L_f h + L_g h\, u$$

And then, in the considered case of $L_g h = 0$ for all $X \in \Omega$, we derivate (3.3) again:

$$\ddot{y} = L_f^2 h + L_g L_f h\, u \tag{3.4}$$

If $L_g L_f h = 0$ for all $X \in \Omega$ we continue to derivate until:

$$y^{(r)} = L_f^r h + L_g L_f^{r-1} h\, u; \quad L_g L_f^{r-1} h \neq 0 \text{ for some } X \in \Omega \tag{3.5}$$

Now, consider the subset of Ω, Ω' which consists of all the points X' where $L_g L_f^{r-1} h \neq 0$. In Ω', consider the following control law:

$$\begin{aligned} u &= [L_g L_f^{r-1} h]^{-1} \left(v - L_f^r h \right) \\ v &= y_d^{(r)} - \sum_{i=0}^{r-1} k_i \left(y^{(i)} - y_d^{(i)} \right) \end{aligned} \tag{3.6}$$

such that the coefficients k_i are chosen to render the polynomial:

$$\mu(\lambda) = \lambda^r + \textstyle\sum_{i=0}^{r-1} k_i \lambda^i$$

Hurwitz.

Check the controller scheme as presented in figure 3.1. As we can see, that are two feedback signals. One is to generate the error with respect to the desired trajectory and the other is to cancel the plants nonlinearities (also called the linearization loop (STEVENS; LEWIS, 2003)).

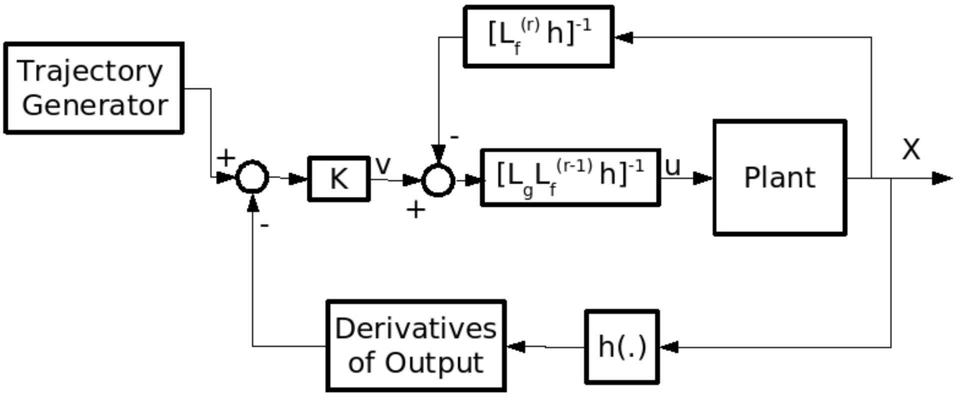


FIGURE 3.1 – The feedback linearization controller block diagram scheme.

We can see that (3.6) will drive the output to follow a desired trajectory $y_d(t)$, because by substituting (3.6) in (3.5), and defining the *error* as $e(t) \equiv y(t) - y_d(t)$ we get:

$$e^{(r)} + \sum_{i=1}^{r-1} k_i e^{(i)} = 0$$

and considering the choice of k_i we have that $e \to 0$ when $t \to \infty$ and thus the output trajectory becomes the same as the desired trajectory.

3.2.1 The Normal Form

Although it was showed one choice of control that can drive the output in the desired way, nothing was said about the rest of the state variables. When a system as (3.2) has an output that can be written in the form (3.5) we call the number r the *relative degree* of the system. Now, there are two possibilities to consider: $r = n$ and $r < n$ (r can never be greater then n (SLOTINE; LI, 1991)). In the first case ($r = n$), we can define a coordinate transformation $\mathbf{Z} = \mathbf{Z}(X)$:

$$\mathbf{Z} = \begin{bmatrix} h \\ L_f h \\ L_f^2 h \\ \vdots \\ L_f^{r-1} h \end{bmatrix}$$

and use control $u = u(X, v)$ with u defined in (3.6) to write the new system as:

$$\dot{\mathbf{Z}} = A\mathbf{Z} + bv$$
$$\text{Where: } A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 \\ 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \text{ and } b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{3.7}$$

In this case, $\mathbf{Z}$ is called the *linearizing state* and (3.6) is called the *linearizing control law* (SLOTINE; LI, 1991). Also (3.7) is called the ***normal form*** of a system whose relative degree r equals the dimension of the state vector (n). The normal form allow us to observe more directly the dynamics obtained by feedback linearization of a system, as it can be noticed by the simplicity of the dynamics written in the new variables.

Now we follow to the case when $r < n$. Let $\mathbf{Z} = [h, L_f h, \ldots, L_f^{r-1} h]^T$. As $dim(\mathbf{Z}) = r$, we have to complete the state transformation with a vector $\mathbf{W}$ such that $dim(\mathbf{W}) = n - r$.

The normal form in this case will be written as:

$$\dot{\mathbf{Z}} = \begin{bmatrix} L_f h \\ L_f^2 h \\ \vdots \\ L_f^{r-1} h \\ \alpha(\mathbf{Z},\mathbf{W}) + \beta(\mathbf{Z},\mathbf{W})u \end{bmatrix} \qquad \dot{\mathbf{W}} = \psi(\mathbf{Z},\mathbf{W}) \tag{3.8}$$

with the output written as $y = Z_1$.

The new state variables $\mathbf{Z}$ and $\mathbf{W}$ are called the *normal coordinates* of the system in the domain Ω' where the transformation is valid. Of course, we would have to prove that first, it exists a coordinate transformation of the form (3.8) where $\mathbf{Z}$ and $\mathbf{W}$ are the new state; then, that this transformation is indeed a local *diffeomorphism*. We will not deal with this matter here, please refer to (SLOTINE; LI, 1991; ISIDORI, 1995).

Zero Dynamics When proving that $\mathbf{Z}$ and $\mathbf{W}$ form a local diffeomorphism in the domain Ω' it is also noticed the interesting fact that the vector $\mathbf{W}$ form a part of the state that is unobservable by the control u. That is, writing $\mathbf{W} = [\psi_1(X), \ldots, \psi_{n-r}(X)]^T$, it is proved that $\nabla\psi_i\, g = 0$ for $i = 1, \ldots, n-r$. The practical consequence is that no matter what we try with the command u, when $r < n$, there will always be a part of the state over which we will not be able to "reach".

This part of the state is called the *zero dynamics*. That is because if we adjust the control u in (3.8) to cancel out all the components of the $\mathbf{Z}$ normal coordinates part, the output will be guaranteed to continue zero. On the other hand, nothing is known about the evolution of the internal variables in $\mathbf{W}$.

The zero dynamics plays a very important role when applying the method of feedback linearization, once it has to be guaranteed stable. To study the dynamic behavior of the zero dynamics, methods of Lyapunov analysis can be used (see section 4.2). If this part of the state is discovered to be unstable, a different choice of output must be selected (SLOTINE; LI, 1991; STEVENS; LEWIS, 2003).

3.3 The MIMO Case

The procedure for feedback linearization of **M**ultiple **I**nput, **M**ultiple **O**utput nonlinear systems is analogous to the SISO case. Lets consider the system (3.1). Differentiating output y_i for r_i times until at least one of the inputs appears in the equation for y_i we have:

$$
\begin{gathered}
y_i^{(r_i)} = L_f^{r_i} h_i + \sum_{j=1}^{m} \left(L_{g_j} L_f^{r_i - 1} h_i \, u_j \right) \\
Where: \quad L_{g_j} L_f^{r_i-1} h_i \neq 0 \text{ for at least one } j \text{ in } \Omega_i'
\end{gathered}
\tag{3.9}
$$

Performing the same procedure for all the outputs, we can write:

$$
\begin{gathered}
\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} L_f^{r_1} h_1 \\ \vdots \\ L_f^{r_m} h_m \end{bmatrix} + \mathbf{E}(X)\, u \\
Where: \ \mathbf{E}_{i,j} = L_{g_j} L_f^{r_i-1} h_i \quad i,j = 1, \dots, m
\end{gathered}
\tag{3.10}
$$

Now, let Ω^T be the intersection of all the sets Ω_i'. If $\mathbf{E}$ is invertible in Ω^T then we can choose the control:

$$
u = \mathbf{E}^{-1} \begin{bmatrix} v_1 - L_f^{r_1} h_1 \\ \vdots \\ v_m - L_f^{r_m} h_m \end{bmatrix} \tag{3.11}
$$

Obtaining the equations $y_i^{(r_i)} = v_i$. Finally, we choose v_i to make the system converge to each desired trajectory $y_{id}(t)$.

The number $r = \sum_{j=1}^{m} r_j$ is called the *total relative degree* of the system (3.1). The definitions for zero dynamics can be extended also to the MIMO case. When $\mathbf{E}$ is a singular matrix in some point of the domain of interest, the outputs have to be modified or the output has to be sequentially differentiated to obtain a new system where $\mathbf{E}$ is invertible (SLOTINE; LI, 1991).

To clarify the procedure, the next sections this exact procedure will be applied to the euler angle dynamics of the aircraft.

3.4 Summary of the theory

The application to feedback linearization to aircraft control can bring a good number of advantages. For instance, the simulated controller code can be programmed directly in the avionics of the aircraft, with no needs for look up tables or gain scheduling, once this controller can perform equally in any flight condition. This method can also be used both by stabilizing and tracking problems in severe, large amplitude manoeuvres, as long as the physical description of the aircraft remain the same.

It also naturally has some important limitations. In the case of aircraft control, we can emphasize two inconveniences:

- We have to measure the entire state to use the controller. This is not easy from a practical point of view and can get specially hard in the case of nonlinear systems, where observers are harder to construct and the convergence of the coupling between the observer and the control system may not always be proofed.
- We need a complete, precise model of the system dynamics. Although it is possible to have a fair description of the airplane dynamics (see chapter 2), the values used are not always (normally never) known precisely enough, due to environment perturbations, fabrication defects or just a simplified analysis of the system. The not so well modelled dynamics can not be accounted for in a formal fashion in the case of feedback linearization (robustness issues (SLOTINE; LI, 1991)), but there are ways of trying to solve this problem, as the adaptive control schemes (section 5).

3.5 Application to Flight Control

Let us now apply what was presented to the case of controlling the attitude of an aircraft.

3.5.1 Dynamic Inversion of Aircraft Fast Dynamics

Since we are, for the time being, only interested in the orientation dynamics, let us take the last six equations out of (2.13), that is, we will be dealing with the state $X = [\phi, \theta, \psi, p, q, r]$ and the control $U = [\mathrm{d}_p, \mathrm{d}_a, \mathrm{d}_r]$. Our output will be $Y = [\phi, \theta, \psi]$ so that our system will be square. Then we have:

$$\begin{aligned} \dot{X} &= f(X) + g(X)U \\ Y &= HX \end{aligned} \tag{3.12}$$

Where f and g are defined as indicated in (2.13) and H is:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Following the procedure of feedback linearization for the MIMO case and considering the simple nature of matrix H, after derivation the output Y, we get:

$$\dot{Y} = H\left(f(X) + g(X)U\right)$$

but as $Hg(X) = 0$ we get:

$$\dot{Y} = Hf(X) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

by adopting as notation f_i for indicating the i^{th} line of f and $g^{<j>}$ for indicating the j^{th} column of g. As the input variables do not appear in the expression for $\dot{Y}$ (that is,

$L_g H = 0$), we continue derivating to obtain:

$$\ddot{Y} = \begin{bmatrix} \nabla f_1 \cdot f(X) \\ \nabla f_2 \cdot f(X) \\ \nabla f_3 \cdot f(X) \end{bmatrix} + E(X)U \equiv \Psi(X) + E(X)U$$

$$\text{Where: } E_{i,j} = \nabla f_i \cdot g^{<j>} \tag{3.13}$$

Now let us suppose we would like each of our euler angles to behave with a dynamics equivalent to:

$$\ddot{x}_c + 2\xi\omega_n\dot{x}_c + \omega_n^2 x_c = 0$$

where the values for ξ and ω_n are chosen for each euler angle.

For the subsets of the domain of interest where E is invertible we choose U such as:

$$U = E^{-1}(X)\left(-\Psi(X) - 2\Xi\Omega_n\dot{Y} - \Omega_n^2(Y - Y_d)\right)$$

$$\text{With: } \Xi = \begin{bmatrix} \xi_\phi & 0 & 0 \\ 0 & \xi_\theta & 0 \\ 0 & 0 & \xi_\psi \end{bmatrix} ; \ \Omega_n = \begin{bmatrix} \omega_{n\phi} & 0 & 0 \\ 0 & \omega_{n\theta} & 0 \\ 0 & 0 & \omega_{n\psi} \end{bmatrix} \tag{3.14}$$

with Y_d the vector for the desired Euler angles, so that we will obtain exactly the control deflections needed to achieve the desired behaviour of the outputs.

3.5.1.1 Euler Angles Controller

The control law obtained in (3.14) is summarized in the block diagram shown in figure 3.2. Note that by feeding back the nonlinear terms in $\Psi(X)$ the proportional and derivative gains become directly the values of the desired dynamic characteristics, which is a consequence of the transformation we are doing in the state coordinates (section 3.2.1). Note that since we derived twice each output to obtain expressions dependent of the control variables, the *total relative degree* of the system is six. Although this is the same number of state variables chosen for our simplified system, we will keep in mind that our

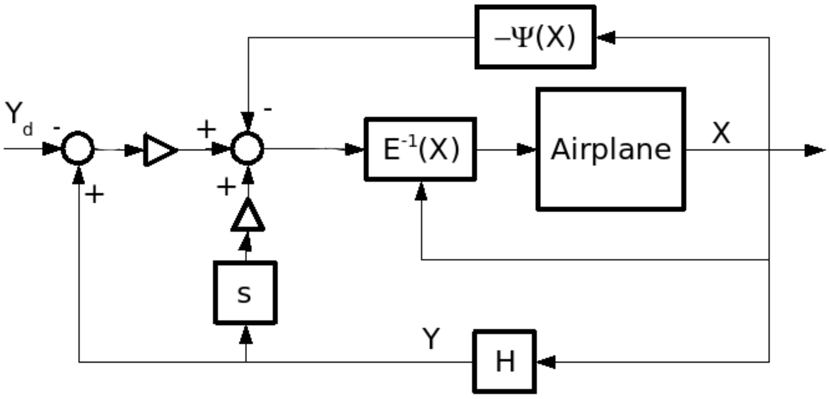


FIGURE 3.2 – Block Diagram for the Feedback Linearising controller.

system here represents only a part of the airplane complete dynamics. The values of the other state variables (V, α, β) remained "hidden" during the feedback linearization process, and the thus we are left with an internal dynamics (the zero dynamics), for which the stability needs to be checked. Instead of presenting a theoretical proof of the stability of the variables $(V, \alpha\beta)$ we will leave to the simulation results to show that for the cases of interest, all the variables will behave acceptably.

As one can see in (2.13), the heading angle ψ does not take part in the dynamic equations, but it is taken here as an output to keep the system *square* and avoid further control allocation problems since we will be dealing with a three control surface aircraft. During the simulation though, the heading angle desired value always coincides with the instantaneous heading angle, so that its dynamics do not influence the controls.

3.5.2 Simulation

The dynamic simulation model used for all the results presented in this work is the Navion's model presented in appendix A. The model also considered the actuators saturation and dynamics with the objective of seeing controller response to phenomena not considered in controller design. The results are presented in the following order:

- Initially, the results for separate longitudinal and lateral controls are showed, the first command is a step of 45° for the angle θ and the second a step of 50° for the angle ϕ. In the first set of graphics, the response of a classic PD controller is showed

TABLE 3.1 – Various controller parameters for simulation.

Nonlinear Controller		PD Controller	
$\xi_\phi = 0.7$	$\omega_{n\phi} = 4$	$K_P = 2$	$K_D = 1.5$
$\xi_\theta = 0.7$	$\omega_{n\theta} = 2.8$		
$\xi_\psi = 0$	$\omega_{n\psi} = 0$		
Parameter Changing			
$C^*_{m0} = 0.5C_{m0}$	$C^*_{mq} = 0.5C_{mq}$		
$C^*_{mdp} = 0.5C_{mdp}$			

for comparison.

- Then, control commands that demand considerable mixing between the surface commands are performed.
- Finally, the values of the parameters used in the controller to describe aircraft dynamics are changed, to see what sort of response is to be expected from poorly modelling of aircraft dynamics.

Results The controller parameters used in the simulation are shown in table 3.1. The parameters marked with an asterisk are the parameters used to perform the surface deflection calculations in the last simulation case.

3.6 Commentaries

We see first, by figure 3.3 that even in the presence of some actuator saturation there is a good controller response even to large steps in the output variables. The PD controller used for comparison had it's gains adjusted to present the minimum possible final integral error for precisely this step case. Although this is not a standard method in classical controller design, it can give us a reference for comparison in the sense that we cannot expect even better responses from that scheme. Also, it will be noticed here that the gains used for this step will naturally not give the same response for steps of other amplitudes, while the nonlinear controller presents the same design parameters for all flight conditions.

The nonlinear controller is capable of following the reference model as long as no

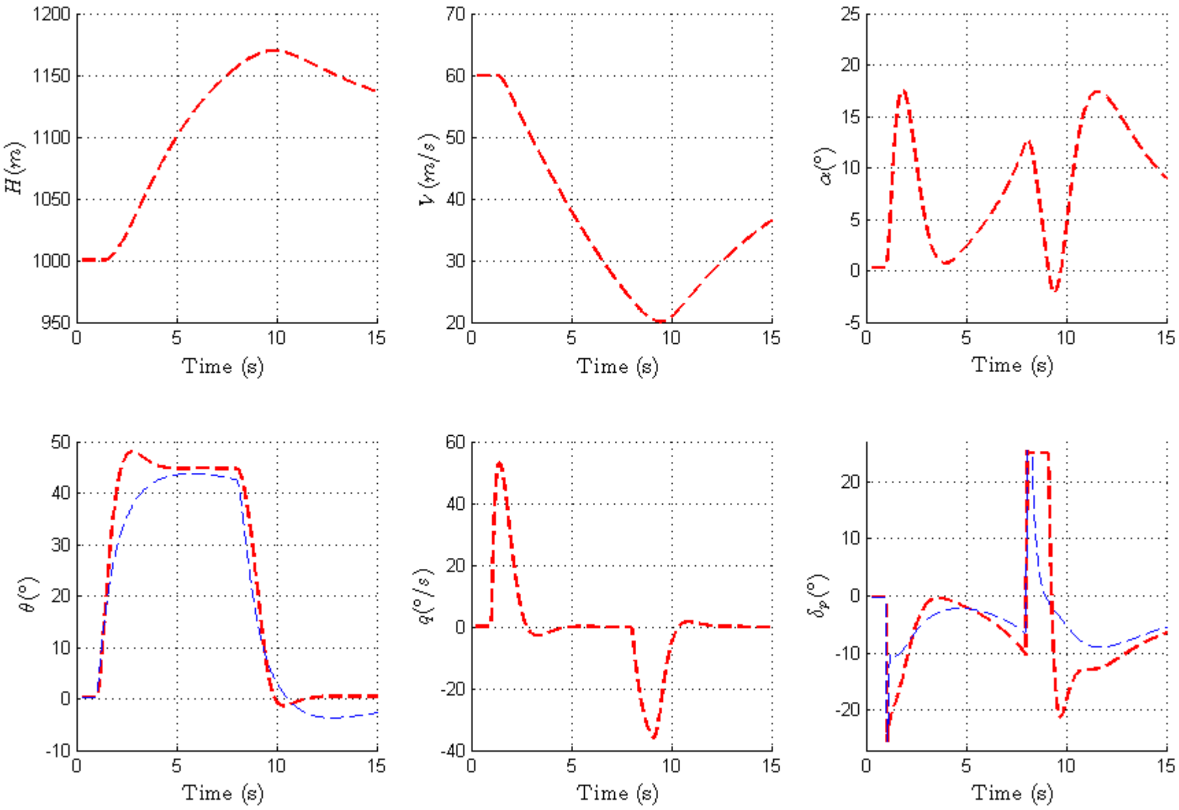


FIGURE 3.3 – Step response for elevation angle of nonlinear controller (red). Classical PD response in blue for comparison.

control saturation is present, or while the controller dynamics is not too slow. Of course, the amount of saturation and the energy demand from the control channel are directly related to the reference model chosen. Given the plant dynamics, the model dynamic parameters are the only to be adjusted and these will regulate both the quality of response and how much control effort will be used to achieve it.

The same qualitative behaviour is seen for the cases of a step in roll angle and the mixed pitch-roll command (figures 3.4 and 3.5) where the nonlinear controller was able to follow the desired behaviour.

Finally, figure 3.6 shows the controller behaviour when the plant parameters are not well modelled. In this case, the controller is still capable of reasonably follow the reference model, but the errors become naturally more evident. More important then that, it is seen by the surface deflection that the more the plant model used does not correspond to reality, the more energy will be demanded from the control surface. This will account for

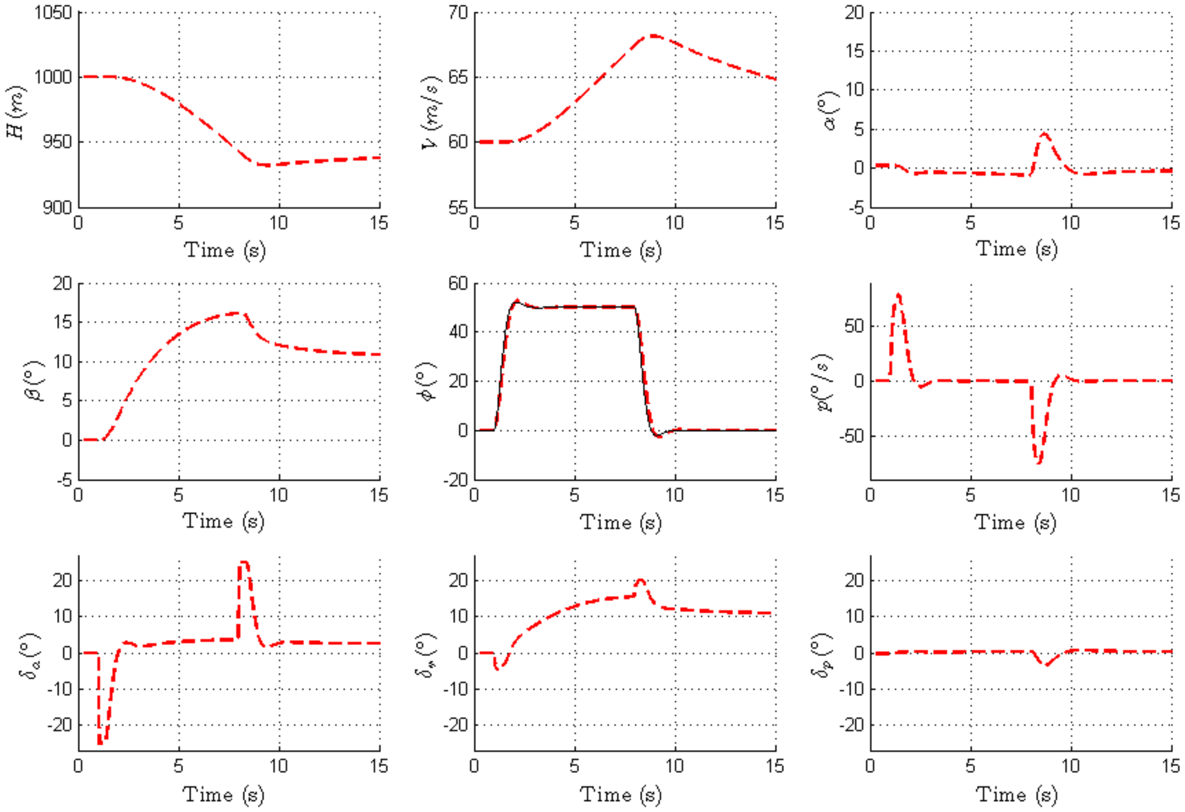


FIGURE 3.4 – Step response for roll angle of nonlinear controller.

a higher risk of saturation of control surface and poorer controller response.

