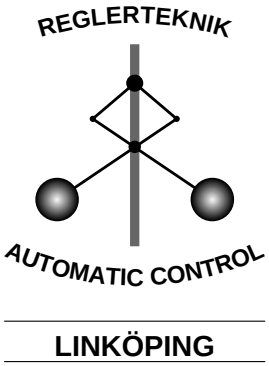
Linköping Studies in Science and Technology
Thesis No. 875

Flight Control Design Using Backstepping

Ola Härkegård



Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE–581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
E-mail: ola@isy.liu.se

Linköping 2001

Flight Control Design Using Backstepping

Department of Electrical Engineering,
Linköpings universitet,
SE–581 83 Linköping,
Sweden.

ISBN 91-7219-995-4
ISSN 0280-7971
LiU-TEK-LIC-2001:12

Printed by UniTryck, Linköping, Sweden 2001

To Eva

Abstract

Aircraft flight control design is traditionally based on linear control theory, due to the existing wealth of tools for linear design and analysis. However, in order to achieve tactical advantages, modern fighter aircraft strive towards performing maneuvers outside the region where the dynamics of flight are linear, and the need for nonlinear tools arises.

In this thesis we investigate *backstepping* as a new framework for nonlinear flight control design. Backstepping is a recently developed design tool for constructing globally stabilizing control laws for a certain class of nonlinear dynamic systems. Flight control laws for two different control objectives are designed. First, general purpose maneuvering is considered, where the angle of attack, the sideslip angle, and the roll rate are the controlled variables. Second, automatic control of the flight path angle control is considered.

The key idea of the backstepping designs is to benefit from the naturally stabilizing aerodynamic forces acting on the aircraft. The resulting state feedback control laws thereby rely on less knowledge of these forces compared to control laws based on feedback linearization, which today is the prevailing nonlinear design technique within aircraft flight control.

The backstepping control laws are shown to be inverse optimal with respect to meaningful cost functionals. This gives the controllers certain gain margins which implies that stability is preserved for a certain amount of control surface saturation.

Also, the problem of handling a model error appearing at the input of a nonlinear dynamic system is treated, by considering the model error as an unknown, additive disturbance. Two schemes, based on adaptive backstepping and nonlinear observer design, are proposed for estimating and adapting to such a disturbance. These are used to deal with model errors in the description of the aerodynamic moments acting on the aircraft.

The designed control laws are evaluated using realistic aircraft simulation models and the results are highly encouraging.

Acknowledgments

First of all, I want to thank Professor Lennart Ljung for drafting me to the Automatic Control group in Linköping, and hereby giving me the opportunity to perform research within a most professional, ambitious, and inspiring group of people. I also want to thank my supervisors Professor Torkel Glad and Karin Ståhl Gunnarsson for their guidance and expertise within nonlinear control theory and aircraft control, respectively.

Besides these key persons in particular and the Automatic Control group in general, a few people deserve an explicit "Thank you!": Fredrik Tjärnström, Mikael Norrlöf, and Jacob Roll proofread the thesis and provided valuable comments, significantly increasing the quality of the result. Anders Helmersson shared his practical flight experience and commented on the computer simulation results. Ingegerd Skoglund and Mikael Rönnqvist at the Department of Mathematics suggested the numerical schemes which were used for control allocation in the implementation of the controllers.

And now to something completely different: *Ett stort tack till släkt och vänner, och inte minst till kärestan, som uthärdat den senaste tiden då jag levt i ett socialt vakuum, och som stöttat mig i vått och torrt!*

This work was sponsored by the graduate school ECSEL.

Ola Härkegård

Linköping, March 2001

Contents

1 Introduction 1
- 1.1 Introductory Example: Sideslip Regulation . . . 2
- 1.2 Outline of the Thesis . . . 4
- 1.3 Contributions . . . 4

2 Aircraft Primer 7
- 2.1 The Impact of Automatic Control . . . 7
- 2.2 Control Objectives . . . 9
- 2.3 Control Means . . . 12
- 2.4 Aircraft Dynamics . . . 12
  - 2.4.1 Governing physics . . . 13
  - 2.4.2 Modeling for control . . . 16
- 2.5 Current Approaches to Flight Control Design . . . 20
  - 2.5.1 Gain-scheduling . . . 20
  - 2.5.2 Dynamic inversion (feedback linearization) . . . 21
  - 2.5.3 Other nonlinear approaches . . . 24