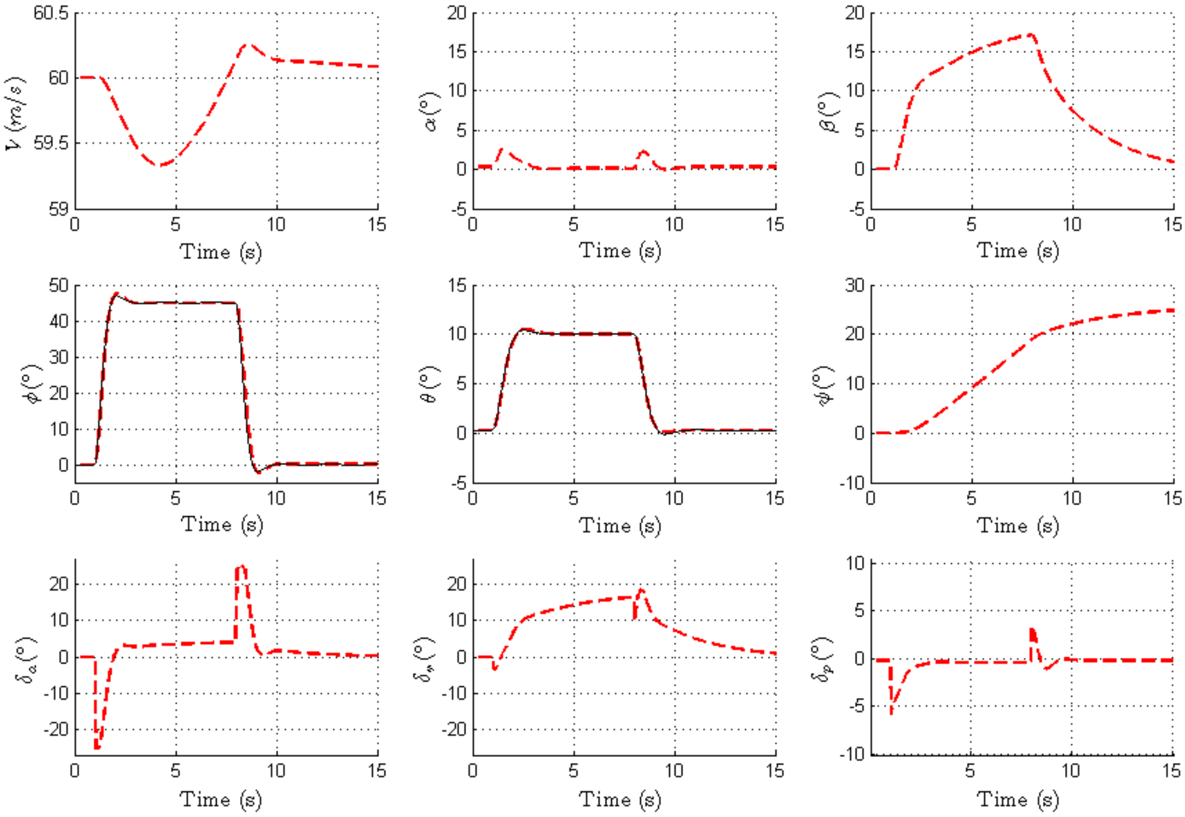


FIGURE 3.5 – Step response for mixed elevation and roll angles commands of nonlinear controller.

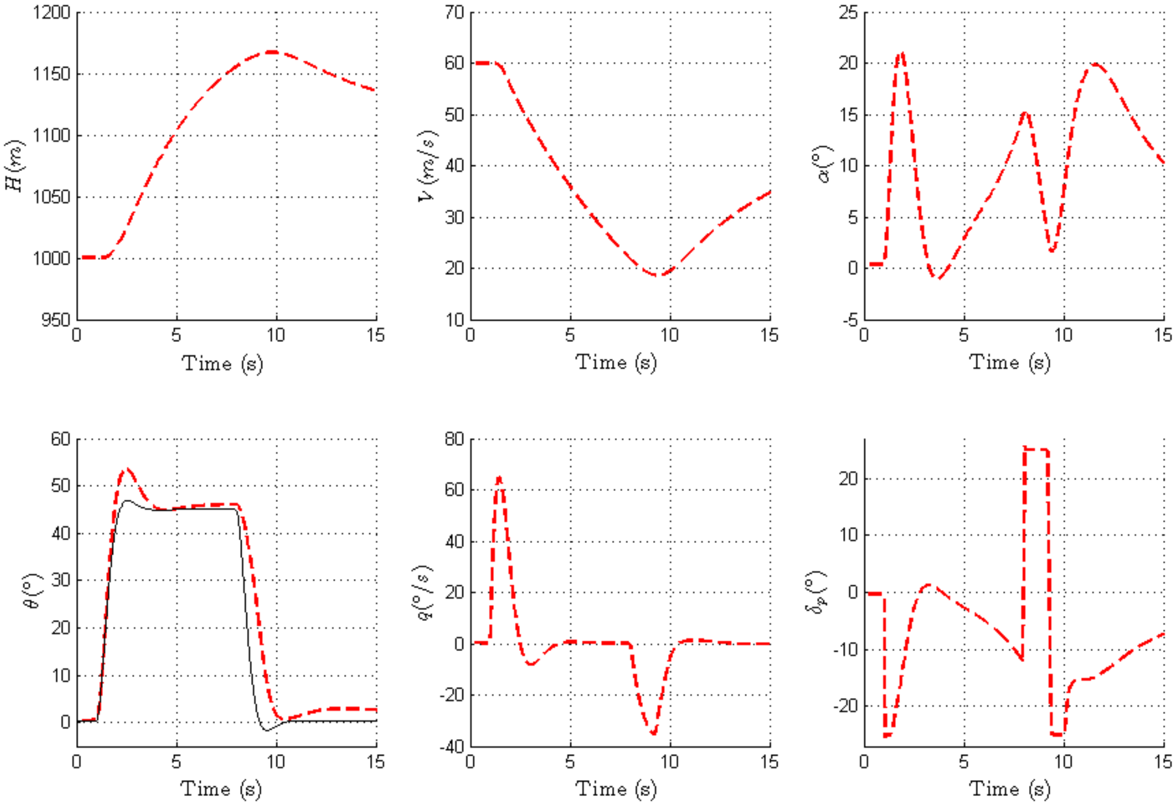


FIGURE 3.6 – Step response for elevation angle commands of nonlinear controller with bad parameter estimation.

4 Backstepping

4.1 Introduction

The Backstepping Method is derived directly from the study of the *Lyapunov's Direct Method*, a general method for finding stability properties of nonlinear dynamic systems, and its extensions. According to (FOSSEN; STRAND, 1998), it is a *recursive* design methodology for finding both the feedback control laws and *Lyapunov Functions* in a systematic manner.

Historically, the idea of the integrator backstepping has been developed in the end of the decade of 1980 (FOSSEN; STRAND, 1998) and soon developed to cascaded systems analysis, which permitted the design of complete nonlinear systems that could be put in an specific form: *the strict feedback form.*

One good advantage of Lyapunov Analysis, an thus backstepping design, is that it allows one to use physical properties of a system, as dissipation of energy, natural damping or potential force fields, in advantage of the controller, thus ignoring "good" nonlinearities, that is, characteristics of the system that are inherently stable, and focusing the control effort where it is really needed, that is, in that part of the dynamics which is not naturally stable (FOSSEN; STRAND, 1998; HARKEGARD, 2001b; HARKEGARD, 2001a).

4.2 Lyapunov Analysis of Nonlinear Systems

In this subsection we will deal first with *Autonomous Systems*, that is, systems whose description and parameters do not vary with time. After the major results are extended to the case of *Non-Autonomous Systems* which is the case of the airplane (it burns fuel, deploys the landing gear, deflects high lift surfaces, etc) and also the case when, even considering the airplane an autonomous system for a given flight condition, adaptive control methods are used.

Consider then the following nonlinear time-invariant system:

$$\dot{X} = \mathbf{f}(X); \quad \mathbf{x}(0) = \mathbf{x_0} \tag{4.1}$$

Let $\mathbf{x}(t)$ be a solution and $\mathbf{x_e}$ be an equilibrium point of (4.1) (i.e. $\mathbf{f}(\mathbf{x_e}) = \mathbf{0}$).

Definitions Lets list a set of definitions necessary to well pose the theorem of Lyapunov's direct method and the invariant set theorem (LaSalle's theorem) which together form a powerful tool of analysis of stability for nonlinear autonomous systems.

Definition 2 *If $\mathbf{f}(X)$ is a continuous function in $\mathbf{x_e}$ and there exists a strictly positive real number $\boldsymbol{L}$ such that:*

$$\| \mathbf{f}(X_1) - \mathbf{f}(X_2) \| \leqslant \boldsymbol{L} \| X_1 - X_2 \|$$

For all X_1 and X_2 in a finite neighborhood of $\mathbf{x_e}$ then $\mathbf{f}$ is said to be ***locally Lipschitz*** *in x_e.*

Definition 3 $\mathbf{x}$*(t) is* ***bounded*** *if there exists a non-negative constant $\gamma(\mathbf{x}(t))$ such that:*

$$\| \mathbf{x}(t) \| \leqslant \gamma(\mathbf{x}(t)), \quad \forall t \geqslant 0$$

Definition 4 $\mathbf{x_e}$ *is* ***stable*** *if $\forall \epsilon > 0$ there exists $\delta(\epsilon) > 0$ such that:*

$$\| x(0) - x_e \| < \delta(\epsilon) \Rightarrow \| x(t) - x_e \| < \epsilon \quad \forall t \geqslant 0$$

And $\mathbf{x_e}$ *is* ***unstable*** *when not stable.*

Definition 5 $\mathbf{x_e}$ *is* ***attractive*** *if there exists a non-negative real function* $\mathbf{r}(\mathbf{x}(t))$ *such that:*

$$\parallel x(0) - x_e \parallel \leqslant \mathbf{r}(\mathbf{x}(t)) \Rightarrow \lim_{t\to\infty} \parallel x(t) - x_e \parallel \to 0$$

In this case, we can define a subset Ω *of* $\mathbb{R}^n$ *such that* $\mathbf{x_e} \in \Omega$ *and that every* $\mathbf{X}(0) \in \Omega$ *will produce a trajectory such that* $\lim_{t\to\infty} \parallel \mathbf{x}(t) - \mathbf{x_e} \parallel \to 0$*. Then,* Ω *is called* ***the domain of attraction*** *of* $\mathbf{x_e}$

Definition 6 $\mathbf{x_e}$ *is (locally)* ***asymptotically stable*** *(AS) if it is* ***stable*** *and* ***attractive****.*

Definition 7 $\mathbf{x_e}$ *is* ***globally asymptotically stable*** *(GAS) if it is* ***stable*** *and* ***attractive*** *for all* $\mathbf{X}(\mathbf{0})$*. That is, the* ***the domain of attraction*** *of* $\mathbf{x_e}$ *is the whole* $\mathbb{R}^n$*.*

Definition 8 *A scalar field* $V : \mathbb{R}^n \to \mathbb{R}_+$ *is said to be* ***positive (semi-) definite*** *in a ball* $B_{R_{X_e}}$ *if:*

$$If\ X \neq X_e \Rightarrow V(X) > 0 \quad (V(X) \geqslant 0) \quad \forall X \in \mathbb{R}^n$$

And ***negative (semi-) definite*** *if -V(X) is* ***positive (semi-) definite****.*

Definition 9 *A scalar field* $V : \mathbb{R}^n \to \mathbb{R}_+$ *is* ***radially unbounded*** *if:*

$$\parallel V(X) \parallel \to \infty \quad when \quad \parallel X \parallel \to \infty$$

Definition 10 $G \subset \mathbb{R}^n$ *is said to be an* ***invariant set*** *of a dynamical system if all the trajectories of the system starting in G remain in G for all future time.*

Now we are ready to define the theorems:

Theorem 1 (Lyapunov's direct method) *Let* $\mathbf{x_e}$ *be a equilibrium point of* (4.1) *and* $\mathbf{f}(X)$ *be locally Lipschitz in* x_e*. Let* $V : \mathbb{R}^n \to \mathbb{R}$ *have continuous first partial derivatives, be positive definite and radially unbounded. In such conditions:*

- *x_e is globally stable (GS), if $\dot{V}(X)$ is negative semi-definite.*
- *x_e is globally asymptotically stable (GAS), if $\dot{V}(X)$ is negative definite.*

Theorem 2 (LaSalle's invariant set theorem) *Consider the system in (4.1), with $\mathbf{f}(X)$ a continuous function. Let $V : \mathbb{R}^n \to \mathbb{R}$ have/be:*

- *continuous first partial derivatives*
- *positive definite*
- *radially unbounded*
- *$\dot{V}(X)$ is negative semi-definite*

Let also Ω be the set of all points in $\mathbb{R}^n$ where $\dot{V}(X) = 0$.

If M is the largest invariant set in Ω, then all the trajectories of (4.1) asymptotically converge to M.

The proofs of these theorems can be found in (SLOTINE; LI, 1991; KHALIL, 2002; ISIDORI, 1995). ■

4.2.1 An heuristic view of Lyapunov's direct method

We now analyse what does the theorems above say from an informal, more geometric manner. To be able to picture it, lets think of a state vector of two dimensions, that is: $X \in \mathbb{R}^2$. Then every trajectory of the system can be drawn in the plane, given the starting point $X(0)$ and the function $\mathbf{f}(X)$.

More than that, we can geometrically recognize all the definitions above. An equilibrium point X_e, for example, is such that if we start our trajectory there, we will not move. This equilibrium point will be stable if every time we start a trajectory "close enough" to it, we will stay "close enough" to it for all future times. It is attractive with a domain of attraction Ω if for all $X(0) \in \Omega$ our trajectory will be a curve starting at $X(0)$ and finishing in X_e, and so on.

Our objective is to find an easier way to guarantee that, no matter what starting point we choose (or at least be able to define a set of starting points that we can choose such that), the dynamical system will converge to an equilibrium point, or stay stable in a given trajectory, without having to directly solve the normally very complex nonlinear differential equation system (4.1).

To do that, consider a scalar field $V : \mathbb{R}^2 \to \mathbb{R}$. We then could also have a geometric representation of $V(X)$ as a surface in the euclidean space. Now, if we say that: $V(X)$ is continuous with continuous first partial derivatives, positive definite and radially unbounded, you could think of something like the surface seen in figure 4.1.

Consider that we will follow a given starting point $X(0)$ in the plane as it moves through its trajectory $X(t)$. Also, lets follow the three-dimensional curve described by $V(X(t))$ and lets call this curve $\zeta(t)$. Evidently, $\zeta(t) \subset V(X)$ (see again figure 4.1).

Now, if we say that $\dot{V}(X)$ is negative semi-definite, it means that at any given time $\zeta(t)$ will be headed "downhill" with respect to the surface $V(X)$ or tangent to a level curve of $V(X)$. In the other hand, $V(X)$ is positive definite and radially unbounded. This means that $V(X)$ *has* a minimum and plus, this minimum is somewhere "around the origin" (proved in (KHALIL, 2002)).

To join this two facts and come to a conclusion, we see that following $\zeta(t)$ we will *inevitably* have to reach the minimum of $V(X)$ or forever stay in a level curve of this scalar function. This reasoning gives direct relationship between the local geometry of the surface $V(X)$ and local dynamic characteristics of the system (4.1).

To cite the first example, lets look at the first result given by the Lyapunov's direct method theorem. If $\dot{V}(X)$ is guaranteed to be negative for all $X \neq X_e$ than no matter what is your starting point $X(0)$ it will describe a trajectory such as it will always be "coming down" the surface until it reaches X_e and then it will finally stop. In other words, X_e in this case will be GAS. Notice also that in this case, X_e is inevitably the

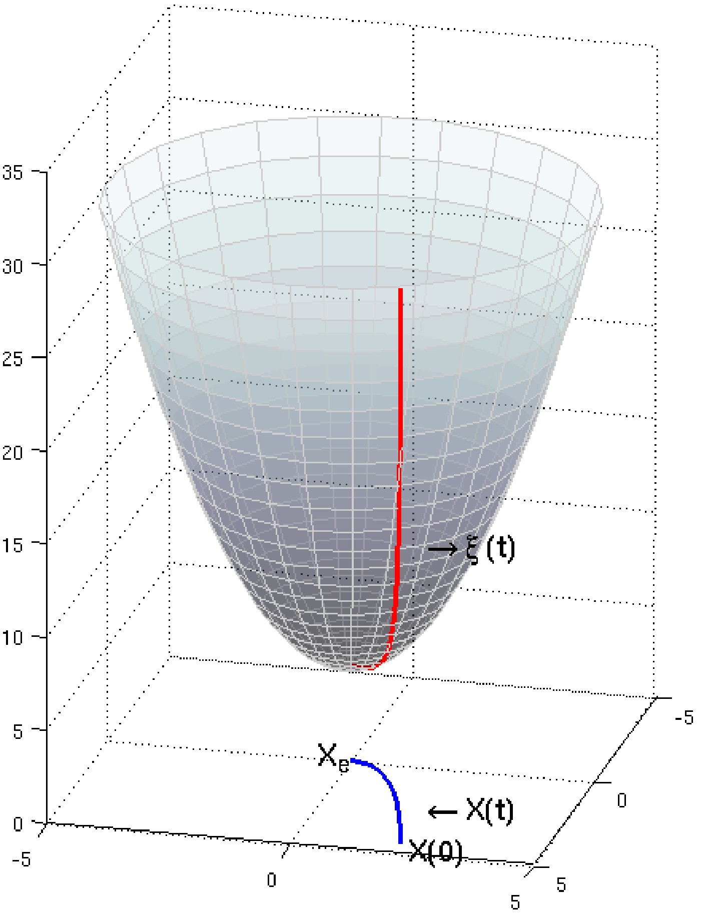


FIGURE 4.1 – What a continuous, positive definite and radially unbounded scalar field can look like.

global minimum of $V(X)$.

The second case, when $\dot{V}(X)$ is negative semi-definite, gives rise to the cases when $V(X)$ does not have only a point as global minimum but a set of points. Plus, it considers the case when $\zeta(t)$ degenerates in a level curve of $V(X)$. LaSalle's theorem takes exactly those possibilities (the set defined by $\Omega = \{X \in \mathbb{R}^n | \dot{V}(X) = 0\}$) and says that when that happens, our trajectory will converge to the union of all invariant sets found in Ω (again check figure 4.1).

Now notice that all this conclusions were drawn having supposed that we already knew a scalar field $V(X)$ that obeyed all the necessary properties for the theorems to be applied. This function is actually so important that any function that satisfy its properties for a given system is called a *Lyapunov Function*.

In practice, however, we have to find (or at worse, guess) a function $V(X)$ that will have the desired properties in the case we are willing to study. Normally, physical arguments can be used to construct this functions. In the case of design of control systems, this becomes a little bit easier, since we can choose our control such as to make a Lyapunov Function and is precisely this method that is described below.

4.2.2 Extension to non-autonomous systems

Lets now generalize the results obtained above for the case of non-autonomous systems, that is, for systems that when written in the form (4.1) will have a function of the form $\mathbf{f}(X,t)$ to describe its dynamics. This includes the systems whose equations can change its parameters with time. We will be interested naturally with smooth changes, or more precisely, with continuous functions $\mathbf{f}(X,t)$ such that $\frac{\partial}{\partial t}\mathbf{f}(X,t)$ is also continuous.

Definition 11 *A function $f:\mathbb{R}\to\mathbb{R}$ is said to be **uniformly continuous** if:*

$$\forall\delta>0,\exists\epsilon(\delta)>0|\forall x,x_0\in\mathbb{R},\parallel x-x_0\parallel<\epsilon\Rightarrow\parallel f(x)-f(x_0)\parallel<\delta$$

Lemma 1 (Barballat's Lemma) *If the differentiable function $f(t)$ has a finite limit as $t\to\infty$ and $\dot{f}(t)$ is uniformly continuous then $\lim_{t\to\infty}\dot{f}(t)=0$*

Theorem 3 *Let $V:\mathbb{R}^n\times\mathbb{R}\to\mathbb{R}$ be a continuous scalar field $V(X(t),t)$, have continuous first partial derivatives and be such that:*

- *$V(x,t)$ is lower bounded*
- *$\dot{V}(x,t)$ is negative semi-definite*
- *$V(x,t)$ is radially unbounded*

Then $\dot{V}(x,t)\to 0$ as $t\to\infty$.

Moreover, in the case of a dynamic system $\dot{X}=\mathbf{f}(X,t)$, if a function $V(X(t),t)$ is found and have the properties above, it approaches a finite limit V_∞ when $t\to\infty$ and $V_\infty\leqslant V(X(0),0)$. Any trajectory of the system will then be globally uniformly bounded if

$\dot{V}(X,t)$ is negative semi-definite and will have a globally asymptotically stable equilibrium point if $\dot{V}(X,t)$ is negative definite.

The analysis of non-autonomous systems using the theorem above then follows closely the analysis of autonomous systems by means of the invariant set theorem (SLOTINE; LI, 1991; KHALIL, 2002). Again, we notice that we still have to find a Lyapunov Function to satisfy all the properties cited.

4.3 Lyapunov Methods Applied to Control

We will now apply the results obtained in the last section to the control of nonlinear systems. Lyapunov-like analysis give a useful way of verifying if a certain equilibrium point is globally asymptotically stable. It would be very nice if we could use our control input to render a given state of interest globally stable, and thus guarantee that our system will converge to this state of interest. That is the basic idea behind backstepping. We will follow here (KHALIL, 2002; HARKEGARD, 2001b).

4.3.1 The Main Result

Consider the following system:

$$\begin{cases} \dot{\eta} = f(\eta) + g(\eta)\xi \\ \dot{\xi} = u \end{cases} \tag{4.2}$$

With $f, g : D \to \mathbb{R}^n$; $u \in \mathbb{R}$ and $f, g \in C^\infty$.

This is described in figure 4.2 below:

Consider that a feedback control law $\xi_{des}(\eta)$ is known to make the origin of the system $\dot{\eta} = f(\eta) + g(\eta)\xi$ GAS. This can be assumed without loss of generality, for any system that have a GAS equilibrium point can fit in this case through a coordinate transformation. Suppose also that we know a Lyapunov Function that proves this stability. That is, we

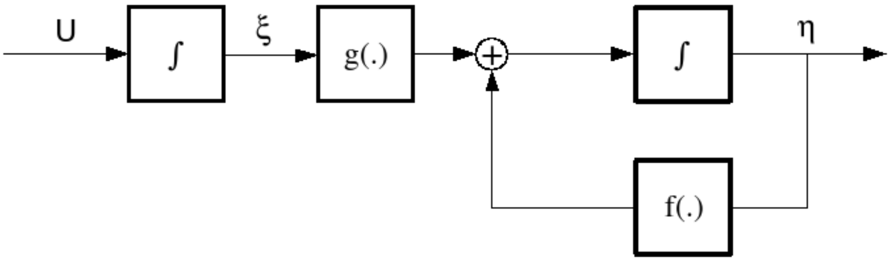


FIGURE 4.2 – A cascaded dynamic system.

know $V_1(\eta)$ such as $\frac{\partial V_1}{\partial \eta}[f(\eta) + g(\eta)\xi_{des}(\eta)] \leqslant -W(\eta)$ where $W(\eta)$ is positive definite.

We do not have "direct access" to the function $\xi(\eta)$ to make it the desired function, unless for the dynamic equation in u, as showed in (4.2). Lets define an error: $z \equiv \xi - \xi_{des}$ so that we can re-write (4.2) as:

$$\begin{cases} \dot{\eta} = f(\eta) + g(\eta)\xi_{des}(\eta) + g(\eta)\, z \\ v \equiv \dot{z} = u - \dot{\xi}_{des} \end{cases} \tag{4.3}$$

This system is completely equivalent to (4.2) with the advantage that the command $v = 0$ stabilizes it in the origin. That means we can propose a *Lyapunov Candidate Function (LCF)* for it as:

$$V_C(\eta, Z) = V_1(\eta) + \frac{1}{2}z^2$$

So that:

$$\dot{V}_C = \frac{\partial V_1}{\partial \eta}[f(\eta) + g(\eta)\xi_{des}(\eta)] + \frac{\partial V_1}{\partial \eta} g(\eta)\, z + zv$$
$$\dot{V}_C \leqslant -W(\eta) + \frac{\partial V_1}{\partial \eta} g(\eta)\, z + zv$$
$$\text{if: } v = -\frac{\partial V_1}{\partial \eta} g(\eta) - kz \text{ with: } k > 0$$
$$\text{We get: } \dot{V}_C \leqslant -W(\eta) - kz^2$$

This shows that the origin $[\eta^T, Z]^T$ is GAS. The control law:

$$u = \frac{\partial \xi_{des}}{\partial \eta}[f(\eta) + g(\eta)\xi] - \frac{\partial V_1}{\partial \eta} g(\eta) - k[\xi - \xi_{des}(\eta)] \tag{4.4}$$

Stabilizes the system (4.2) in the origin.

Notice what we have accomplished. We considered first a system that could be seen as two separated, independent systems (4.2). Then, we supposed only the first part, and treated the variable ξ as a control input for that system. In that case ξ is called the *virtual control* input. Then, from a virtual control that could stabilize the system ξ_{des}, we constructed a new Lyapunov Function Candidate that considered also a new variable: the *error* between the desired value of the virtual control variable and its actual value. Finally, we used our real control u to guarantee that our function candidate was really a Lyapunov Function for the whole system, finishing the recursive process.

This process can naturally be extended. Before doing that, lets consider the more general system:

$$\begin{cases} \dot{\eta} = f(\eta) + g(\eta)\xi \\ \dot{\xi} = f_1(\eta,\xi) + g_1(\eta,\xi)u \end{cases} \tag{4.5}$$

With $f, g, f_1, g_1 \in C^\infty$ and $g_1(\eta,\xi) \neq 0$ in the domain of interest. Then we have that $\exists\ u = \frac{1}{g_1(\eta,\xi)}[u_1 - f_1(\eta,\xi)]$ and if u_1 is given by (4.4), u will make the system (4.5) have a GAS equilibrium point.

Now, to extend the process, suppose that u in (4.5) is also a state variable. Furthermore, suppose that the u dynamics can be written as $\dot{u} = f_2(\eta,\xi,u) + g_2(\eta,\xi,u)v$. We can recursively apply the procedure to find a control v that makes the origin of the new system asymptotically stable. Generalizing, systems that can be written in the form:

$$\begin{aligned} \dot{x_0} &= f(x_0) + g(x_0)x_1 \\ \dot{x_1} &= f_1(x_0,x_1) + g_1(x_0,x_1)x_2 \\ &\vdots \\ \dot{x_n} &= f_n(x_0,\ldots,x_n) + g_n(x_0,\ldots,x_n)u \end{aligned} \tag{4.6}$$

Are desirable for applying the backstepping method, using each state variable as a virtual control until we reach the control u. Systems written in the form (4.6) are said to be in the *strict feedback form*.

4.3.2 A Backstepping Example

This example is directly taken from (HARKEGARD, 2001b). Suppose the following system:

$$\dot{x} = -x^3 + x + u$$

Notice that the uncontrolled system ($u = 0$) is not stable at the origin. We want a control that makes the origin a GAS equilibrium point. Lets consider the LCF:

$$V(x) = \frac{1}{2}x^2$$

Then:

$$\dot{V}(x) = x \cdot \dot{x} = -x^4 + x^2 + xu$$

$$\text{Choosing: } u = -kx \tag{4.7}$$

$$\text{We have: } \dot{V}(x) = -x^4 - (k-1)x^2 \text{ so if: } k > 1 \longrightarrow \dot{V}(x) < 0 \text{ if } x \neq 0.$$

In this case, we did not had to bother with the nonlinear term of the system (x^3), because it was stabilizing. By using (4.7) instead of the feedback linearization control law:

$$u = x^3 - kx \text{ with: } k > 1 \tag{4.8}$$

We avoid to waste control effort in balancing the already stable nonlinear part of the plant as well as keep ourselves from *destabilizing* the system in the case of model error. Just imagine if (4.8) was used in a plant that was *really* described by: $\dot{x} = -0.9x^3 + x + u$, what would happen then?

4.3.3 Summary of the theory

The backstepping technique derives form the Lyapunov or Lyapunov-like analysis of nonlinear systems. Due to the fact that physical arguments are normally used to construct the so called *Lyapunov Functions*, backstepping can also give the ability to take into consideration certain physical characteristics of the system, thus allowing one to cancel only the destabilizing dynamics and keeping the "good" nonlinearities.

On the other hand, the method needs the system to be written in a specific format to be applied and does not guarantee precise dynamic characteristics to the controlled system. That is, it only guarantees a certain point to be stable, but it does not give the control over how does the trajectory will behave until it reaches the desired equilibrium. Lets remember that when stabilizing piloted aircrafts, there are specific scales that classify the airplane as to its dynamical characteristics (NELSON; SMITH, 1989). In the UAV case, on the other hand, this fact is not as important as the global stability characteristics.

4.4 Application to Flight Control

Following closely what was proposed in (HARKEGARD, 2001b), we apply the theory of the so called backstepping controller design to the control of the aircraft's wind angles (α, β).

To achieve the objective of deriving a backstepping controller, first the aircraft dynamics will be considered in the light of some simplifying assumptions which will allow us to write the airplane dynamics in the *strict feedback form* (see section 4.3), then the globally stabilizing controller will be derived using the results of Lyapunov's direct method. Finally the performance of the obtained control law will be judged based on the simulation results and the simplifying assumptions made initially.

4.4.1 Simplified Aircraft Dynamics

We search to write the wind angle variables (α, β) dynamics in the *strict feedback form* starting from our airplane equations (2.13). To do that, lets first consider yet another referential, the so called **stability referential (S)**. This is the referential we get by rotating the body referential **B** an angle $-\alpha$ around the $\mathbf{y_B}$ axis. Looking again at figure 2.2, one can see that when the side slip angle is zero, the stability referential coincides with the wind reference frame.

So, considering the referential **(S)**, let us write the body angular velocity in these new coordinates. We will get:

$$\omega_s = R_{bs}\omega$$

or, expanding it:

$$\begin{bmatrix} p_s \\ q_s \\ r_s \end{bmatrix} = \begin{bmatrix} cos(\alpha) & 0 & sin(\alpha) \\ 0 & 1 & 0 \\ -sin(\alpha) & 0 & cos(\alpha) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{4.9}$$

Hypotheses Before substituting (4.9) in (2.13) we will establish a set of hypothesis that will simplify the aircrafts dynamics. Those are:

- The wind angles dynamics dependence on the control surface deflections, throttle change or angular rates will be ignored.
- Longitudinal and lateral control are not applied simultaneously, thus when α is controlled, β and p_s are regarded as constants and when β is controlled, α is regarded as constant.

The first assumption does indeed hold as a good approximation, since control surface deflections, at least in the case of conventional configuration aircrafts, do not influence the total forces as they do with the total moments.

The second assumption certainly does not represent well real flight situations, but simulations where considerable mixing between the control channels is present (section 4.5) show that the controller is still capable of achieving good performance.

Now, going back to (2.13) and replacing (4.9) in the expressions for the wind angles dynamics we get:

$$\dot{\alpha} = q_s - tan(\beta)p_s + \frac{1}{V\mathbf{C}_\beta}\left(\frac{-L_0}{m} + g_2\right) \tag{4.10a}$$

$$\dot{\beta} = -r_s + \frac{1}{V}\left(\frac{Y_0}{m} + g_3\right) \tag{4.10b}$$

$$\dot{\omega}_s = R_{bs}\left(f_\omega(X) + G_w U\right) \tag{4.10c}$$

With $X = [\alpha, \beta, p_s, q_s, r_s]$, $U = [\mathrm{d}p, \mathrm{d}a, \mathrm{d}r]$, function $f_\omega(X)$ and G_w are defined following (2.13) and (g_2, g_3) defined as in (2.14). Note then that, considering the second of our hypotheses, (4.10) can be decoupled and used to represent both α ((4.10a) and (4.10c)) and β ((4.10b) and (4.10c)) in the *strict feedback form.*

4.4.2 The Backstepping Controller for wind angles

From now on, we will work with the dynamics for each wind angle separately. Following what is done in (HARKEGARD, 2001b), we can write the second order systems for the wind angles in a general form as:

$$\begin{aligned} \dot{w}_1 &= f_1(w_1, y) + w_2 \\ \dot{w}_2 &= f_2(w_1, w_2) + G\delta \end{aligned} \tag{4.11}$$

Where the relations between the quantities in (4.11) and (4.10) are summarized in table 4.1. To calculate δ later on, we will need to invert G. As for the β system G is not a square matrix, because both the ailerons and the rudder deflections considerably affect

TABLE 4.1 – Relation between the general system in (4.12) and the systems in (4.10).

General System (4.12)	α dynamics (4.10a)	β dynamics (4.10b)
w_1	α	β
w_2	q_s	$-r_s$
y	$H, V, \beta, \phi, \theta, p_s$	$H, V, \alpha, \phi, \theta$
$f_1(w_1, y)$	$f_\alpha(\alpha, y_\alpha)$	$f_\beta(\beta, y_\beta)$
$f_2(w_1, w_2)$	$[R_{bs} f_\omega(X)]_2$	$[R_{bs} f_\omega(X)]_3$
G	$[R_{bs} G_w]_2$	$[[R_{bs} G_w]_1\ [R_{bs} G_w]_3]$
δ	$\mathrm{d}p$	$[\mathrm{d}a\ \mathrm{d}r]^T$

β dynamics, we will consider a more simple controller:

$$\begin{aligned} \dot{w}_1 &= f_1(w_1, y) + w_2 \\ \dot{w}_2 &= u \end{aligned} \tag{4.12}$$

So that when u is calculated both for α and β we can perform the inversion directly using (4.10c). The functions $f_\alpha(\alpha, y_\alpha)$ and $f_\beta(\beta, y_\beta)$ are defined as:

$$f_\alpha(\alpha, y_\alpha) = -tan(\beta) p_s + \frac{1}{V\mathbf{C}_\beta}\left(\frac{-L_0(\alpha)}{m} + g_2\right) \tag{4.13a}$$

$$f_\beta(\beta, y_\beta) = \frac{1}{V}\left(\frac{Y_0(\beta)}{m} + g_3\right) \tag{4.13b}$$

Our objective then is, using the control δ, drive the system (4.12) to a state like $[w_{1des} 0]^T$. As it is more comfortable to have the origin as the desired equilibrium point, we make a coordinate transformation in the variables $[w_1 w_2]^T$:

$$\begin{aligned} x_1 &= w_1 - w_{1des} \\ x_2 &= w_2 + f_1(w_{1des}, y) \end{aligned} \tag{4.14}$$

consider w_{1des} to be a constant value, we have then:

$$\dot{x_1} = \Phi(x_1) + x_2 \tag{4.15a}$$

$$\dot{x_2} = u + \nabla f_1(w_{1des}, y) \cdot \dot{y} \tag{4.15b}$$

where

$$\Phi(x_1) = f_1(x_1 + w_{1des}, y) - f_1(w_{1des}, y)$$

and thus $\Phi(0) = 0$

Now, analysing the second term on the right side of (4.15b) we see that it is given by:

$$\nabla f_1(w_{1des}, y) \cdot \dot{y} = \sum_{i=1}^{n} \frac{\partial f_1}{\partial y_i} y_i \tag{4.16}$$

that is, its influence will be measured both by the absolute value of the time derivative of each variable in y and by the absolute value of the partial derivative of f_1 with respect to each variable in y.

In the light of our second hypothesis, we already have that when analysing the α dynamics, $\dot{\beta} = 0$ and $\dot{p}_s = 0$ and when dealing with β dynamics, $\dot{\alpha} = 0$. For the control task dealt here, we can reasonably consider the velocity and density to vary slowly when compared to the output, so that also $\dot{V} = 0$ and $\dot{\rho} = 0$. Finally, note that, for the α dynamics:

$$\frac{\partial f_\alpha}{\partial \phi}\dot{\phi} = \frac{\partial g_2}{\partial \phi}\dot{\phi} = -g\mathbf{C}_\theta\mathbf{S}_\phi\mathbf{C}_\alpha\dot{\phi}$$
$$\frac{\partial f_\alpha}{\partial \theta}\dot{\theta} = \frac{\partial g_2}{\partial \theta}\dot{\phi} = g\left(-\mathbf{S}_\theta\mathbf{C}_\phi\mathbf{C}_\alpha + \mathbf{C}_\theta\mathbf{S}_\alpha\right)\dot{\theta}$$

so that in regular, levelled flight situations:

$$\frac{\partial g_2}{\partial \phi} \approx 0 \text{ and: } \frac{\partial g_2}{\partial \theta} \approx 0 \tag{4.17}$$

the same reasoning can be made for β with similar results, that is, for regular, levelled flight:

$$\frac{\partial g_3}{\partial \phi} \approx 0 \text{ and: } \frac{\partial g_3}{\partial \theta} \approx 0 \tag{4.18}$$

We are then motivated to make a last simplifying assumption to our model:

- The term $\nabla f_1(w_{1des}, y) \cdot \dot{y}$ in (4.15b) can be ignored in a first approximation.

Controller Design We are ready to design the controller. Starting from (4.15) we will first try to find a virtual control $x_{2des} = -\Psi(x_1)$ that leads x_1 to the origin in (4.15a). For that, let us choose the LCF:

$$U = \frac{1}{2}x_1^2$$

so that $\Psi(x_1)$ has to be chosen such as:

$$\dot{U} = x_1 \left(\Phi(x_1) - \Psi(x_1)\right) = -W(x_1) \leqslant 0 \tag{4.19}$$

where $W(x_1)$ is a positive semi-definite scalar field.

Now, supposing that we have $\Psi(x_1)$ we will try to find a control law for δ to drive x_2 to x_{2des}. Taking the error in x_2:

$$\tilde{x}_2 = x_2 - x_{2des} = x_2 + \Psi(x_1) \tag{4.20}$$

we can re-write the system (4.15) as:

$$\begin{aligned} \dot{x}_1 &= \Phi(x_1) - \Psi(x_1) + \tilde{x}_2 \\ \dot{\tilde{x}}_2 &= u + \Psi'(x_1)\left(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2\right) \end{aligned} \tag{4.21}$$

For the new system, we chose a LCF like (see (HARKEGARD, 2001b) for details):

$$V(x_1, \tilde{x}_2) = F(x_1) + \frac{1}{2}\tilde{x}_2^2 \tag{4.22}$$

with $F(x_1)$ a positive definite scalar field and a valid Lyapunov function to the first system, that is:

$$\dot{F}(x_1)|_{x_2 = x_{2des}} = F'(x_1)\left(\Phi(x_1) - \Psi(x_1)\right) = -W(x_1) \tag{4.23}$$

with $W(x_1)$ positive definite.

Derivating (4.22) with respect to time we have:

$$\begin{aligned}\dot{V} &= F'(x_1)\left(\Phi(x_1) - \Psi(x_1)\right) + \\ &\quad + \tilde{x}_2\left(F'(x_1) + u + \Psi'(x_1)(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2)\right) \\ &= -W(x_1) + \tilde{x}_2(u + \Psi'(x_1)\tilde{x}_2 + \\ &\quad + F'(x_1) + \Psi'(x_1)(\Phi(x_1) - \Psi(x_1)))\end{aligned} \tag{4.24}$$

it would be smart to choose $F'(x_1)$ such as to cancel the terms in x_1 multiplying $\tilde{x}_2$ in (4.24), that is:

$$F'(x_1) = -\Psi'(x_1)\left(\Phi(x_1) - \Psi(x_1)\right) \tag{4.25}$$

substituting (4.25) in (4.23) we have that:

$$W(x_1) = \Psi'(x_1)\left(\Phi(x_1) - \Psi(x_1)\right)^2$$

as $W(x_1)$ is positive definite and, by (4.19), we have that $\Phi(x_1) \neq \Psi(x_1)$ except when $x_1 = 0$, our function $\Psi(x_1)$ has to satisfy:

$$\Psi'(x_1) > 0 \tag{4.26}$$

This means that our choice of $\Psi(x_1)$ is subjected to the conditions in (4.19) and (4.26). Moreover, dividing both sides of (4.19) by x_1^2 we obtain:

$$\frac{\Phi(x_1)}{x_1} < \frac{\Psi(x_1)}{x_1} \leqslant \Psi'(x_1)$$

That is, when choosing the virtual control law for the system in x_1, the information we need about the non-linearity $\Phi(x_1)$ is an upper bound in its derivative, or more specifically, an upper bound in $\frac{\Phi(x_1)}{x_1}$.

Now, returning to our choice of F in (4.25), the derivative of our LCF becomes:

$$\dot{V} = -W(x_1) + \tilde{x}_2 \left(u + \Psi'(x_1)\tilde{x}_2\right) \tag{4.27}$$

so that, if we choose $\Psi(x_1)$ such that $\Psi'(x_1)$ has an upper bound, that is, it exists a positive constant k satisfying $k > \Psi'(x_1)$, we can define the following control law for u:

$$u = -k\tilde{x}_2 = -k(x_2 + \Psi(x_1)) \tag{4.28}$$

so that substituting (4.28) in (4.27) we see that $\dot{V}$ is negative semi-definite and by the corollary of Barballat's lemma that this control law makes the origin of (4.15) GAS.

Partial Summary We can summarize the results obtained so far as follows (HARKEGARD, 2001b):

Consider the system:

$$\dot{x_1} = \Phi(x_1) + x_2$$

$$\dot{x_2} = u$$

Where $\Phi(0) = 0$. Suppose that it exists a constant κ such that:

$$\frac{\Phi(x_1)}{x_1} < \kappa \; \forall x \neq 0 \tag{4.29}$$

Then a globally stabilizing control law:

$$u = -k(x_2 + \Psi(x_1))$$

can be found where:

$$x_1 \left(\Phi(x_1) - \Psi(x_1)\right) < 0 \; \forall x \neq 0$$

and

$$0 < \Psi'(x_1) < k$$

Moreover, a Lyapunov function for this system is given by:

$$V(x_1, x_2) = \int_0^{x_1} -\Psi'(y)\,(\Phi(y) - \Psi(y))\,dy + \frac{1}{2}(x_2 + \Psi(x_1))^2$$

which satisfies:

$$\dot{V} = -\Psi'(x_1)\,(\Phi(x_1) - \Psi(x_1))^2 - (k - \Psi'(x_1))(x_2 + \Psi(x_1))^2$$

□

Note that the only information we need to now about the non linearity is (4.29).

Airplane Variables We now have to replace the airplane variables in the control law (4.28) with the help of table 4.1 and equations (4.13). Also, we have to choose a function $x_{2des} = \Psi(x_1)$ for each case. In order to keep the laws as simple as possible and also to compare the resulting controller with a classical PID, we will choose $\Psi(x_1)$ as a linear function.

Then the controlling law for α will be:

$$u_\alpha = -k_{\alpha,2}\,(q_s + f_\alpha(\alpha_{ref}, y_\alpha) + k_{\alpha,1}(\alpha - \alpha_{ref})) \tag{4.30}$$

Where the parameters $k_{\alpha,1}$ and $k_{\alpha,2}$ should satisfy ((HARKEGARD, 2001b)):

$$k_{\alpha,2} > k_{\alpha,1} \text{ and: } k_{\alpha,1} > \max 0, \kappa_\alpha$$

where:

$$\kappa_\alpha = \max_{\alpha, y_\alpha} \frac{\partial f_\alpha(\alpha, y_\alpha)}{\partial \alpha}$$

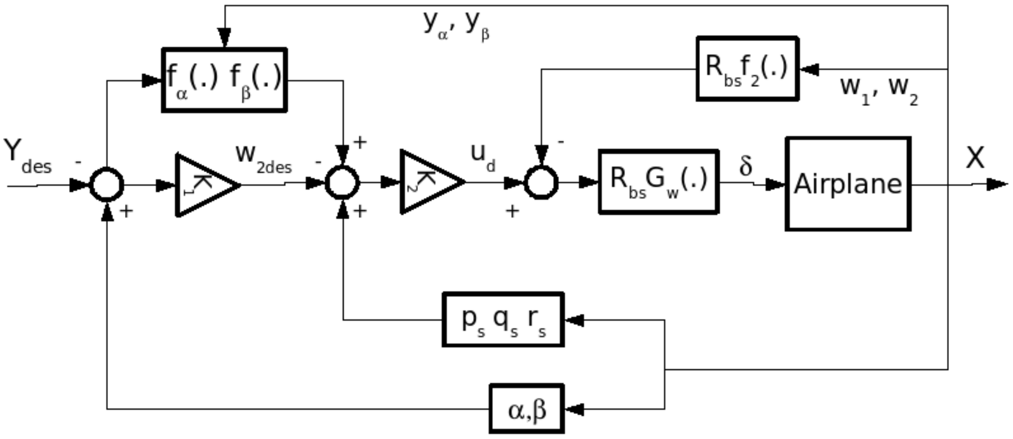


FIGURE 4.3 – Block Diagram for the Complete Backstepping controller.

The controlling law for β, considering $\beta_{ref} = 0$ will be:

$$u_\beta = -k_{\beta,2}\left(-r_s + f_\beta(0, y_\beta) + k_{\beta,1}(\beta)\right) \tag{4.31}$$

Where the parameters $k_{\beta,1}$ and $k_{\beta,2}$ should satisfy ((HARKEGARD, 2001b)):

$$k_{\beta,2} > k_{\beta,1} \text{ and: } k_{\beta,1} > \max 0, \kappa_\beta$$

$$\text{where:}$$

$$\kappa_\beta = \max_{\beta, y_\beta} \frac{f_\beta(\beta, y_\beta) - f_\beta(0, y_\beta)}{\beta}$$

To complete the control laws, we can also control the roll rate in the stability axis (with a simple proportional law):

$$u_{p_s} = -k_{p_s}\left(p_s - p_{sref}\right) \tag{4.32}$$

Finally, we can substitute the vector $[u_{p_s}, u_\alpha, u_\beta]$ in the place of $\dot{\omega}_s$ in (4.10c) and invert it to obtain the control surface deflections. Note then that this last part is nothing more then a simple first order dynamic inversion controller, so that our complete backstepping control scheme can be summarized in the block diagram of figure 4.3.