2.A Wind-axes Force Equations . . . 25

3 BACKSTEPPING 29
3.1 Lyapunov Theory . . . 30
3.2 Lyapunov Based Control Design . . . 32
3.3 Backstepping . . . 33
3.3.1 Main result . . . 34
3.3.2 Which systems can be handled? . . . 36
3.3.3 Which design choices are there? . . . 37
3.4 Related Lyapunov Designs . . . 42
3.4.1 Forwarding . . . 43
3.4.2 Adaptive, robust, and observer backstepping . . . 43
3.5 Applications of Backstepping . . . 43

4 INVERSE OPTIMAL CONTROL 45
4.1 Optimal Control . . . 45
4.2 Inverse Optimal Control . . . 46
4.3 Robustness of Optimal Control . . . 48

5 BACKSTEPPING DESIGNS FOR FLIGHT CONTROL 51
5.1 General Maneuvering . . . 51
5.1.1 Objectives, dynamics, and assumptions . . . 52
5.1.2 Backstepping control design . . . 54
5.1.3 Flight control laws . . . 62
5.1.4 Practical issues . . . 65
5.2 Flight Path Angle Control . . . 65
5.2.1 Objectives, dynamics, and assumptions . . . 66
5.2.2 Backstepping control design . . . 67
5.2.3 Flight control law . . . 72
5.2.4 Practical issues . . . 72

6 ADAPTING TO INPUT NONLINEARITIES AND UNCERTAINTIES 75
6.1 Background . . . 75
6.2 Problem Formulation . . . 76
6.3 Adaptive Backstepping . . . 78
6.4 Observer Based Adaption . . . 79
6.4.1 The general case . . . 79
6.4.2 The optimal control case . . . 81
6.5 A Water Tank Example . . . 82
6.6 Application to Flight Control . . . 86
6.6.1 General maneuvering . . . 86
6.6.2 Flight path angle control . . . 86

7 IMPLEMENTATION AND SIMULATION 87
7.1 Aircraft Simulation Models . . . 87
7.1.1 GAM/ADMIRE . . . 87
7.1.2 HIRM . . . 88

7.1.3 FDC 88
7.2 Controller Implementation 89
7.2.1 Control allocation 90
7.3 Simulation 92
7.3.1 Conditions 92
7.3.2 Controller parameters 92
7.3.3 Simulation results 93

7.A Aircraft Data 100

8 Conclusions 101

Bibliography 103

Notation

Symbols

$\mathbb{R}$	the set of real numbers
$x_1, \ldots, x_n$	state variables
$x = (x_1 \ \cdots \ x_n)^T$	state vector
u	control input
$k(x)$	state feedback control law
$V(x)$	Lyapunov function, control Lyapunov function
x_1^{ref}	reference value of x_1
x_1^{des}	desired value of x_1
Γ	multiplicative control input perturbation
e	unknown additive input bias
$\hat{e}$	estimate of e

Operators

$\|x\| = \sqrt{x_1^2 + \ldots + x_n^2}$	Euclidian norm
$\dot{V} = \frac{dV}{dt}$	time derivative of V

$V'(x_1) = \frac{dV(x_1)}{dx_1}$	derivative of V w.r.t. its only argument x_1
$V_x(x) = (\frac{\partial V(x)}{\partial x_1} \ \cdots \ \frac{\partial V(x)}{\partial x_n})$	gradient of V w.r.t. x

Acronyms

clf	control Lyapunov function
GAS	global asymptotic stability, globally asymptotically stable
NDI	nonlinear dynamic inversion
TVC	thrust vectored control
GAM	Generic Aerodata Model
HIRM	High Incidence Research Model

Aircraft nomenclature

State variables

Symbol	**Unit**	**Definition**
α	rad	angle of attack
β	rad	sideslip angle
γ	rad	flight path angle
$\Phi = (\phi, \theta, \psi)^T$		aircraft orientation (Euler angles)
ϕ	rad	roll angle
θ	rad	pitch angle
ψ	rad	yaw angle
$\omega = (p, q, r)^T$		body-axes angular velocity
$\omega_s = (p_s, q_s, r_s)^T$		stability-axes angular velocity
p	rad/s	roll rate
q	rad/s	pitch rate
r	rad/s	yaw rate
$\mathbf{p} = (p_N, p_E, h)^T$		aircraft position
p_N	m	position north
p_E	m	position east
h	m	altitude
$\mathbf{V} = (u, v, w)^T$		body-axes velocity vector
u	m/s	longitudinal velocity
v	m/s	lateral velocity
w	m/s	normal velocity
V_T	m/s	total velocity
M	-	Mach number
n_z	g	normal acceleration, load factor

Control surface deflections

Symbol	Unit	Definition
δ		collective representation of all control surfaces
δ_{es}	rad	symmetrical elevon deflection
δ_{ed}	rad	differential elevon deflection
δ_{cs}	rad	symmetrical canard deflection
δ_{cd}	rad	differential canard deflection
δ_r	rad	rudder deflection