Note that we still depend on the knowledge of the functions $f_\alpha(\alpha_{ref}, y_\alpha)$ and $f_\beta(0, y_\beta)$.

TABLE 4.2 – Parameters for backstepping controller.

p_s Control	α Control	β Control
$k_{p_s} = 2.5$	$k_{\alpha,1} = 1$ $k_{\alpha,2} = 3$	$k_{\beta,1} = 1.5$ $k_{\beta,2} = 4$

On the other hand, those functions are outside the feedback control loop, acting as pre-filters, thus only translating the equilibrium point (as seen by the results in section 4.5). In other words, even in the presence of modelling errors in f_α and f_β, the proposed control laws will still guarantee stability.

Feeding forward this nonlinear functions in the controller brings the consequence of not having the same resulting dynamics for all flight situations. It becomes then necessary tuning the gains ($k_{\alpha,1}$,$k_{\beta,1}$,...,k_{p_s}). One way of doing that is to choose a desired flight condition, and obtain the gains following the classical methods.

Compare the block diagram in figure 4.3 with the one in figure 3.2. Notice that the difference between the two is in what comes before the control mixing, that is, before inverting matrix G to calculate control deflections. In the feedback linearization case, as a complete inversion of the system is supposed, the gains become simply the values which will represent the desired dynamics for the output variables. Meanwhile, the backstepping technique leaves some of the aircraft dynamics “untouched” (f_α and f_β) whilst still using the equivalent of proportional and derivative gains to guarantee controller convergence.

4.5 Results

In table 4.2 we show the parameters used for each of the laws in (4.30) to (4.32).

4.6 Commentaries

The results in figure 4.4 shows the response to a step in the angle of attack. Note that the controller present the expected behaviour for the case of bad modelling of the lift force (which is the term that dominates $f_\alpha(.)$). It is seen that, although more control

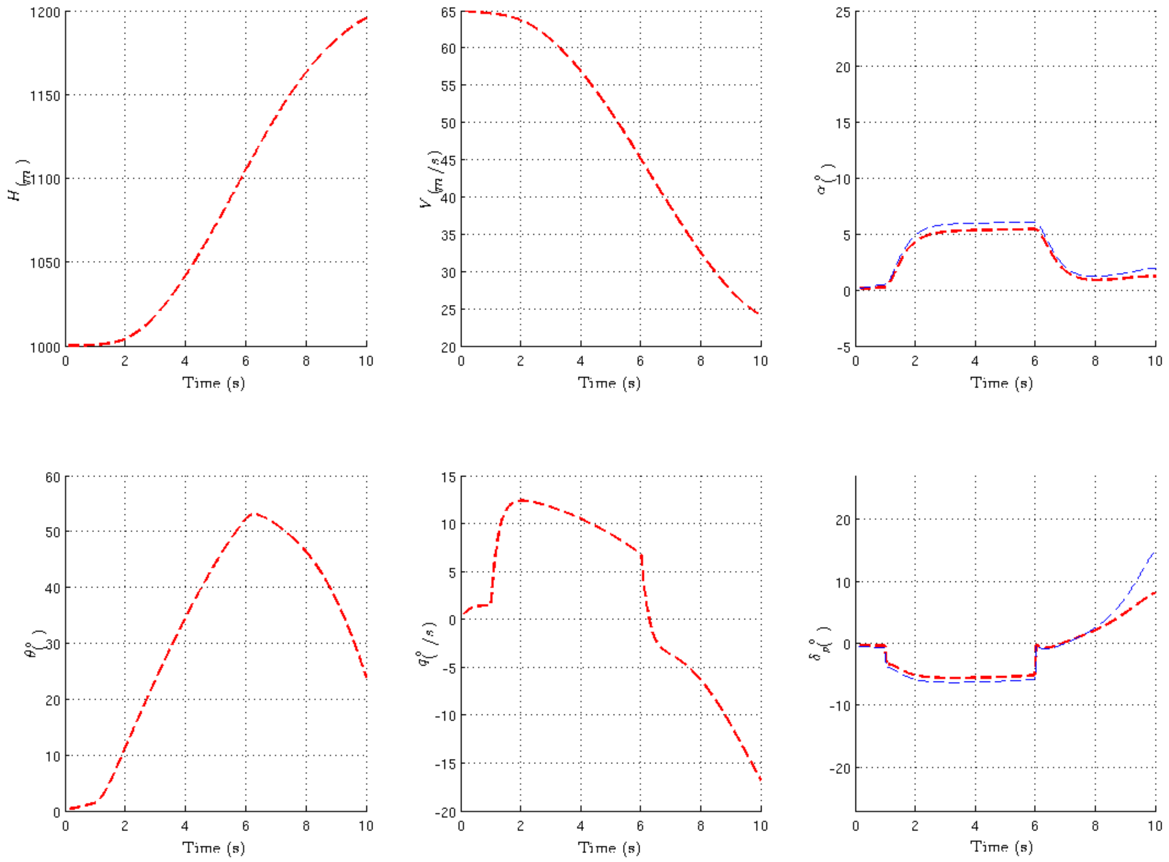


FIGURE 4.4 – Aircraft states and controller response to step of 5° in α (red). In comparison, α response and control deflection for controller with 10% of error in lift estimation (blue).

effort is utilized, the output is only translated from the ideal behaviour. Moreover, even for this case of reasonable error in model estimation, the behaviour does not differ much of the nominal behaviour, given a measure of the robustness of this method.

Figures 4.5 and 4.6 show then the nominal response for the case of a step command in the stability roll rate and then a mixed command of a step in α and a step in the roll rate respectively. Notice in 4.6 the resulting trajectory for α. See that near the fourth second of simulation there is a perturbation in the desired value. Going back to our last assumption of section 4.4.2, it is remembered that the controller was designed thinking of rather "normal" flight situations, which is not the case after three seconds of fixed roll rate command. In other words, the model states passes in a region of the domain where the approximations (4.17) and (4.18) are no longer valid.

Thus, as disadvantages of the backstepping technique we can cite: not being able

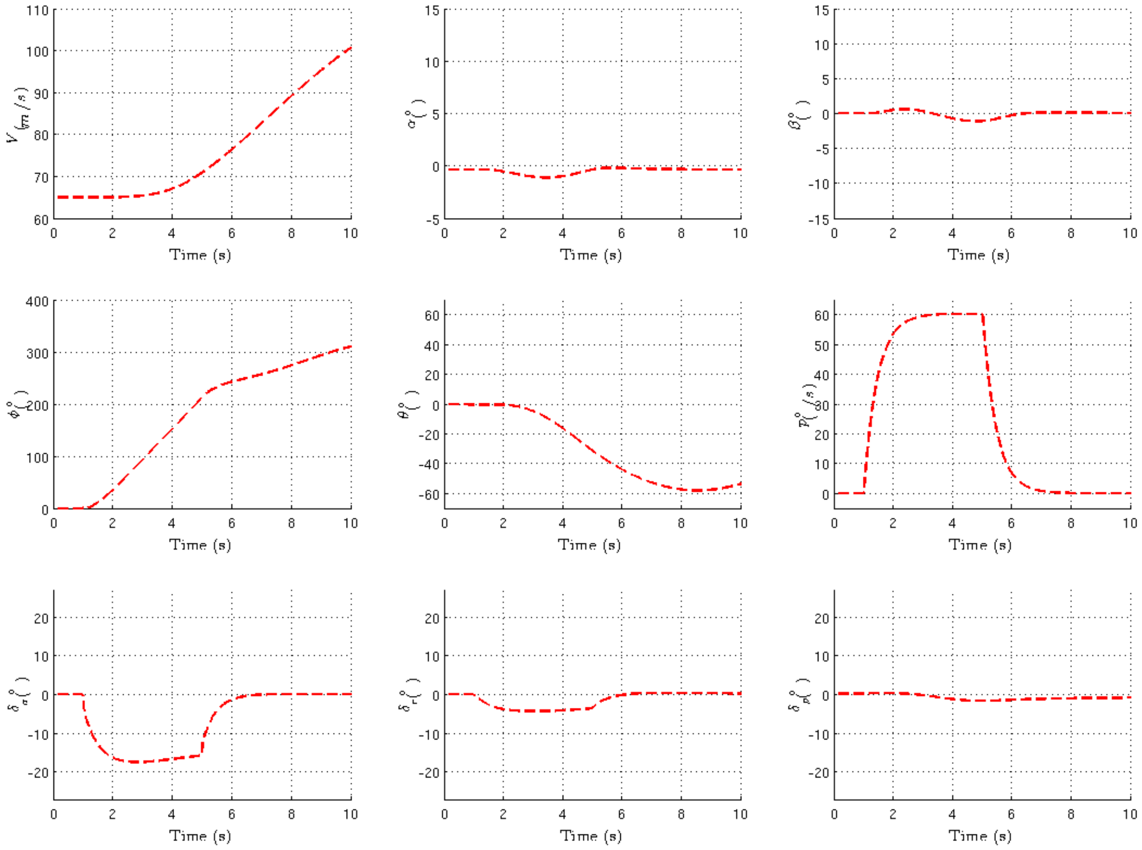


FIGURE 4.5 – Aircraft states and controller response to step command of 60°$/s$ in p_s.

to easily predict the resulting dynamic characteristics of the closed loop system; also not being able to design a controller that will guarantee the same dynamic behaviour throughout the flight envelope. Both of those are direct consequences of leaving the plant's nonlinear terms that do not destabilize it untouched.

As a general conclusion, however, it is seen that the backstepping technique shows promising results for applications in high performance aircrafts and UAV's. By keeping some of the inherently stable nonlinear characteristics of the aircraft, it is possible to design simple controllers laws (notice that laws (4.30), (4.31) and (4.32) are in fact linear control laws with some nonlinear correction terms) that still guarantee global convergence of the system.

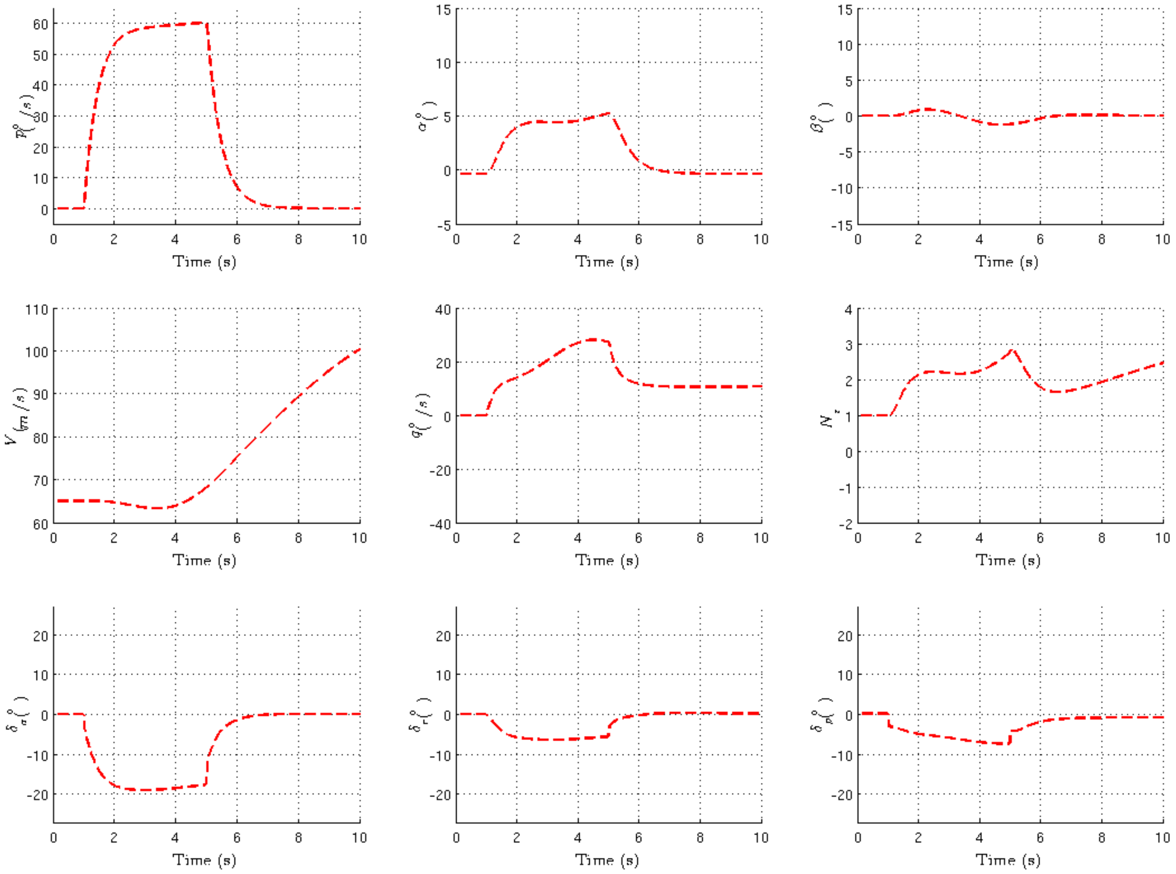


FIGURE 4.6 – Aircraft states and controller response to step command both in α (5°) and in p_s ($60^\circ/s$).

5 Adaptive Control

5.1 Introduction

Models of the airplane dynamics are not exact. There are approximations done in the analysis that may sometimes only work in certain flight regimes, while others depend on parameters that are very hard to measure or calculate precisely. As showed above, however, feedback linearization, and in a less extent, backstepping, depend on the knowledge of the systems dynamics to be accurately and safely applied. One way to solve this problem is to use Adaptive control schemes (NAM; ARAPOSTATHIS, 1988; SLOTINE; LI, 1991; SINGH; STEINBERG, 1996).

Nowadays, it can be noted the progressive evolution and application of UAVs (Unmanned Aircraft Vehicles) and MAVs (Micro Aerial Vehicles). The dynamic equations parameters for those types of aircraft are normally not known with the same certainty as for commercial airplanes, also, they are subjected to sudden parameter change (as dropping a load or breaking a fragile command surface). Adaptive controllers then promise to be very useful in those cases.

Adaptive controllers present a relatively recent research subject (NAM; ARAPOSTATHIS, 1988; SLOTINE; LI, 1991; KRSTIĆ; KANELLAKOPOULOS; KOKOTOVIĆ, 1992), and most part of the theory was already developed since the beginning of the last decade. Several applications to flight control have been also studied (MONAHEMI; KRISTIĆ, 1996; SINGH; STEINBERG, 1996; STEINBERG; PAGE, 1998) until the end of the same decade.

5.2 Concepts of Adaptive Controllers

The concept of an Adaptive controller is to use the sensor data (both of input as of output) to “check” at each reading of the system, whether the model parameters used are indeed correct, and if possible, in what direction should they be changed to be coherent with the real data that is being acquired. In the case of nonlinear control, it is very hard to find a general approach to the problem of model adaptation, and the only solutions proposed are based in parameterized models, that is, nonlinear models that are weighted sums of known nonlinear functions:

$$F(X) = \sum_{i=1}^{n} a_i f_i(X) \tag{5.1}$$

From the aircraft control point of view, that should pose no problems, once the equations of motion can be written in such a way, as will be seen in chapter 2.

There are basically two concepts of adaptive controllers, the model reference adaptive controller (MRAC) and the self tuning adaptive controller (STC).

5.2.1 The MRAC

A scheme of an MRAC is showed in figure 5.1. Let $\hat{a}_i$ be the estimated parameters. What the MRAC does is to take a generic model response to the entry signal and use it as a reference of what the plant was supposed to be doing. As it can be seen, the reference model generates no more then the trajectory we wished to be following. From the error between the real trajectory and the reference trajectory, the parameters are adapted according to an adequate law an updated in the controller, which will in turn give better inputs to the system that will eventually drive the plant to follow the reference trajectory.

The analysis and design procedure of a MRAC for nonlinear systems has to be done with the use of Lyapunov techniques (NAM; ARAPOSTATHIS, 1988; SINGH; STEINBERG, 1996). While there are many results in this field of study, lets only take a brief

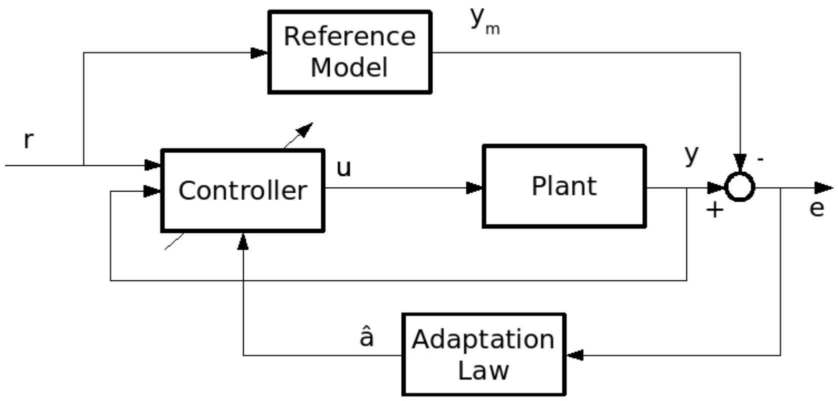


FIGURE 5.1 – The Model Reference Adaptive Controller

view of the process we are going to use later on with the aircraft.

Consider first a SISO system. The first step is to create a generalized error function from the error vector. Lets call the generalized error the variable s. This function is normally a linear combination of the error and its derivatives, so that when the error stabilizes at zero the generalized error is also stable at zero and *vice-versa*, for example:

$$s = \sum_{i=1}^{n-1} k_i e^{(i)}$$

Where: k_i would make the polynomial $\mu(\lambda) = \lambda^r + \sum_{i=0}^{r-1} k_i \lambda^i$ Hurwitz.

Then, using the known parameterized dynamics of the model, we write the s dynamics as a function of the real parameters and our control input. As we do not know the value of the real parameters, we are obliged to use a control law dependent of the estimated parameters and the measured state:

$$u = u(X, \hat{a}_i)$$

Finally, a Lyapunov function candidate is proposed for the augmented system in the variables $[s, \hat{a}_i]$ and the control u as well as the adaptation laws $\dot{\hat{a}}_i$ are chosen in such a fashion to render the Lyapunov function candidate time derivative a negative semi-definite function. According to Barballat's lemma, this would guarantee the convergence of the system. There are still various problems of implementation and convergence to be discussed and so the method is better furthered explained with an example.

MRAC Example Lets consider the motion of a mass subjected to the effect of gravity and a vertical force controlling it. The most general way of writing such a system is:

$$\ddot{z} = -K_1 + K_2 u \tag{5.2}$$

Where z is the hight of the mass and u the input value. Here, we are supposing we don't know the value of the mass and the characteristics of our actuator, so that K_2 is an unknown parameter. K_1 is the value of gravity, but for the example, lets also suppose we don't know this value either. We wish to control the mass so it will behave like the following model:

$$\ddot{z_m} = -10\dot{z_m} - 25(z_m - r(t)) \tag{5.3}$$

Where $r(t)$ is a reference position desired value. Had we known the values of K_1 and K_2, it would suffice to use the control law:

$$u(\dot{z}, z) = \frac{1}{K_2}\left(K_1 - 10\dot{z} - 25(z - r(t))\right)$$

To achieve the desired behavior. But instead, we will be working with the estimated parameters $\hat{K}_1$ and $\hat{K}_2$ so that our real control will have to be a function of these two parameters also:

$$u = u(\dot{z}, z, \hat{K}_1, \hat{K}_2) \tag{5.4}$$

Of course, we are also supposing that we can measure the full state: $[\dot{z}, z]$.

Now the error between the model position and the real plant position is given by:

$e \equiv z - z_m$. As our model is of the second order, we take the generalized error to be

$$\begin{gathered} s \equiv \dot{e} + \lambda_1 e + \lambda_2 x_s \\ \text{Where: } \dot{x_s} = e \end{gathered} \tag{5.5}$$

Note the integral term x_s in (5.5). This idea, taken from (SINGH; STEINBERG, 1996), is to guarantee better a trajectory tracking of the model. By taking the derivative of the generalized error we have

$$\dot{s} = \ddot{z} - \ddot{z_m} + \lambda_1 \dot{e} + \lambda_2 e \tag{5.6}$$

substituting (5.2) in (5.6) we have:

$$\dot{s} = -K_1 + K_2 u - \ddot{z_m} + \lambda_1 \dot{e} + \lambda_2 e \tag{5.7a}$$

$$\text{Defining: } \nu = -\ddot{z_m} + \lambda_1 \dot{e} + \lambda_2 e \tag{5.7b}$$

$$\text{We get: } \dot{s} = -K_1 + K_2 u + \nu \tag{5.7c}$$

Considering (5.4) and (5.7c) we have a dynamical system in the variables $[s, \hat{K}_1, \hat{K}_2]$. The advantage now is that we have control over the dynamics of the estimated parameters, which are exactly the adaptation laws we will be choosing. That is, we can define the functions:

$$\dot{\hat{K}}_1 = \kappa_1(s, \hat{K}_1, \hat{K}_2) \tag{5.8}$$

$$\dot{\hat{K}}_2 = \kappa_2(s, \hat{K}_1, \hat{K}_2) \tag{5.9}$$

We want to define these functions in such a way that our plant will follow the desired trajectory and the parameters will converge to their real values. This motivates us to propose the following Lyapunov functions candidates for the augmented system:

$$V = \frac{1}{2}s^2 + \gamma_1 \frac{1}{2}\tilde{K}_1^2 + \gamma_2 \frac{1}{2}\tilde{K}_2^2 \tag{5.10}$$

Where γ_1,γ_2 are strictly positive constants and $\tilde{K}_1^2$,$\tilde{K}_2^2$ are the errors in the parameters estimations, that is:

$$\tilde{K}_1^2 \equiv K_1 - \hat{K}_1 \tag{5.11}$$

$$\tilde{K}_2^2 \equiv K_2 - \hat{K}_2 \tag{5.12}$$

Derivating (5.10) with respect to time we have:

$$\dot{V} = s(-K_1 + K_2 u + \nu) + \gamma_1 \tilde{K}_1 \dot{\tilde{K}}_1 + \gamma_2 \tilde{K}_2 \dot{\tilde{K}}_2 \tag{5.13}$$

Finally, choosing the following laws for the control and adaptation:

$$u = \frac{1}{\hat{K}_2}\left(-c_1 s + \hat{K}_1 - \nu\right) \tag{5.14a}$$

$$\dot{\tilde{K}}_1 = \frac{1}{\gamma_1} s \tag{5.14b}$$

$$\dot{\tilde{K}}_2 = -\frac{1}{\gamma_2} s u \tag{5.14c}$$

With c_1 a strictly positive constant, we would have as a result for (5.13):

$$\dot{V} = -c_1 s^2 \leqslant 0$$

And so, from the Barballat's lemma corollary, the whole system in $[s, \tilde{K}_1, \tilde{K}_2]$ would converge to the origin.

Simulation In figure 5.2 are presented the results for a simulation of (5.2) with the results in (5.14). The parameters used for the simulation were $\gamma_1 = \gamma_2 = 3$, $\lambda_1 = \lambda_2 = 2$ and c_1, with initial values of the estimation parameters as $\hat{K}_1(0) = 5$ and $\hat{K}_2(0) = 30$ the reference model is the one described in (5.3) and the reference signal was a combination of sinusoids and steps. Although the trajectory is capable of following the model after some time, it is clearly seen that the parameters do not converge to the desired values (represented as black lines), instead, we see the estimated values oscillating in time. Note that by inserting the parameters estimates and their dynamics we are increasing the order,

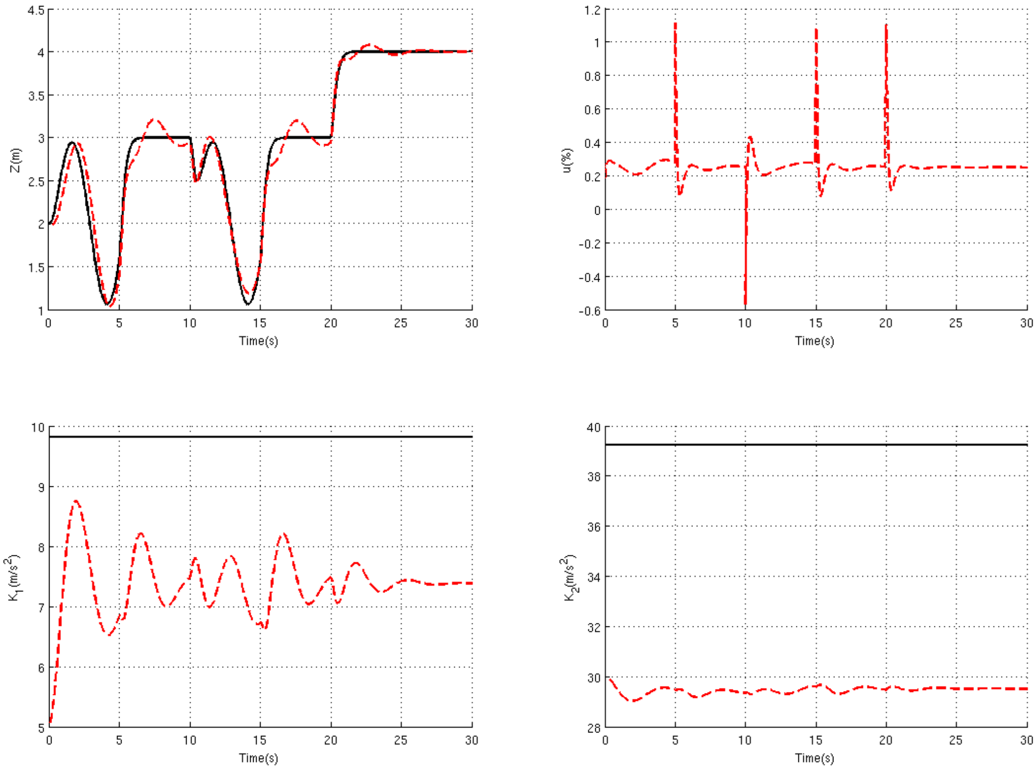


FIGURE 5.2 – Results for the complete adaptive controller. Reference signal $r(t) = 2 + sin(2\pi t/5)$ for $0 \leq t \leq 5$ and $10 \leq t \leq 15$. Step commands for the other time intervals.

and thus the complexity of the system. What the result shows is hence that there are many different equilibrium points for the system $[s, \tilde{K}_1, \tilde{K}_2]$ which are not necessarily the origin (SLOTINE; LI, 1991). One way to avoid this is to take a more complex excitation, with different frequencies, which will oblige the controller to converge to the real parameters. In figure 5.3 we see the same adaptive controller as used for the previous results, but with a reference signal with more frequencies attached. Note that the values of the estimated parameter now show the tendency of convergence towards the real values.

It can also be noted that if one of the parameters is already know *a priori* the proposed adaptive controller is able to converge the unknown parameter estimate to the real value, as seen in figure 5.4.

In the cases showed above, the complexity of the total adaptive controller was smaller and thus also the complexity of the resulting dynamical system, thus only the real values are the ones that can make the system follow the desired model correctly.

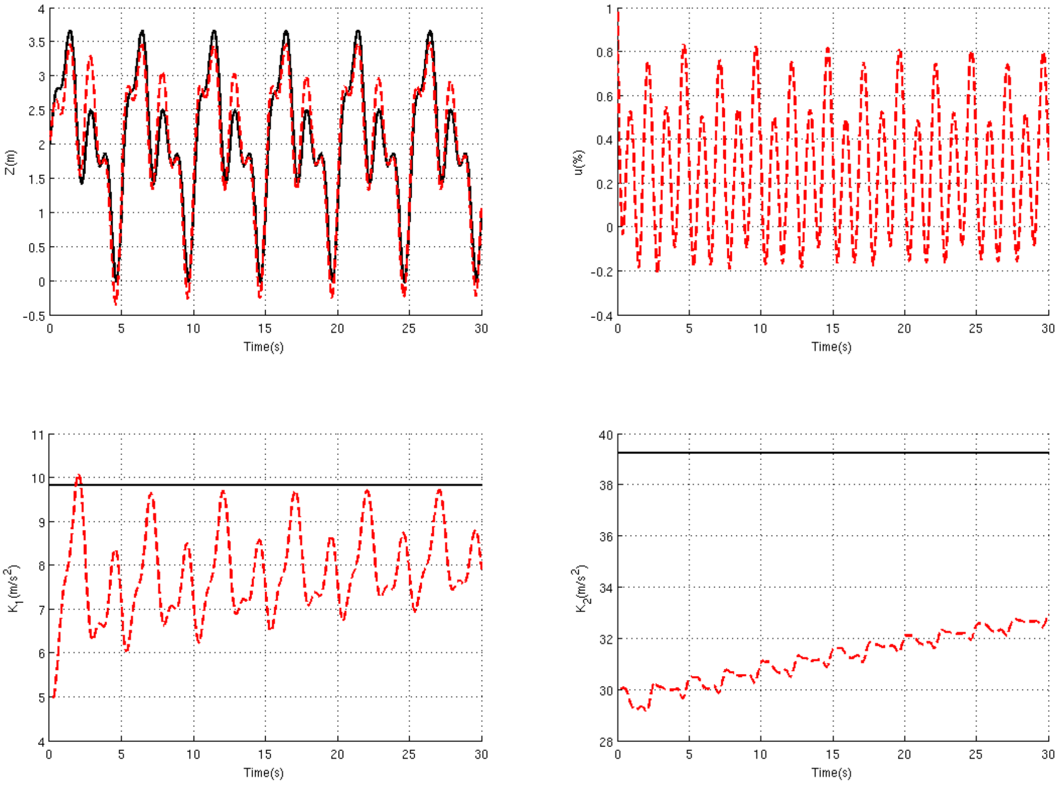
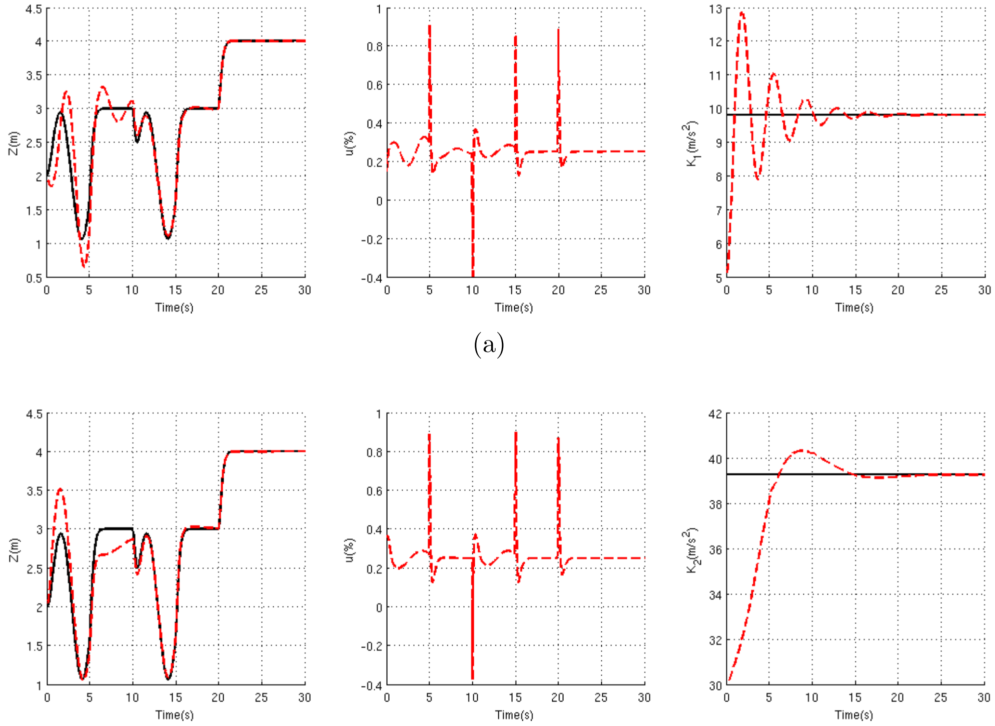


FIGURE 5.3 – Results for the complete adaptive controller. Reference signal $r(t) = 2 + \sin 2\pi t/5 + \sin 4\pi t/5 + \cos 8\pi t/5$

Finally, we point out that control saturation, both in amplitude as well as speed was not addressed in this simple example. This can be dealt with since we have the liberty to choose the values of the constants which will define our adaptive control laws. This subject will be better discussed further on in this chapter.

5.2.2 The STC

The *self tunning controller (STC)* as described by figure 5.5 does not rely on a reference model to estimate the plant's parameters. What it does is to take the output and input measurement signals and try to fit the data by adapting the parameters estimates. These parameters estimates are then passed to the controller and used to calculate the input for the plant to follow a desired trajectory. This process is then recursively repeated in time so that, if the parameters are constant or slowly varying, the total controller scheme



(a)

(b)

FIGURE 5.4 – Results for the same adaptive controller as in figure 5.2 but with *a priori* knowledge of one of the parameters. 5.4(a) Only the K_1 adaptation. 5.4(b) Only the K_2 adaptation.

will be able to continuously estimate the real plant parameters (in the case of a discrete controller based on time sample analysis, of course this will depend on a high sampling rate).

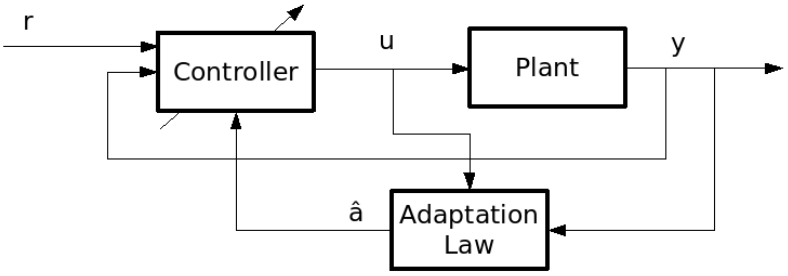


FIGURE 5.5 – The Model Reference Adaptive Controller

There are many methods available for doing these parameters estimations at each time step and they may or may not contain physical information of the plant that one wishes to control. For linear plants, normally the least-square method or its variations are used (SLOTINE; LI, 1991). These methods can normally be extended to nonlinear plants as well although the stability analysis of a STC controller is more difficult then for the analysis of a MRAC controller.

One important aspect of the STC controller is that it only “fits” the data received, and this data doesn’t have to be exactly the vectors y and u for example, but filtered and treated versions of it y_f and u_f, so that a linear relation between those last two can be inferred. Finally, easier methods can be used so that parameter convergence and tracking can be guaranteed when using y_f and u_f as inputs for the parameter estimator.

A great set of results are available in the literature over the subject of plant parameter estimation from the knowledge of input-output data, and as it will not be used in this work, this section is just to present this type of controller as an option for Adaptive control design.

5.2.3 Summary of the theory

Adaptive control can be very useful for parameter estimation of nonlinear plants, since the nonlinear controller normally demand a good knowledge of the model to operate adequately. However, the stability analysis of such schemes normally are complex to do, since they augment the complexity of the dynamical system by introducing more parameters.

There are two principal types of adaptive controllers, namely the *model reference adaptive controller* (MRAC) and the *self tuning controller* (STC), which arise from different perspectives (SLOTINE; LI, 1991). In the MRAC scheme the parameters are adapted in such a way to guarantee that the plant will follow a reference dynamical model, while the STC only takes the signal measurements from the input and output of the plant and tries to “fit” this data by constantly updating the estimated parameters. Since this work will only work with the MRAC implementation, the STC was only briefly presented.

5.3 Applications to Flight Control

It was already presented that for the feedback linearization scheme to be implemented, a complete model of the aircraft dynamics needs to be present. In chapter 3 we have already tried to observe the possible effects of modeling errors to controller response, by using aircraft parameters in the controller that were different then the parameters in the simulation model. Also, in the backstepping controller discussed in chapter 4, the last control step (see section 4.4.2 for details) consists of calculating the surface deflections by inverting the angular rate dynamics of the airplane system, which implies good knowledge of the variables involved for the controller to have the expected response.

In this chapter it will be applied a procedure very similar to what was developed in (SINGH; STEINBERG, 1996) (work from which we will also borrow most of the notation) to develop a model reference adaptive controller to the euler angles control of a three command surface aircraft.

5.4 MRAC Scheme for Aircraft Fast Dynamics Control

We start with equation (3.13), which we repeat here as equation (5.15):

$$\ddot{Y} = \begin{bmatrix} \nabla f_1 \cdot f(X) \\ \nabla f_2 \cdot f(X) \\ \nabla f_3 \cdot f(X) \end{bmatrix} + E(X)U \equiv \Psi(X) + E(X)U \tag{5.15}$$

$$\text{Where: } E_{i,j} = \nabla f_i \cdot g^{<j>}$$

Now, considering our aerodynamic modeling (section 2.1.2), it is seen that we can separate function $\Psi(X)$ in (5.15) in two parts, as:

$$\Psi(X) = \Psi_0(X) + \Psi_1(X)\mathrm{d_e} \tag{5.16}$$

Where the term $\mathrm{d_e}$ represents the aerodynamic coefficients defined as in (2.3). Now, the dimension of the output Y in (5.15) is $m = 3$ and according to our definition of $\mathrm{d_e}$ we have $dim(\mathrm{d_e}) = 6$, what means that $\Psi_1(X)$ is a $3x6$ matrix. Analogously, the matrix $E(X)$ in (5.15) is not just a function of X but also of some aerodynamic coefficients. Thus, collecting all these last coefficients in a vector $\mathrm{d_c}$ we have that, in reality:

$$E(X) = E(X, \mathrm{d_c})$$

In the aerodynamic notation, the vector $\mathrm{d_e}$ is composed of *stability derivatives* and $\mathrm{d_c}$ is composed of *control derivatives* [1]. These are the variables we will suppose not to know precisely. Moreover, note that the "real" quantities related to those values are supposed to keep constant or vary slowly during flight.

Note that by linearly writing the output dynamics in both the stability and control derivatives, we are working with the case described by (5.1). So that finally we can represent the output dynamics as:

$$\ddot{Y} = \Psi_0(X) + \Psi_1(X)\mathrm{d_e} + E(X, \mathrm{d_c})U \tag{5.17}$$

If we precisely knew the vectors $\mathrm{d_e}$ and $\mathrm{d_c}$ we could simply invert (5.17). But as we suppose only to have ***estimates*** $\hat{\mathrm{d}}_\mathrm{e}$ and $\hat{\mathrm{d}}_\mathrm{c}$ of the real values. Our objective will be then to find *adaptation laws* for those estimates based on the measurements of the states and of the commands that are made during flight.

Generalized Error To derive these adaptation laws, we start by defining the *generalized error* (SINGH; STEINBERG, 1996). First, we consider a model reference trajectory

[1] Specifically:
$\mathrm{d_e} = [C_{\mathcal{L}p}C_{\mathcal{L}r}C_{\mathcal{M}0}C_{\mathcal{M}q}C_{\mathcal{N}p}C_{\mathcal{N}r}]$;
$\mathrm{d_c} = [C_{\mathcal{L}\mathrm{d}a C_{\mathcal{L}\mathrm{d}r} C_{\mathcal{M}\mathrm{d}p} C_{\mathcal{N}\mathrm{d}a} C_{\mathcal{N}\mathrm{d}r}}]$

y_{ir} for each output Y generated by:

$$\ddot{y}_{ir} + c_{i1}\dot{y}_{ir} + c_{i0}(y_{ir} - y_{ides}(t) = 0 \quad |i = 1, 2, 3 \tag{5.18}$$

where $y_{ides}(t)$ is the instantaneous desired value for the output y_i.

Then, we will define the generalized error s to be such that:

$$\begin{gathered} s_i \equiv \dot{e}_i + k_{i1}e_i + k_{i0}x_{s_i} \quad |i = 1, 2, 3 \\ \text{where: } e_i = y_i - y_{ir} \\ \dot{x}_{s_i} = e_i \end{gathered} \tag{5.19}$$

That is, x_{s_i} is the integral of the error in the variable y_i. The constants k_{ij} are defined in such a way that the polynomials:

$$\mu_i(\lambda) = \lambda^2 + k_{i1}\lambda + k_{i0}$$

are Hurwitz, so that when each error (and its integral) goes to zero, so it does s_i. Conversely, if s is the origin we can say that our output is perfectly following the generated desired trajectory. Finally, observe that s is composed of quantities we can measure at each control step. Derivating s we have:

$$\begin{aligned} \dot{s} &= \ddot{Y} - \ddot{Y}_r + \sum_{j=0}^{1} K_j(y^{(j)} - y_c^{(j)}) \\ &= \Psi_1(X)\mathrm{d_e} + E(X, \mathrm{d_c})U + \nu(X, Y_r) \\ &\text{where: } \nu(X, Y_r) \equiv \Psi_0(X) - \ddot{Y}_r + \sum_{j=0}^{1} K_j(y^{(j)} - y_c^{(j)}) \end{aligned} \tag{5.20}$$

Controller Design Our objective is to design a controller that will both lead the output to the desired trajectory and converge the estimated parameters to their real values.

Defining then the estimation errors:

$$
\begin{aligned}
\tilde{\mathrm{d}}_{\mathrm{e}} &\equiv \mathrm{d}_{\mathrm{e}} - \hat{\mathrm{d}}_{\mathrm{e}} \\
\tilde{\mathrm{d}}_{\mathrm{c}} &\equiv \mathrm{d}_{\mathrm{c}} - \hat{\mathrm{d}}_{\mathrm{c}}
\end{aligned}
\tag{5.21}
$$

From the definitions made above, it is possible to consider the following augmented dynamical system in $[s, \tilde{\mathrm{d}}_{\mathrm{e}}, \tilde{\mathrm{d}}_{\mathrm{c}}]$:

$$\dot{s} = \Psi_1(X)\mathrm{d}_{\mathrm{e}} + E(X, \mathrm{d}_{\mathrm{c}})U + \nu(X, Y_r) \tag{5.22a}$$

$$\dot{\tilde{\mathrm{d}}}_{\mathrm{e}} = -\dot{\hat{\mathrm{d}}}_{\mathrm{e}} \tag{5.22b}$$

$$\dot{\tilde{\mathrm{d}}}_{\mathrm{c}} = -\dot{\hat{\mathrm{d}}}_{\mathrm{c}} \tag{5.22c}$$

considering the supposition that the real plant parameters are constant or slowly varying. The expressions $\dot{\hat{\mathrm{d}}}_{\mathrm{e}}$ and $\dot{\hat{\mathrm{d}}}_{\mathrm{c}}$ are precisely the adaptation laws whose definitions are the controller designer choice.

We can then propose the LFC for the augmented system :

$$U = \frac{1}{2}\left(s^T s + \tilde{\mathrm{d}}_{\mathrm{e}}^T P_1 \tilde{\mathrm{d}}_{\mathrm{e}} + \tilde{\mathrm{d}}_{\mathrm{c}}^T P_2 \tilde{\mathrm{d}}_{\mathrm{c}}\right) \tag{5.23}$$

where P_1 and P_2 are positive definite weighting matrices. Derivating (5.23) we get:

$$\dot{U} = s^T\left(\Psi_1(X)(\tilde{\mathrm{d}}_{\mathrm{e}} + \hat{\mathrm{d}}_{\mathrm{e}}) + E(X, \mathrm{d}_{\mathrm{c}})U + \nu(X, Y_r)\right) - \tilde{\mathrm{d}}_{\mathrm{e}}^T P_1 \dot{\hat{\mathrm{d}}}_{\mathrm{e}} - \tilde{\mathrm{d}}_{\mathrm{c}}^T P_2 \dot{\hat{\mathrm{d}}}_{\mathrm{c}} \tag{5.24}$$

Now, choose $U, \dot{\hat{\mathrm{d}}}_{\mathrm{e}}$ and $\dot{\hat{\mathrm{d}}}_{\mathrm{c}}$ in such a way that $\dot{U}$ in (5.24) will be negative semi-definite.

Consider:

$$U = \hat{E}^{-1}\left(-\nu(X, Y_c) - \Psi_1(X)\hat{\mathrm{d}}_{\mathrm{e}} - c_1 s\right) \tag{5.25a}$$

$$\dot{\hat{\mathrm{d}}}_{\mathrm{e}} = P_1^{-1}\Psi_1^T(X)s \tag{5.25b}$$

where c_1 is a strictly positive constant and:

$$\hat{E} = E(X, \hat{\mathrm{d}}_{\mathrm{c}}) \tag{5.26}$$

Before defining the adaptation dynamics for the control derivatives, let us see the what becomes the expression for $\dot{U}$ after replacing (5.25) in (5.24):

$$\dot{U} = -c_1 s^T s + s^T(E - \hat{E})U - \tilde{\mathrm{d}}_{\mathrm{c}}^T P_2 \dot{\hat{\mathrm{d}}}_{\mathrm{c}} \tag{5.27}$$

As a consequence of the fact that the control derivatives appear linearly in the expression for $E(X, \mathrm{d}_{\mathrm{c}})$ we can write:

$$(E - \hat{E})U = \Psi_d^T(X, U)\tilde{\mathrm{d}}_{\mathrm{c}} \tag{5.28}$$

So by substituting (5.28) in (5.27) and by using:

$$\dot{\hat{\mathrm{d}}}_{\mathrm{c}} = P_2^{-1}\Psi_d^T(X, U)s \tag{5.29}$$

We have as a result:

$$\dot{U} = -c_1 s^T s \leqslant 0$$

So that the control laws (5.25a), (5.25b) and (5.29) make the origin of the system (5.22) GAS.

Summary We can now summarize our adaptive controller scheme applied to the case of the airplane control. Consider the second order system of the airplane's fast dynamics, given by:

$$\ddot{Y} = \Psi_0(X) + \Psi_1(X)\mathrm{d_e} + E(X, \mathrm{d_c})U$$

Then, given a desired trajectory for each output as represented in (5.18) and defining a *generalized error* s as in (5.19) the control laws:

$$U = \hat{E}^{-1}\left(-\nu(X, Y_c) - \Psi_1(X)\hat{\mathrm{d}}_\mathrm{e} - c_1 s\right)$$
$$\dot{\hat{\mathrm{d}}}_\mathrm{e} = P_1^{-1}\Psi_1^T(X)s$$
$$\dot{\hat{\mathrm{d}}}_\mathrm{c} = P_2^{-1}\Psi_d^T(X, U)s$$

where P_1 and P_2 are positive definite matrices and c_1 in a strictly positive constant render the origin of the augmented system (5.22) GAS.