Aircraft data

Symbol	Unit	Definition
m	kg	aircraft mass
$I = \begin{pmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{xz} & 0 & I_z \end{pmatrix}$	$\mathrm{kg\,m^2}$	aircraft inertial matrix
S	$\mathrm{m^2}$	wing planform area
b	m	wing span
$\bar{c}$	m	mean aerodynamic chord
Z_{TP}	m	z_b-position of engine thrust point

Atmosphere

Symbol	Unit	Definition
ρ	$\mathrm{kg/m^3}$	air density
$\bar{q}$	$\mathrm{N/m^2}$	dynamic pressure

Forces and moments

Symbol	Unit	Definition
g	$\mathrm{m/s^2}$	acceleration due to gravity
F_T	N	engine thrust force
$D = \bar{q}SC_D$	N	drag force
$L = \bar{q}SC_L$	N	lift force
$Y = \bar{q}SC_Y$	N	side force
$\bar{L} = \bar{q}SbC_l$	Nm	rolling moment
$M = \bar{q}S\bar{c}C_m$	Nm	pitching moment
$N = \bar{q}SbC_n$	Nm	yawing moment

Coordinate systems

Symbol	Definition
(x_b, y_b, z_b)	body-axes coordinate system
(x_s, y_s, z_s)	stability-axes coordinate system
(x_w, y_w, z_w)	wind-axes coordinate system

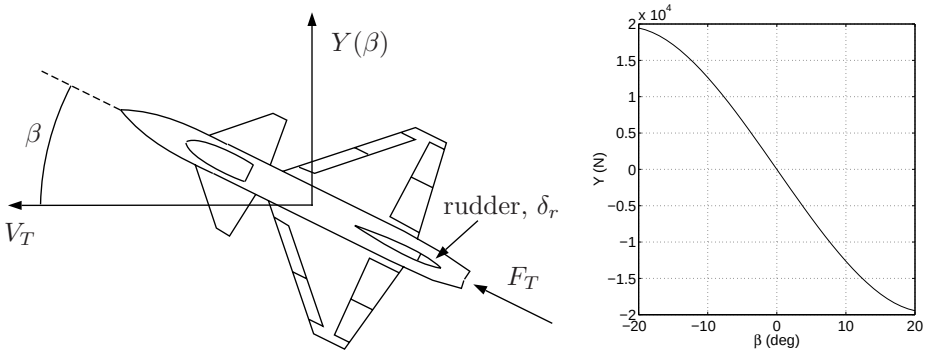
1

Introduction

During the past 15 years, several new design methods for control of nonlinear dynamic systems have been invented. One of these methods is known as backstepping. Backstepping allows a designer to methodically construct stabilizing control laws for a certain class of nonlinear systems.

Parallel to this development within nonlinear control theory, one finds a desire within aircraft technology to push the performance limits of fighter aircraft towards "supermaneuverability". By utilizing high angles of attack, tactical advantages can be achieved, as demonstrated by Herbst [32] and Well et al. [77], who consider aircraft reversal maneuvers for performance evaluation. The aim is for the aircraft to return to a point of departure at the same speed and altitude but with an opposite heading at minimum time. It is shown that using high angles of attack during the turn, the aircraft is able to maneuver in less air space and complete the maneuver in shorter time. These types of maneuvers are performed outside the region where the dynamics of flight are linear. Thus, linear control design tools, traditionally used for flight control design, are no longer sufficient.

In this thesis, we investigate how backstepping can be used for flight control design to achieve stability over the entire flight envelope. Control laws for a number of flight control objectives are derived and their properties are investigated, to see what the possible benefits of using backstepping are. Let us begin by illustrating the key ideas of the design methodology with a concrete example.



(a) Top view of aircraft

(b) Side force vs. sideslip

Figure 1.1 *The sideslip, β, is in general desired to be kept zero. The aerodynamic side force, $Y(\beta)$, naturally acts to fulfill this objective.*

1.1 Introductory Example: Sideslip Regulation

An important aircraft variable to be controlled is the sideslip angle, β, which is depicted in Figure 1.1(a). Nonzero sideslip means that the aircraft is "skidding" through the air. This is unwanted for several reasons: it is unsuitable for the engines, especially at high speeds, it increases the air resistance of the aircraft, and it is uncomfortable for the pilot. Thus, $\beta = 0$ is in general the control objective.