5.5 Results

Considerations As the parameter adaptation lies in the "measurement" of the states and control input of the aircraft at **each time step**, the adaptation scheme is stopped whenever there is control saturation. This leads to bad controller behavior for big amounts of saturation and that is why the amplitude of the step had to be smaller when compared to the simulations in chapter 3.

In order not to occupy too much space, only simulations of longitudinal dynamic variables will be presented, but we will keep in mind that the controller designed is valid for both longitudinal, latero-directional and mixed controls tasks.

In table 5.1 are presented the controller parameters, the model reference parameters, the desired trajectory signal and the initial values for the stability and control derivatives (the real parameters are presented in appendix A) for each simulation case.

TABLE 5.1 – Parameters used in the simulation of the adaptive controller. I_i represents the i^{th} order identity matrix.

Case	Reference Model	Controller Parameters	Desired Trajectory
I	$\omega_{n\phi} = 4$ $\omega_{n\theta} = 2.8$	$K_{s1} = 10I_3$ $K_{s0} = 5I_3$ $P_1 = diag(I_4, 10^{-2}, 10^{-1})$	$\theta_{des} = \theta_0 \;\; 0 \leqslant t < 1$ $\theta_{des} = 30^\circ \;\; 1 \leqslant t < 6$ $\theta_{des} = \frac{30(t-9)}{3}^\circ \;\; 6 \leqslant t < 9$ $\theta_{des} = \theta_0 \;\; t \geq 9$
II	$\xi_\phi = 0.7$ $\xi_\theta = 0.7$	$P_2 = diag(I_2, 10^{-1}, I_2)$ $c_1 = 10$	$\theta_{des} = \theta_0 \;\; 0 \leqslant t < 1$ $\theta_{des} = 5^\circ + 5^\circ(\sin \pi t/2)+$ $+2.5^\circ(\sin \pi t/4) \;\; t \geq 1$
Initial Parameter Guess			$\hat{C}_{M0}\|_{t=0} = -0.1$ $\hat{C}_{Mq}\|_{t=0} = 0.5C_{Mq}$ $\hat{C}_{Mdp}\|_{t=0} = 0.5C_{Mdp}$

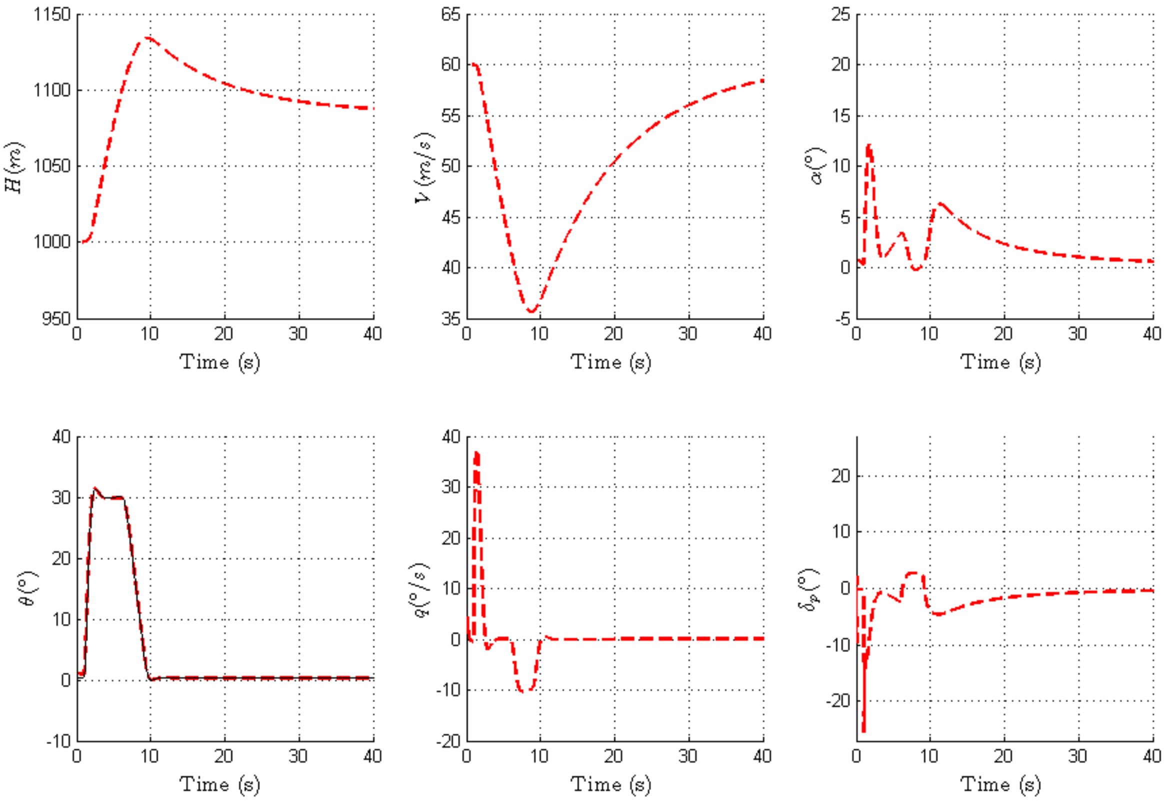


FIGURE 5.6 – Simulation case I, state variables and command surface deflection.

5.6 Commentaries

The difference between the two simulation cases lies in the desired trajectory. In the first case, a simple step in pitch was applied, while in the second case a combination of sinusoid signals was given. In both situations the adaptive controller is capable of

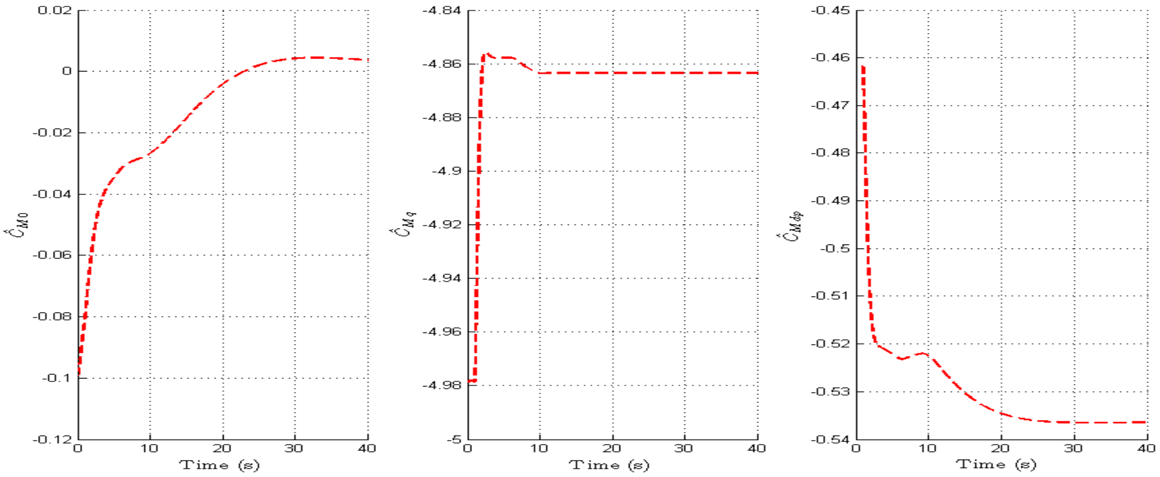


FIGURE 5.7 – Simulation case I, stability and control derivatives estimates.

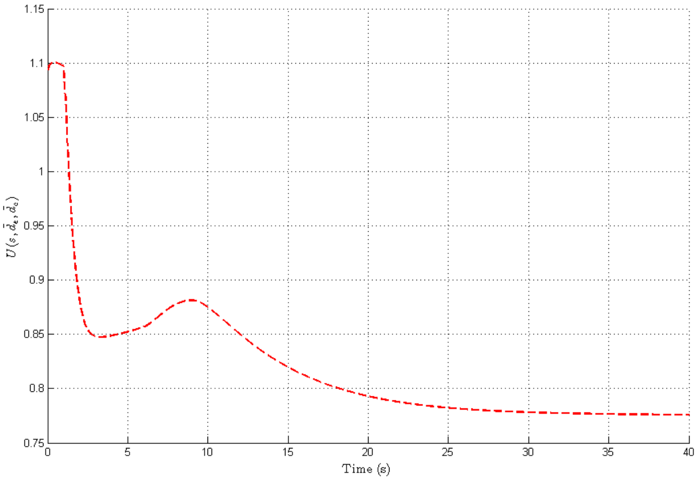


FIGURE 5.8 – Simulation case I, Lyapunov function value.

following the desired reference model trajectory as is seen in figures 5.6 and 5.9.

Notice now the difference between the parameter adaptation dynamics in figures 5.7 and 5.10. First see that the estimates only change their values if there is enough excitation in the given trajectory signal, as we also noticed with our simple example in section 5.2.1. As a result, for a simple step, control adaptation is stopped once the system is following a constant signal. On the other hand, for the case of persistent excitation the estimates do not stop changing their values.

Observe also the evolution of the proposed Lyapunov function in figures 5.8 and 5.11. In both cases the the Lyapunov function goes to a smaller value, but the excepted behavior for its derivative is not seen, that is, the relation $\dot{U} \leqslant 0$ does not hold for all instants.

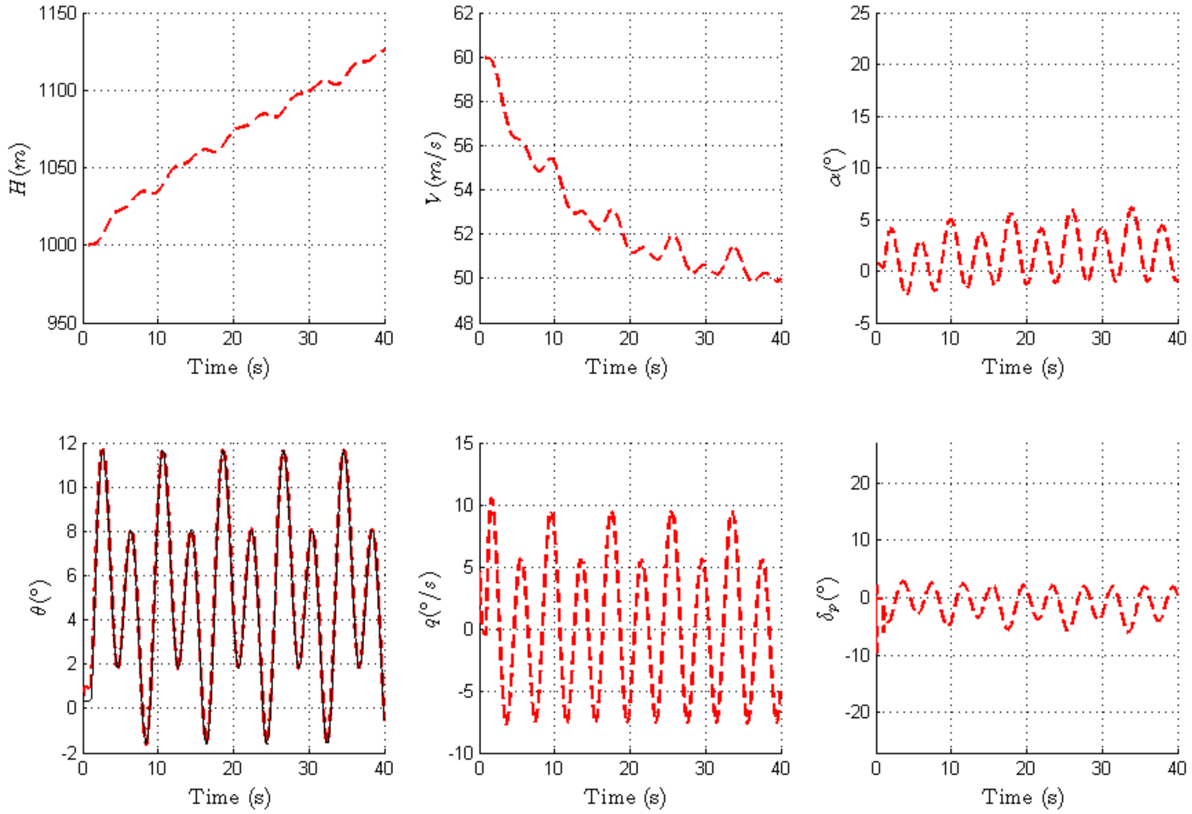


FIGURE 5.9 – Simulation case II, state variables and command surface deflection.

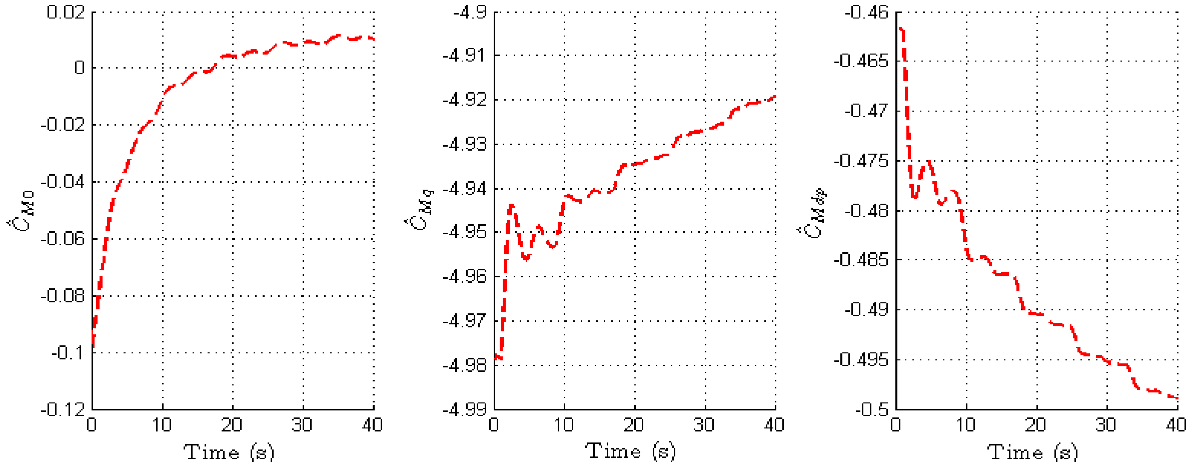


FIGURE 5.10 – Simulation case II, stability and control derivatives estimates.

This is a possible consequence of two facts:

- The controller does not consider actuator saturation (both in speed or amplitude), and so when any of those two takes place in the simulation the adaptation scheme is no longer working with the desired dynamic model.

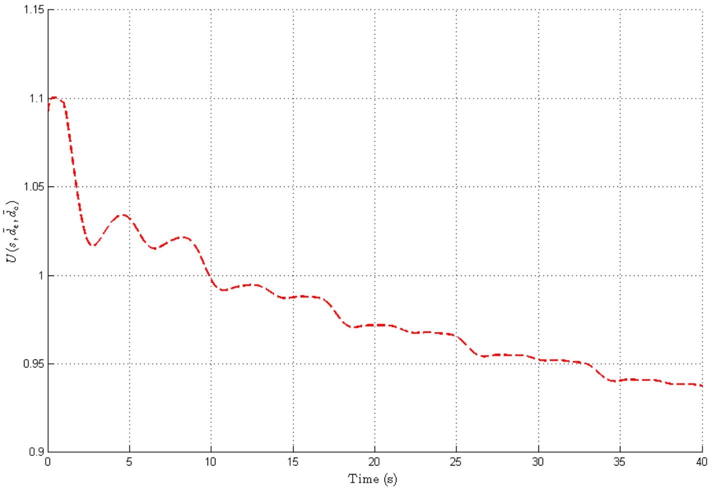


FIGURE 5.11 – Simulation case II, Lyapunov function value.

- As it can be seen further in (SINGH; STEINBERG, 1996), this adaptation scheme is only guaranteed to converge to the real plant parameters if the initial estimates lie "close enough" (within a given ball centered in the real parameters in the parameter space) to those in the first place. A way of measuring how far of the parameters can we go before convergence is no longer guaranteed is not yet well developed in theory.

Finally, let us notice an important fact. The value of the parameter $\hat{C}_{M0}$ which represents the natural pitching tendency of the aircraft and is not related to any of the flight variables specifically. That is the parameter that presents the better convergence. The adaptation behavior seen is precisely what we would expect if the CG location changed during flight (because of a load drop).

6 Sliding Mode Control

6.1 Introduction

As an alternative for dealing with inaccuracies in the dynamic models used for control, the Sliding Mode Control (SMC) is a technique which can be classified as a robust nonlinear method. This means that it can address uncertainties in the model during the design process to present a controller of satisfactory behaviour.

The theory of sliding mode control comes is a result of the study of *variable structure systems* (VSS) (UTKIN, 1977). A system with variable structure is described by structurally distinct differential equations depending on the region of the state space that one is considering. It will be seen that it is possible o combine useful properties of each of the structures to achieve a desired response. Moreover, it is possible to define a switching logic such as to make the system insensitive to certain parametric uncertainties in the models.

Naturally, this technique has its inconvenients such as introducing hard non-linearities into the control logic, which cannot be performed by physical systems, introducing delays and affecting the performance of the systems. In the worst cases, high frequency dynamics introduced by this switching between different structures can further affect the difference between the control model and the reality.

The theory presented in this chapter is based on (SLOTINE; LI, 1991; DECARLO; ZAK; MATTHEWS, 1988; YOUNG; UTKIN; OZGUNER, 1999) and the application to aircraft control, as well as further references on the fundamentals of variable structure control can be found in (YOUNG, 1977) and (UTKIN, 1977) respectively. It is worth noticing that, according to (SLOTINE; LI, 1991): “Sliding control has been successfully applied to robot manipulators, underwater vehicles, automotive transmissions and engines, high-performance electric motors, and power systems.”

6.2 Definitions

To introduce the concepts involved in the sliding mode control design let us consider the following simple second order system:

$$\ddot{x} - \xi\dot{x} + \psi x = 0, \quad \xi > 0 \tag{6.1}$$

where we will chose ψ such as:

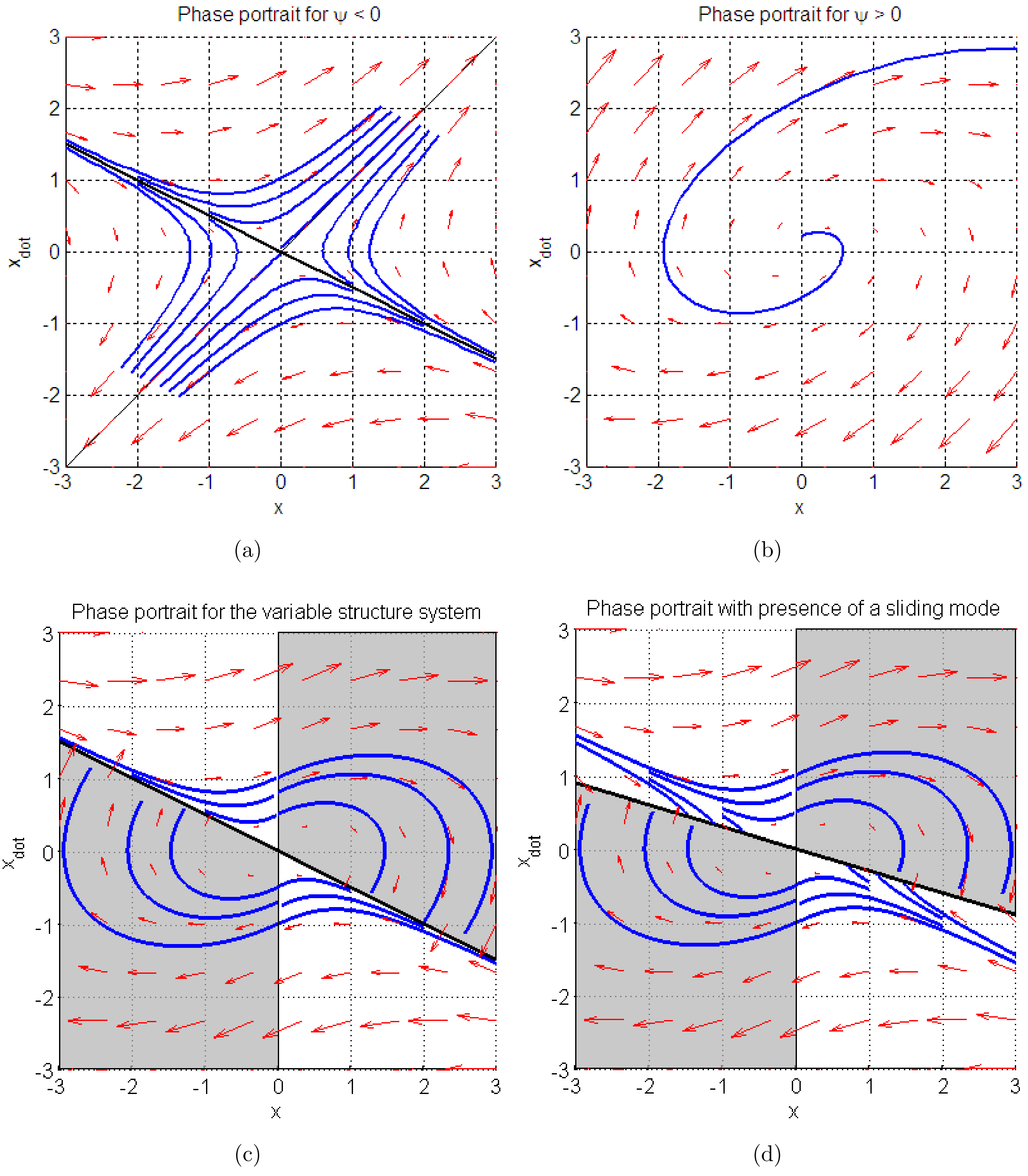
$$\psi = \begin{cases} \alpha & \text{if } xs > 0 \\ -\alpha & \text{if } xs < 0 \end{cases} \quad \text{where } s = cx + \dot{x} \tag{6.2}$$

$$c = \lambda_1 = -\tfrac{\xi}{2} + \sqrt{\tfrac{\xi^2}{4} + \alpha}, \quad \alpha > 0$$

That is, the structure of the system changes from one side of the surface $s = cx + \dot{x}$ to the other. To better understand the effect of this choice let us look at the phase portraits in figure 6.1.

In 6.1(a) we have the case when $\psi = -\alpha$. We see that the system is unstable and diverges asymptotically to the line defined by the eigenvector correspondent to the eigen value $\lambda_2 = -\frac{\xi}{2} - \sqrt{\frac{\xi^2}{4} + \alpha}$. The only stable path is the one associated with the direction defined by the eigenvector correspondent to λ_1. The origin in this case is a saddle point, thus the system is not stable. In 6.1(b) we have the case when $\psi = \alpha$. The system is also unstable and it spirals away from the origin.

Now, in 6.1(c) we have the phase portrait of the variable structure system defined by



(a) (b) (c) (d)

FIGURE 6.1 – Phase portraits for the simple VSS of second order

equations 6.1 and 6.2. It is seen that this new system is asymptotically stable, in contrast to its two unstable dynamics that form it. As said by (UTKIN, 1977), an even more fundamental aspect of VSS is the possibility to obtain trajectories not inherent in any of the structures. This trajectories describe a new type of motion - the sliding mode.

To see how this mode exists and behaves, let us consider the switching surface defined by $\sigma = \sigma(x, \dot{x}) = cx + \dot{x} = 0$ where now $0 < c < \lambda_1$. This means that our control law will have the form:

$$\psi = \begin{cases} \alpha & \text{if } x\sigma > 0 \\ -\alpha & \text{if } x\sigma < 0 \end{cases} \tag{6.3}$$

Note that the phase trajectories (figure 6.1(d)) are now directed towards the switching line and therefore, once on this line the system will remain on it. In order words, the system's velocities are always directed towards the line. This means that a new trajectory was introduced in the system. This trajectory is called the *sliding mode* and the switching surface used to achieve it is called the *sliding surface*. Note that since the sliding surface is an invariant set, it is both a surface and a dynamics for the system.

Now our attention turns into defining a rational procedure for choosing the switching surface such as to make the resulting system behave with a desired dynamics and selecting the controller gains such as to guarantee the existence and stability of the sliding mode.

6.3 The MIMO SMC design procedure

As seen by our simple example the design of variable structure controllers (VSC) breaks down into two major parts:

- The imposition of a switching surface so that the original system when restricted to move on the surface will respond in a desired manner
- The development of the switching control law which guarantees "sufficient conditions" for the existence and reachability of a sliding mode

Before developing the procedure it is necessary to present some general definitions, together with the citations of some theorems and theoretical background that justify the design process.

6.3.1 General definitions

6.3.1.1 Model

Let us consider again the nonlinear system with multiple inputs and multiple outputs, but affine in the control:

$$\dot{X} = f(X) + g(X)U \quad X \in \mathbb{R}^n \quad U \in \mathbb{R}^m \tag{6.4}$$

and now let us consider the following control dynamics:

$$u_i(t,x) = \begin{cases} u_i^+ & \text{if } \sigma_i > 0 \\ u_i^- & \text{if } \sigma_i < 0 \end{cases} \quad i = 1, 2, \ldots, m \tag{6.5}$$

6.3.1.2 Switching surface

In equation 6.5 we make use of the existence of an $(n-m)$-dimensional switching surface $\sigma = 0$ of which $\sigma_i(X) = 0$ is the i^{th} component, that is:

$$\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_m] = 0 \tag{6.6}$$

As noticed in (DECARLO; ZAK; MATTHEWS, 1988), although it is possible to use general nonlinear switching surfaces as 6.6, it is more common to use linear ones (hyperplanes), so in this work the surfaces will have the form:

$$\sigma = SX(t) = 0 \quad \text{where } S \text{ is a constant } nXn \text{ matrix} \tag{6.7}$$

6.3.1.3 Sliding mode: existence and uniqueness theorems

Once a switching surface is defined according to one needs (next section) it will be of our interest to generate a control law such that this surface is reached at a time t_0 and that the system stays in it for subsequent times $t \geq t_0$. We are talking then of turning our switching surface into an asymptotically attractive, invariant set of a discontinuous differential equation.

Based on our previous example in section 6.2 we see that in order for a sliding mode to exist, the velocities vectors have to point towards the surface in its vicinities, that is, as stated in (UTKIN, 1977) and further clarified in the references therein:

$$\lim_{s\to -0} \dot{s} > 0 \quad \text{and} \quad \lim_{s\to +0} \dot{s} < 0 \tag{6.8}$$

This can be proved by using the second Lyapunov method for the stability of nonlinear systems, considering a neighbourhood of the chosen switching surface and demonstrating that a valid Lyapunov function $V(t, X, \sigma)$ exists in this domain. More information as well as complete references for these proofs can be found in (UTKIN, 1977; DECARLO; ZAK; MATTHEWS, 1988). Note that for 6.8 to exist, we need an infinitely fast switching device. As this is never achievable by physical systems, for reasons such as delays and hysteresis, an oscillation will occur in the neighbourhood of the sliding mode. This oscillation is called *chattering*, and can be a serious drawback of sliding mode control, if the introduced frequencies in the system excite dynamics which are not normally considered in the initial model.

Once guaranteed that the sliding mode exists, it is left to prove that this mode will be unique for it to be useful for our control purposes. In (DECARLO; ZAK; MATTHEWS, 1988; SLOTINE; LI, 1991) we can find the description of the Filippov method, in which it is proven that indeed this motion is unique. We leave the reader to the references for more specific information.

6.3.2 Sliding surface design

6.3.2.1 Equivalent Control

The concept of equivalent control was introduced by Utkin and Drazenovic, as an alternate way to the Filippov method to describe the unique dynamics that a system undergoes when in the sliding mode. Moreover, this concept is more readily applicable to multi-input systems, which makes it very useful in the subsequent analysis.

Suppose a system passes through a sliding surface $\sigma = 0$ in time t_0 and stays there for subsequent times. Then it is right to state that for $t \geq t_0$:

$$\sigma(X) = 0$$
$$\dot{\sigma}(X) = 0$$

This second expression can be expressed in the following form:

$$\dot{\sigma}(X) = \frac{\partial\sigma}{\partial X}\dot{X} = \frac{\partial\sigma}{\partial X}\left[f(X) + g(X)U\right]$$

The control U that satisfies the condition $\dot{\sigma}(X) = 0$ is called the *Utkin-Drazenovic equivalent control* and its given by:

$$U_{eq} = -\left[\frac{\partial\sigma}{\partial X}g(X)\right]^{-1}\frac{\partial\sigma}{\partial X}f(X) \tag{6.9}$$

When this equivalent control is applied once again into the dynamics 6.4 we have the sliding mode dynamics:

$$\dot{X} = \left[I - g(X)\left[\frac{\partial\sigma}{\partial X}g(X)\right]^{-1}\frac{\partial\sigma}{\partial X}\right]f(X) \tag{6.10}$$

In the case discussed further on in this work, we will consider that $\sigma(X)$ is the intersection of hyperplanes ($\sigma(X) = SX$) so that:

$$\dot{X} = \left[I - g(X)\left[Sg(X)\right]^{-1}S\right]f(X) \tag{6.11}$$

6.3.2.2 Regular Form and order reduction

Equation 6.11 together with the sliding surface equation 6.7 provide a way to calculate S, our desired sliding surface, from a given desired dynamics for the sliding mode.

To see how we can do that, we first note that the system we are working with once in the sliding mode is composed of only $n-m$ independent differential equations (given that, by hypothesis: $rank(S) = m$). The process of explicitly reducing the order of the system to obtain the resulting dynamics is called order reduction.

However, more then a simple order reduction, once in sliding mode those $n-m$ equations should not depend on the control, since the equivalent control is calculated from the system states. We are then interested in reducing our system to the following form:

$$\begin{cases} \dot{X}_1 = f_1(t,X) & \text{where } X_1 \in \mathbb{R}^{n-m} \\ \dot{X}_2 = f_2(t,X) + g_2(t,X)U & \text{where } X_2 \in \mathbb{R}^{m} \end{cases} \tag{6.12}$$

$$\text{with } SX = [S_1 S_2][X_1 X_2]^T = 0 \tag{6.13}$$

Equation 6.12 is the system in the *regular form.* To have a system as the one in 6.4 presented as 6.12 we try using a non-singular linear transformation of the state X obtaining $Z = TX$. The system can then be written as:

$$\dot{Z} = Tf(t,X) + Tg(t,X)U$$

now, if T is such that:

$$Tg(t,X) = \begin{bmatrix} 0_{(n-m)Xm} \\ \hat{g}_2(t,Z) \end{bmatrix}^T$$

then the system can be written in the regular form as:

$$\begin{cases} \dot{Z}_1 = \hat{f}_1(t,Z) & \text{where } X_1 \in \mathbb{R}^{n-m} \\ \dot{Z}_2 = \hat{f}_2(t,Z) + \hat{g}_2(t,Z)U & \text{where } X_2 \in \mathbb{R}^{m} \end{cases} \tag{6.14}$$

Now, to actually perform the process of order reduction note that from 6.13 we can write:

$$SX = ST^{-1}Z = [S_1 S_2]T^{-1}[Z_1 Z_2]^T = [\hat{S}_1 \hat{S}_2][Z_1 Z_2]^T = 0$$

remembering the hypothesis that $rank(S) = m$ we can suppose that $\hat{S}_2$ is non-singular and solve for Z_2:

$$Z_2 = -\hat{S_2}^{-1}\hat{S}_1 Z_1$$

substituting this result in the equation for Z_1 in 6.14 we get:

$$\dot{Z}_1 = \hat{f}_1(t, Z_1, -\hat{S_2}^{-1}\hat{S}_1 Z_1) \tag{6.15}$$

which is the *reduced order equivalent dynamics* of the sliding mode.

6.3.2.3 Choice of sliding surface

Depending on the nature of $\hat{f}_1$ and on the desired dynamics for the sliding mode, we can try to chose S such as to satisfy the constraints imposed. Note that if $\hat{f}_1$ is a complex nonlinear function this task could be very difficult, if not impossible. Let us turn our attention to the linear case where we can write 6.15 as:

$$\dot{Z}_1 = \left[\hat{A_{11}} - \hat{A_{12}}\hat{S_2}^{-1}\hat{S}_1\right] Z_1 \tag{6.16}$$

Note the very desirable form of 6.16. As noted by (DECARLO; ZAK; MATTHEWS, 1988), if the pair (A_{11}, A_{12}) is controllable, then it is possible to locate the eigenvalues of the reduced order equivalent dynamics anywhere on the complex plane by choosing S accordingly.

It is now clear that, once we have the reduced order equivalent dynamics equation, which depends on S, it is sufficient to impose a desired dynamics to have the means to find the correspondent sliding surface S.

6.3.3 Controller Design

Various different methods for a VSC design are available in the literature (DECARLO; ZAK; MATTHEWS, 1988; SLOTINE; LI, 1991). Here we will turn our attention only to the Method of Control Hierarchy. This is done because of the limited scope of this work, due to the capacity of the method to deal with parameter uncertainties in the model and finally because it is a straight forward approach to multi-input systems.

6.3.3.1 Method of Control Hierarchy

Following the references already presented, once we have chosen our sliding surface, the method of control hierarchy consists in using each control channel to satisfy one of the surface's equations.

We begin by making a cascade of suppositions: first suppose that control u_1 will guarantee that the system is in σ_1 for all values of $\begin{bmatrix} u_2 & \dots & u_m \end{bmatrix}$; then suppose that control u_2 will guarantee that the system will be on the intersection $\sigma_1 \cap \sigma_2$ for all values of $\begin{bmatrix} u_3 & \dots & u_m \end{bmatrix}$ and so on until and including the control channel u_{m-1}.

With that scheme, once we arrive in u_m, by hypothesis the system is already in the intersection $\sigma_1 \cap \sigma_2 \cap \dots \cap \sigma_{m-1}$, and thus it presents an equivalent structure Σ^{m-1} generated by all the previous choices of control channels. It is then only necessary for u_m to guarantee the convergence of the system Σ^{m-1} to the surface σ_m.

This convergence will happen if u_m is chosen such as to make:

$$\sigma_m \dot{\sigma_m} < 0 \tag{6.17}$$

The last step is to come back, designing u_{k+1} for the system Σ^k to reach and stay in the surface σ_{k+1} (that is, to make $\sigma \dot{\sigma_m} < 0$) as a function of the already chosen u_{k+i} $(i = i, \dots, m-k)$.

Summering, we take what is written in (YOUNG, 1977) to present an algorithm for the method of control hierarchy:

1. Suppose a hierarchy of switching hyperplanes is specified by $s_1 \rightarrow s_2 \rightarrow \dots \rightarrow s_m$
2. Let i = m

3. Suppose sliding mode occurs on $s_j = 0$, $j = 1, \ldots, i-1$. Solve for the equivalent control u_{eq}^{i-1} of the variable structure control $(u^{i-1})^T = (u_1, \ldots, u_{i-1})$ as a function of u_k, $k = i, \ldots, m$ from the algebraic equations $s_j = 0$, $j = 1, ..., i-1$

4. For the hyperplane $s_i = 0$, find $u_i^+(x)$ and $u_i^-(x)$ such that $\dot{s}_i s_i < 0$

5. Let $i - 1$. If $i \geq 0$, go to Step 3, else stop.

The method will be clarified by means of the following application to aircraft longitudinal attitude control.

6.3.4 An alternative method for MIMO sliding mode control design

Although the method of control hierarchy with all the procedures described above is one valid solution for the variable structure controller problem, we describe here another method, proposed by (SLOTINE; LI, 1991) and described in (SILVA, 2007) for the nonlinear longitudinal control of an hypersonic aircraft.

This new method will be presented because it is much more straight forward and bears a heavier physical intuition in its development, without state transformations and the calculations required in the procedures already seen.

Instead of exactly following what was presented in (SILVA, 2007), we will develop here of a sliding mode controller for a linear system which is much easier to deduce and present and makes a more adequate example for the objectives of this work. The reader should bear in mind that much more general controllers for nonlinear systems of the type (3.1) can be developed with this theory.

The problem Consider the linear system:

$$\dot{X} = AX + Bu \tag{6.18}$$

and a switching control law:

$$u_{des} = \begin{cases} u_i^+ & \text{if } \sigma_i > 0 \\ u_i^- & \text{if } \sigma_i < 0 \end{cases} \qquad i = 1, \ldots, m \tag{6.19}$$

where the switching surface $\sigma = 0$ is defined by:

$$\sigma = SX = 0 \tag{6.20}$$

where S is a MxN matrix. The problem consists in finding the control law in (6.19) such as to make the chosen sliding surface (6.20) a stable invariant set of the controlled system's dynamics ((6.18),(6.19)).

Note that S can be chosen such as to select an output from the state X. In this case, solving our control problem is equivalent to make our output go to and stay in zero.

Design Procedure Following (SILVA, 2007) and (SLOTINE; LI, 1991), we start by diving our control law in two parts: the equivalent control and a discontinuous control of the form (6.19) such as:

$$u = u_{eq} + u_{dis} \tag{6.21}$$

To calculate u_{eq}, we simply solve the equation for the motion in the sliding surface: $S\dot{X} = 0$. This gives:

$$u_{eq} = -(SB)^{-1}SAX \tag{6.22}$$

we note then the condition that SB needs to be invertible. This needs to be taken into account when choosing S. Thus the choice of the sliding surface in this method is simply constrained by this latter condition, as opposed to the previous method where S was chosen to satisfy a given desired dynamics.

Now, to choose u_{dis} we take into account that our control must:

- Render the sliding surface an attractive, asymptotically stable set.

- Provide stability to the sliding modes.

Then, considering (6.22), the resultant controlled dynamics has the form:

$$S\dot{X} = SBu_{dis} \tag{6.23}$$

So, if u_{dis} is chosen to be:

$$u_{dis} = -(SB)^{-1} \begin{cases} k_1 sign(S_1X) & \\ \vdots & k_i > 0 \\ k_m sign(S_mX) & \end{cases} \tag{6.24}$$

where S_i is the i^{th} column of S and the $sign(.)$ function is defined to be:

$$sign(x) \equiv \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}$$

we will have as a resultant dynamics for the sliding surface the equation:

$$S\dot{X} = - \begin{bmatrix} k_1 & 0 & \dots & 0 \\ 0 & k_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & k_m \end{bmatrix} \begin{bmatrix} sign(S_1X) \\ sign(S_2X) \\ \vdots \\ sign(S_mX) \end{bmatrix}$$

which according to (SLOTINE; LI, 1991) makes the surface an asymptotically stable set. Note that from (6.24) the discontinuous control of each control channel will be used to drive the system to the correspondent hyperplane. He have then used the equivalent control to decouple the control problem. Further on, we can previously chose which outputs we want to drive to zero in the choice of our sliding surface (6.20) and then use the adequate control channels to perform the task more adequately. This process will made clear in the next section.

It is still left to demonstrate that the sliding modes are stable. To do that we use the Lyapunov's direct method of Theorem1, according to (DECARLO; ZAK; MATTHEWS, 1988; SLOTINE; LI, 1991; SILVA, 2007). First we choose as a Candidate Lyapunov Function the quadratic form:

$$V(t, SX) = (SX)^T P(SX)$$

where P is a positive definite matrix. The function is then *positive semi-definite*, *radially unbounded* in $\mathbb{R}^N$. Derivating $V(t, X)$ with respect to time we get:

$$\frac{\partial V}{\partial t} = \dot{X}^T S^T PSX + X^T S^T PS\dot{X}$$

substituting the system dynamics (6.18):

$$\frac{\partial V}{\partial t} = (AX + Bu)^T S^T PSX + X^T S^T PS(AX + Bu)$$

$$\frac{\partial V}{\partial t} = X^T A^T S^T PSX + u^T B^T S^T PSX + X^T S^T PSAX + X^T S^T PSBu$$

finally substituting our control law (6.21) with (6.22) and (6.24):

$$\frac{\partial V}{\partial t} = \begin{bmatrix} -k_1 sign(S_1 X) \\ \vdots \\ -k_m sign(S_m X) \end{bmatrix}^T PSX + X^T S^T P \begin{bmatrix} -k_1 sign(S_1 X) \\ \vdots \\ -k_m sign(S_m X) \end{bmatrix}$$

which is a negative definite function, rendering the sliding surface **GAS** for the controlled system.

Summary We conclude that our design of a control law composed by the equivalent and discontinuous terms, control the system (6.18) in a way that makes the sliding surface a GAS manifold of the state space. To do that, note that the control law will switch between the sides of the sliding surface according to its discontinuous part, making the velocities always point towards the surface, just as in 6.2.

6.4 Application to aircraft longitudinal control

Based on the work of (YOUNG, 1977), but using the technique by (SLOTINE; LI, 1991),we chose, for the application of the methods, a model following, variable structure controller for the linearised longitudinal dynamics of the Navion in levelled flight at $50m/s$ at an altitude of $1000m$:

$$\dot{X}_p = A_p X_p + B_p u_p \tag{6.25}$$

$$A_p = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -2.54 & -5.57 & 0.01 \\ 0 & 0.97 & -1.89 & -0.01 \\ -9.81 & -0.03 & 7.49 & -0.05 \end{bmatrix} \quad B_p = \begin{bmatrix} 0 & 0 \\ -9.42 & 0 \\ -0.15 & 0 \\ -0.18 & 1.79 \end{bmatrix} \tag{6.26}$$

where the state of the plant is $X_p = [\theta \quad q \quad \alpha \quad V]^T$ and its control is the elevator and thrust commands $u_p = [\delta_p \quad \delta_\pi]^T$.

Our objective will be to make its longitudinal dynamics behave as the one of an airbus A310 in the phase of approach to descent, flying levelled at $150m/s$ and $2000m$, given by:

$$\dot{X}_m = A_m X_m + B_m u_m \tag{6.27}$$

$$A_m = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.66 & -2.00 & 0 \\ 0 & 1.00 & -0.82 & 0 \\ -9.81 & 0.02 & 3.91 & -0.01 \end{bmatrix} \quad B_m = \begin{bmatrix} 0 & 0 \\ -2.92 & 0 \\ -0.07 & 0 \\ -0.51 & 3.26 \end{bmatrix} \tag{6.28}$$

where the state X_m and control u_m vectors are defined analogously.

The error between the plant and the model dynamics is defined by:

$$e \equiv X_p - X_m$$

Error Dynamics The error dynamics is then:

$$\dot{e} = A_m e + (A_p - A_m)X_p - B_m u_m + B_p u_p \tag{6.29}$$

Sliding Surface Choice Using our physical intuition, and bearing in mind that according to (6.22) SB_p must be non-singular, we chose as our control switching surface:

$$Se = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.30}$$

That is, the errors in angle of attack and in the total velocity are zero on the surface. By the definition of the discontinuous control part (6.24), we will then use the elevator to control the angle of attack and the thrust to drive the velocity error to zero.

Equivalent Control Calculation By its definition, the equivalent control in this case is defined to make $S\dot{e} = 0$, so from (6.29):

$$u_{peq} = -(SB_p)^-1 \left[SA_m e + S(A_p - A_m)X_p - SB_m u_m \right] \tag{6.31}$$

Discontinuous Control Following (6.24) we have:

$$u_{pdis} = -(SB_p)^{-1} \begin{bmatrix} k_1 sign(S_1 e) \\ k_2 sign(S_2 e) \end{bmatrix} \quad \text{where } S_i \text{ is the } i^{th} \text{ line of } S \tag{6.32}$$

From 6.3.4 we have that the control $u_p = u_{peq} + u_{pdis}$ where u_{peq} and u_{pdis} are defined in (6.31) and (6.32) will make the system (6.29) converge to the surface (6.30) asymptotically and stay there.

Attenuating the *Chattering* Following the suggestion presented in (SLOTINE; LI, 1991), to attenuate the *chattering* phenomenon, instead of choosing a hard non linearity (the switching control law) across the switching surface, we use the saturation function (6.33) to smooth-en the control in a chosen *n dimensional* neighbourhood of thickness Φ of the latter thus attenuating the *chattering*.

$$sat(x,\phi) \equiv \begin{cases} -1 & \text{if } \frac{x}{\phi} < -1 \\ x & \text{if abs}\frac{x}{\phi} \leq 1 \\ 1 & \text{if } \frac{x}{\phi} > 1 \end{cases} \tag{6.33}$$

The discontinuous part of our control law becomes then:

$$u_{pdis} = -(SB_p)^{-1} \begin{bmatrix} k_1 sat(S_1 e, \phi_1) \\ k_2 sat(S_2 e, \phi_2) \end{bmatrix} \quad \Phi = [\phi_1 \quad \phi_2]^T chosen. \tag{6.34}$$

It will not be demonstrated here, but one can find in (SLOTINE; LI, 1991) the demonstration that the control combined by (6.31) and (6.34) also makes the switching surface an asymptotically stable manifold.

6.4.0.1 Simulation

The simulation was performed in the following conditions: The reference model angle of attack is initially +2° out of its equilibrium value and a *doublet* entrance in the elevator of 1° amplitude and duration of $3s$ is applied after the 1^{st} second of simulation. The controlled plant starts its equilibrium and is programmed to follow the reference model angle of attack and absolute wind velocity. The controller gains were chosen $k_1 = 0.7$ and $k_2 = 1$ and the saturation thickness was chosen $\phi_\alpha = 0.05$ (which is equivalent to 2.85°) and $\phi_V = 1$.

The simulation covers the evolution in time of three different systems: The reference linear model in (6.28), the linearised dynamics of the Navion (6.26) and finally the complete, nonlinear dynamics of the Navion. For the linear systems, no actuator dynamics was considered, being possible for the commands ($u_m \quad u_p$) to be discontinuous. For the nonlinear system, the actuators were modelled with a first order dynamics of constants

$\tau_e = 100$ and $\tau_\pi = 4$ where the subscripts e and π mean the elevator and thrust commands respectively. Moreover, the actuators had saturation in positions of $[-20°, 20°]$ for the elevator and of $[0, 1]$ for the thrust.

Three cases were analysed. First, the nominal case, in which the nonlinear model simulated is precisely the one that generated the linear model in (6.26). Second, the mass of the nonlinear model was changed to 95% of its initial value but the linear model remained the same. Finally, the third case is the same as the second, only without saturation in the commands.

Results The reference model variables, the variables of the linear model for the Navion and the variables from the nonlinear complete dynamics are plotted as dashed black lines, blue lines and red lines respectively in figures 6.2 to 6.7.

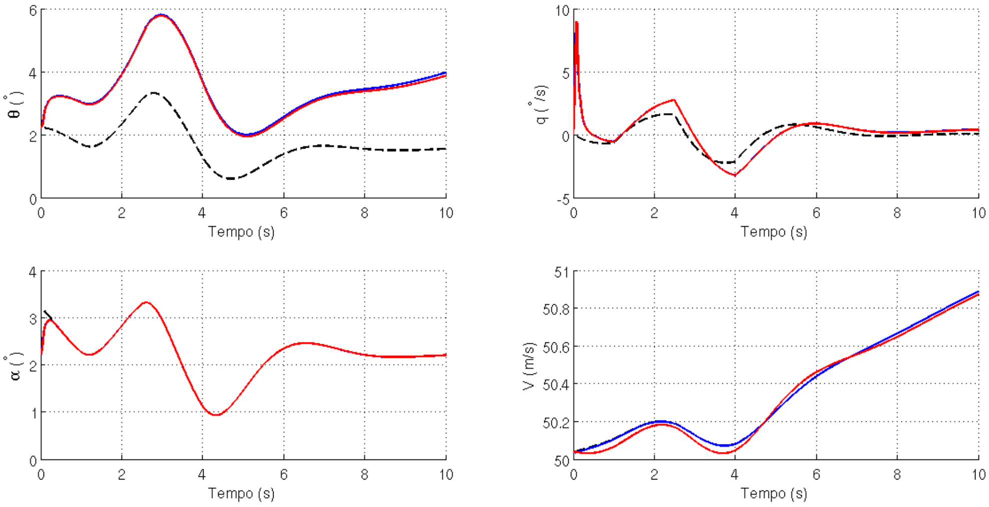


FIGURE 6.2 – Sliding mode controller simulation: Longitudinal states evolution for the nominal case.

Comments It is seen from figures 6.2 to 6.4 that in the nominal case, the controller is capable to perform the task of driving the angle of attack and velocity of the plant to the ones of the model without any saturation. Moreover, from figure 6.4 we note that the error in alpha goes to zero much faster then the one in velocity, due to the efficiency of the control channels in the vehicle dynamics.

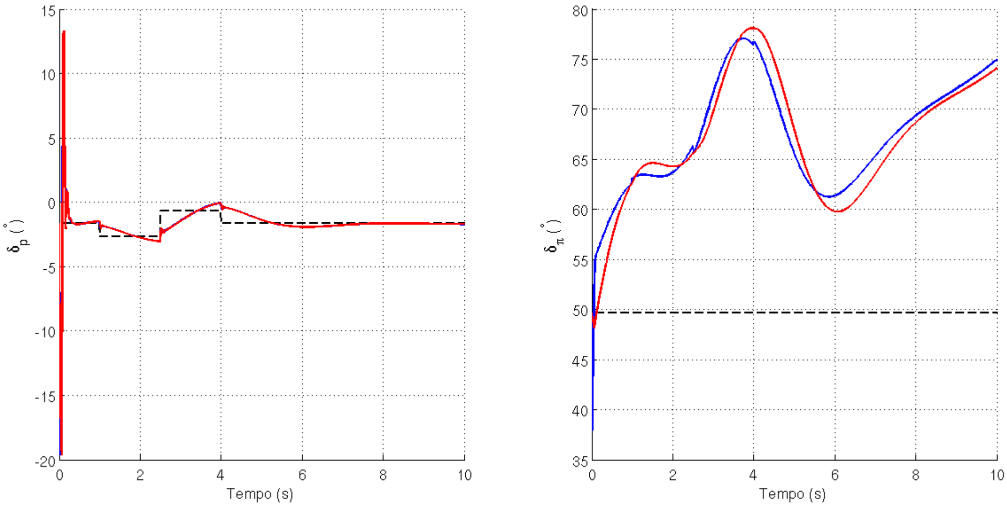


FIGURE 6.3 – Sliding mode controller simulation: Actuators position for the nominal case.

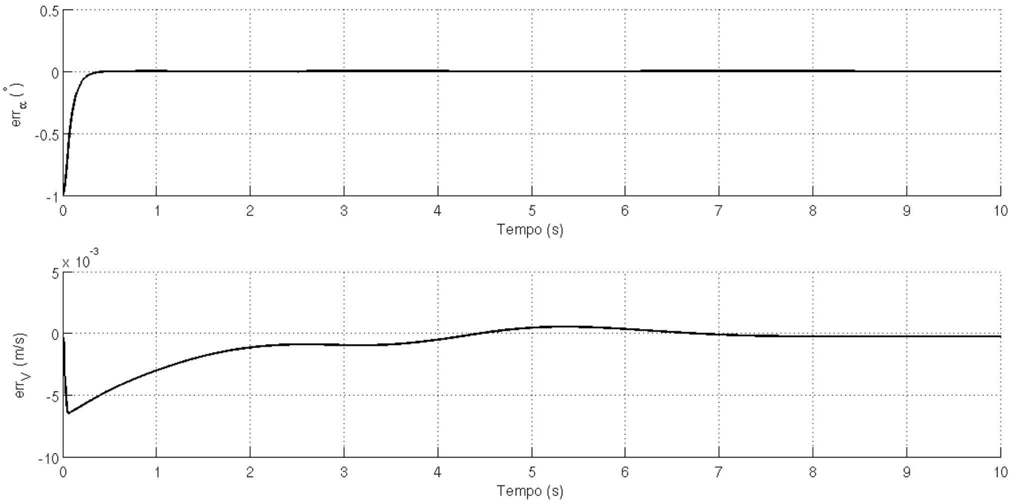


FIGURE 6.4 – Sliding mode controller simulation: Errors between the reference model and the controlled linearised system.

In the second case, despite of the change in mass, the controller is still able to follow the reference model until about $7s$ of simulation 6.5, when the thrust channel saturates 6.6 and tracking is naturally lost. This verifies some robustness of the method with respect to the plant parameters, as is verified finally in 6.7 where we see satisfactory tracking of the state when no saturations were imposed to the system.

Note also that there was no occurrence of the *chattering* phenomenon, as made clear by the smoothness of the states and commands, confirming the validity of the method

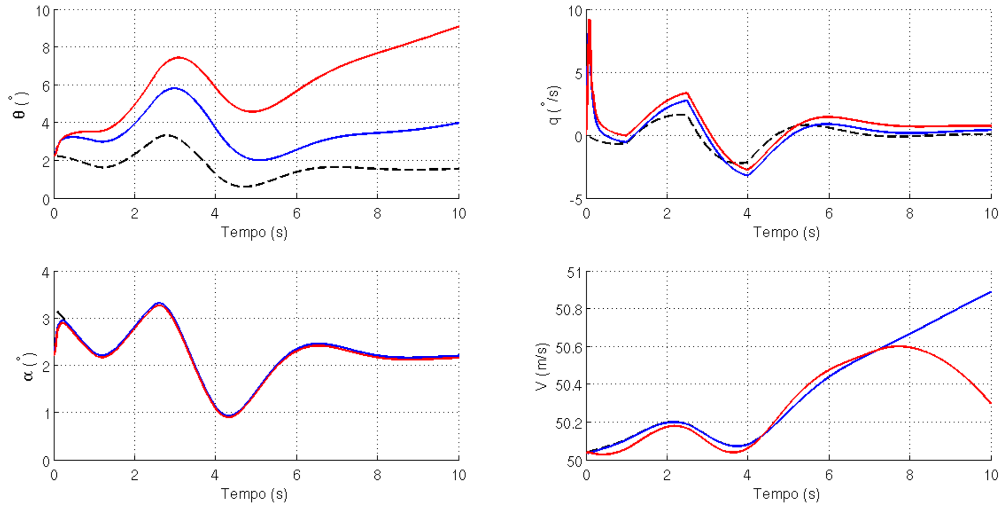


FIGURE 6.5 – Sliding mode controller simulation: Longitudinal states evolution for the case of 95% mass

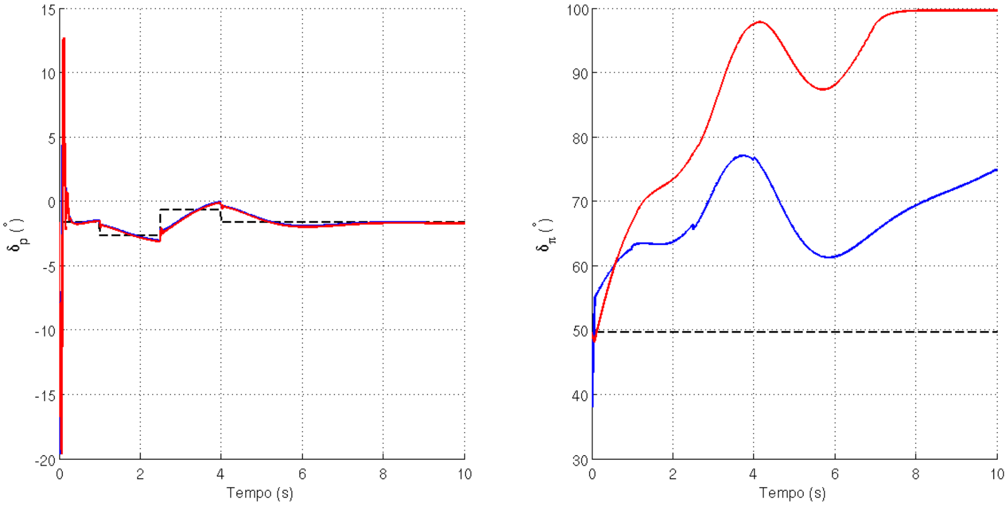


FIGURE 6.6 – Sliding mode controller simulation: Actuators position for the case of 95% mass

presented.

6.5 Commentaries

Much more has been studied and presented in the literature regarding variable structure controllers or sliding mode control, as seen in (SLOTINE; LI, 1991; DECARLO; ZAK; MATTHEWS, 1988; SILVA, 2007; YOUNG; UTKIN; OZGUNER, 1999), to cite

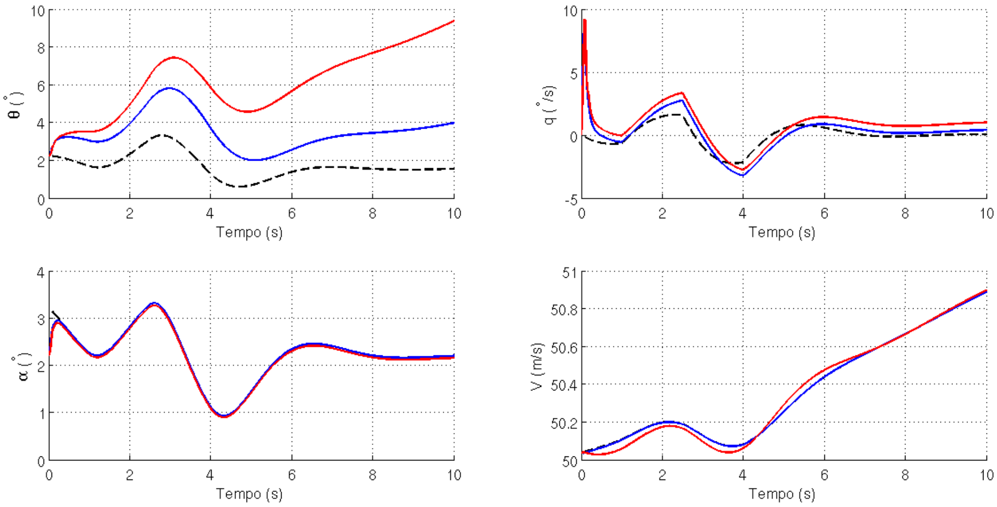


FIGURE 6.7 – Sliding mode controller simulation: Longitudinal states evolution for the case of 95% mass without control saturation

a few. The method can be classified as a nonlinear robust control technique, since it doesn't require linear models of the systems neither perfect models of the equations, since its discontinuous control can handle parameter uncertainty.

The sliding surface, where the switching of controls takes place, can be chosen as a sum of the errors in the outputs of the system, or a vector composed of such outputs that one wants to drive to zero. In that way, we simplify our control problem by reducing the order of the system. Then, using simpler control laws, like the ones in (6.22) and (6.24) we guarantee that the new dynamics will have these surfaces as stable, attractive sets.

As discussed in (YOUNG; UTKIN; OZGUNER, 1999), digital control methods can be used to study the *chattering* phenomenon, but as suggested in (SLOTINE; LI, 1991), the control law can be smoothed in a chosen boundary of the surface in order to considerably reduce or eliminate chattering while keeping the surface an stable and attractive manifold of at least a sub-domain of the state space.

7 Final Comments

7.1 Conclusion

Nonlinear control theory offers a path to perform control tasks not yet imagined with classical controllers. Specifically talking about flight control, it is emphasized its importance to the control of high performance, military aircraft and nowadays its promising application in UAV's and MAV's dedicated to reconnaissance and surveillance tasks, and also a future application to combat oriented unmanned aircraft vehicles. The controller complexity does not pose an obstacle to the application in airplanes, due to the advances in electronics and processing power. Civil aviation however, will see an application of this theory in a much slower pace. That is mainly for two reasons: First, the flight envelope for civil airplanes is not as wide and second due to the many security proofs the controller has to pass to achieve certification.

From a mathematical point of view, we notice the close dependence of the nonlinear control laws with the dynamical modeling of the plant. This represents at the same time an advantage an a drawback. It is an advantage from the point of view that, by considering the dynamical model, we can drive our plant to follow virtually any desired behavior, as long as it is possible to have the required control effort and that the trajectory is entirely in a domain where the used dynamical model is still valid. On the other hand, it becomes a drawback when the model used is not precise enough or in the presence of unmodelled dynamics and when the parameters change during the plant operation. This drawback is such that an "arm" of the nonlinear controller theory nowadays is the research of many different adaptation controller schemes.

To conclude the ideas developed within the text and the results, let us make a list of advantages and drawbacks in the application

7.1.1 Advantages

- The same control laws can be used throughout the hole flight envelope without the need for parameter adjusting.
- Some nonlinear techniques give the designer the ability to keep the naturally stabilizing nonlinearities of the system and thus do not waste control effort.
- To a certain extent, adaptive control can deal with unexpected changes in the dynamics or with dynamics that were not modeled in the first place.
- Nonlinear controller present a promising tool to perform complex trajectories that reach the limits of the flight envelope.

7.1.2 Drawbacks

- There are still no well developed analytical tools to proof the stability of nonlinear controllers in the presence of incertitude in plant parameters. Even though simulations can give a feeling for the performance of such controllers, it is necessary to further develop robustness analysis to nonlinear controllers.
- The performance of nonlinear controller when there is saturation of input is also an active research field. (STEINBERG; PAGE, 1998)
- The computational cost of such schemes is much higher then the classical schemes, thus requiring more electronics and finally making the overall implementation subjected to problems.

7.2 Suggestions for Future Work

It will be left some suggestions for the continuity of this work, divided in the different topics discussed earlier.

Aircraft Models / Simulation

- Further develop the flight simulation model to start considering high angle of attack flight.
- Include atmospheric effects as wind an turbulence.
- Develop a flight model that uses *Quaternions* (ROBINSON, 1958; PHILLIPS; HAILEY; GEBERT, 2001) as means of changing referential frames. This finishes with the singularities found in the dynamic equations for the euler angles and thus permits to analyze more complex trajectories.

Feedback Linearization / Backstepping

- Develop a controller that makes use of *quaternion* dynamics directly in the control loop.
- Use an adaptive control scheme in the second part of the Backstepping controller (the part where the signal “u” was used to obtain the surface deflections) so as to become less sensible to parameter change or dynamics that were not well modeled.

Adaptive Control

- Develop a simple, least square based, self-tuning controller and check its performance for the same task studied in this work.
- Based on (KRSTIĆ; KANELLAKOPOULOS; KOKOTOVIĆ, 1992), program a MRAC without over parameterizing to search for computer efficiency.

- Study the coupling of both adaptation schemes (SLOTINE; LI, 1991) to the airplane case.

Bibliography

DECARLO, R.; ZAK, S.; MATTHEWS, G. Variable structure control of nonlinear multivariable systems: a tutorial. **Proceedings of the IEEE**, v. 76, p. 212–232, 1988.

FOSSEN, T. I.; STRAND, J. P. **Nonlinear Ship Control (Tutorial Section)**. [S.l.], 1998.

HARKEGARD, O. **Backstepping Designs for Aircraft Control: What is there to gain?** [S.l.], 2001.

HARKEGARD, O. **Flight Control Design Using Backstepping**. Dissertação (Mestrado) — Linkopings Universitet, 2001.

ISIDORI, A. **Nonlinear Control Systems**. 3rd. ed. New Jersey: Springer-Verlag, 1995.

KHALIL, H. K. **Nonlinear Systems**. 3rd. ed. New Jersey: Prentice-Hall, 2002.

KRSTIĆ, M.; KANELLAKOPOULOS, I.; KOKOTOVIĆ, P. V. Adaptive nonlinear control without overparametrization. **Systems & Control Letters**, v. 19, p. 177–185, 1992.

MONAHEMI, M. M.; KRISTIĆ, M. Control of wing rock motion using adaptive feedback linearization. **Journal of Guidance, Control, and Dynamics**, v. 19, n. 4, p. 905–912, 1996.

NAM, K.; ARAPOSTATHIS, A. A model reference adaptive control scheme for pure-feedback nonlinear systems. **IEEE Transactions on Automatic Control**, v. 33, n. 9, p. 803–811, 1988.

NELSON, R.; SMITH, S. **Flight Stability and Automatic Control**. New York: McGraw-Hill, 1989.

PHILLIPS, W.; HAILEY, C.; GEBERT, G. Review of attitude representations used for aircraft kinematics. **Journal of Aircraft**, v. 38, n. 4, p. 718–737, 2001.

ROBINSON, A. C. **On the Use of Quaternions in Simulation of Rigid-Body Motion**. [S.l.], 1958.

SCHLICHTING, H.; TRUCKENBRODT, E. **Aerodynamics of the Airplane**. New York: McGraw-Hill, 1979.

SILVA, A. L. D. **Procedimento de Projeto de Leis de Controle de Vôo de Aeronaves Utilizando o Controle a Estrutura Variável**. Dissertação (Mestrado) — Instituto Tecnológico de Aeronáutica - ITA, 2007.

SINGH, S.; STEINBERG, M. Adaptive control of feedback linearizable nonlinear systems with application to flight control. **Journal of Guidance, Control, and Dynamics**, v. 19, n. 4, p. 871–877, 1996.

SLOTINE, J.-J. E.; LI, W. **Applied Nonlinear Control**. Englewood Cliffs, New Jersey: Prentice-Hall, 1991.

STEINBERG, M. L.; PAGE, A. **Nonlinear Flight Control with a Backstepping Design Aproach**. [S.l.], 1998.

STEVENS, B. L.; LEWIS, F. L. **Aircraft Control and Simulation**. [S.l.]: Wiley-IEEE, 2003.

TENENBAUM, R. A. **Fundamentals of Applied Dynamics**. 175 Fifth Avenue, New York: Springer-Verlag, 2003.

UTKIN, V. Variable structure systems with sliding modes. **IEEE Transactions on Automatic Control**, AC-22, p. 214–222, 1977.

YOUNG, K. D.; UTKIN, V. I.; OZGUNER, U. A control engineer's guide to sliding mode control. **IEEE Transactions on control systems technology**, v. 7, n. 3, p. 328–342, 1999.

YOUNG, K.-K. Design of multivariable variable structure model following control systems. **IEEE Conference on Decision and Control**, v. 16, p. 1042–1048, 1977.

Appendix A - Aircraft Model

Navion

Dimensions:			
$S_{ref} = 17.0942m^2$	$mac = 1.7374m$	$b = 10.1803$	
Mass and Inertia:			
m $= 1123.7$ Kg	$I_{xx} = 1415.5Kgm^2$	$I_{yy} = 3999, 7Kgm^2$	$I_{zz} = 4765, 7Kgm^2$
$I_{xz} = -142, 4Kgm^2$			
Propulsion:			
$F_{maxi} = 2200N$	$V_i = 45m/s$	$n_{h0} = 0.3$	$n_{h1} = -0.0602$
$n_{h2} = 3.5355$	$\alpha_f = 0$	$\beta_f = 0$	
Aerodynamics:			
$C_{l0} = 0.3$	$C_{l\alpha} = 4.44$	$C_{ldp} = 0.355$	$C_{lq} = 3.8$
$C_{l\dot{\alpha}} = 0$	$C_{ybeta} = -0.564$	$C_{yda} = 0$	$C_{ydr} = 0.157$
$C_{d0} = 0.0468$	$k_1 = -0.0450$	$k = 0.0750$	$Cm0 = 0$
$C_{m\alpha} = -0.683$	$C_{mdp} = -0.923$	$C_{mq} = -9.96$	$C_{m\dot{\alpha}} = -4.36$
$C_{L\beta}$ = -0.074	$C_{Lp} = -0.41$	$C_{Lr} = 0.107$	$C_{Lda} = -0.134$
$C_{Ldr} = 0.107$	$C_{N\beta} = 0.071$	$C_{Np} = -0.0575$	$C_{Nr} = -0.125$
$C_{Nda} = -0.0035$	$C_{Ndr} = -0.072$		

FOLHA DE REGISTRO DO DOCUMENTO

1. CLASSIFICAÇÃO/TIPO TC	2. DATA 13 de novembro de 2009	3. DOCUMENTO Nº CTA/ITA/TC-119/2009	4. Nº DE PÁGINAS 121

5. TÍTULO E SUBTÍTULO:

Review of Nonlinear Techniques Applied to Aircraft Control

6. AUTOR(ES):

Gustavo Oliveira Violato

7. INSTITUIÇÃO(ÕES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÕES):

Instituto Tecnológico de Aeronáutica. Divisão de Engenharia Aeronáutica – ITA/IEA

8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR:

Control Não-Linear; Mecânica do Vôo; Controle de Aeronaves; Controle Adaptativo

9. PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO:

Controle de aeronaves; Sistemas não-lineares; Controle adaptativo; Controladores; Simulação computadorizada; Mecânica de vôo; Engenharia aeronáutica

10. APRESENTAÇÃO: (**X**) **Nacional** ( ) **Internacional**

ITA, São José dos Campos. Curso de Graduação em Engenharia Aeronáutica. Orientador: Prof. Dr. Ing. Pedro Paglione; co-orientador Prof. Dr. Félix Mora-Camino. Publicado em 2009.

11. RESUMO:

This work covers the study and application of nonlinear control techniques to flight control of fixed wing aircrafts. The techniques covered are dynamic inversion, backstepping, nonlinear adaptive control and sliding mode control. The objective is to present a review of the methods introduced in the literature in the last decades. The fundamentals of each theory are explored followed by demonstrative applications to the problem of flight control. Simulations are performed for various flight cases and the controller performance is critically analysed.

12. GRAU DE SIGILO:

(**X**) **OSTENSIVO** ( ) **RESERVADO** ( ) **CONFIDENCIAL** ( ) **SECRETO**