To regulate the sideslip, the rudder, situated at the back of the aircraft, is used. The (somewhat simplified) sideslip dynamics are given by

$$\dot{\beta} = -r + \frac{1}{mV_T}(Y(\beta) - F_T \sin\beta) \tag{1.1a}$$

$$\dot{r} = cN(\delta_r) \tag{1.1b}$$

Here, r = yaw rate of the aircraft, m = aircraft mass, V_T = aircraft speed, Y = aerodynamic side force due to the air resistance, F_T = thrust force produced by the engines, c = constant related to the aircraft moment of inertia, N = yawing moment, and δ_r = rudder deflection.

The side force, which is the component of the air resistance that affects the sideslip, is a nonlinear function of β. Figure 1.1(b) shows the typical relationship between the two entities. In particular we see that for a negative sideslip, the side force is positive and vice versa for a positive sideslip. Thus, considering (1.1a), the side force is *useful* for bringing the sideslip to zero since, e.g., for $\beta < 0$ its contribution to $\dot{\beta}$ is positive. The same is true for the $\dot{\beta}$ component due to the engine thrust force. For $\beta < 0$, $-F_T \sin\beta$ is positive, bringing β towards zero.

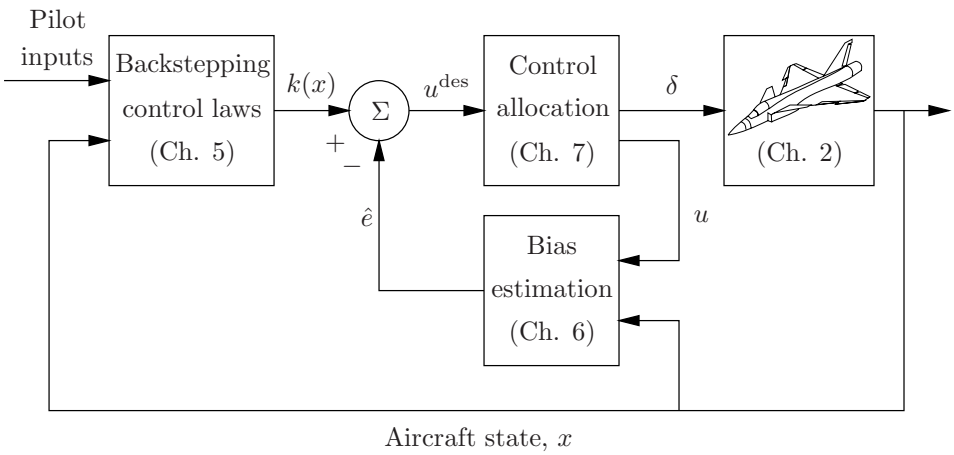


Figure 1.2 *Controller configuration.*

The important implication of these discoveries is that since the nonlinear terms act stabilizing, they need not be cancelled by the controller, and hence complete knowledge of them is not necessary. This idea of recognizing useful nonlinearities and benefitting from them rather than cancelling them is the main idea of this thesis.

In Chapter 5 we use backstepping to design state feedback laws for various flight control objectives. A common feature of the derived control laws is that they are linear in the variables used for feedback, considering the angular acceleration of the aircraft as the input. In our example this corresponds to designing a control law

$$u = k(\beta, r) = -k_1\beta - k_2 r$$

where

$$u = \dot{r} = cN(\delta_r) \tag{1.2}$$

To realize such a control law in terms of the true control input, δ_r, requires perfect knowledge of the yawing moment, N. Since typically this is not the case, we can remodel (1.2) as

$$u = c\hat{N}(\delta_r) + e$$

where $\hat{N}$ is our model of the yawing moment and e is the bias to the actual yawing moment. An intuitively appealing idea is to compute an estimate, $\hat{e}$, of the bias, e, on-line and realize the modified control law

$$c\hat{N}(\delta_r) = k(\beta, r) - \hat{e} \tag{1.3}$$

which, if the estimate is perfect, cancels the effect of the bias and achieves $u = k(\beta, r)$ as desired. In Chapter 6, two such estimation schemes are proposed and shown to give closed loop stability.

The remaining problem is that of control allocation, i.e., how to determine δ_r such that (1.3) is satisfied. This will be discussed in Chapter 7 where two numerical solvers are proposed.

The overall control configuration, along with chapter references, is shown in Figure 1.2.

1.2 Outline of the Thesis

Chapter 2 Contains basic facts about modern fighter aircraft such as their dynamics, the control inputs available and what the control objectives are.

Chapter 3 Introduces the backstepping methodology for control design for a class of nonlinear system. Discusses design choices and contains examples showing how some nonlinearities can actually be useful and how to benefit from them.

Chapter 4 A short chapter on inverse optimal control, i.e., how one can decide whether a given control law is optimal w.r.t. a meaningful cost functional.

Chapter 5 Contains the main contributions of the thesis. Backstepping is used to design state feedback control laws for various flight control objectives.

Chapter 6 Proposes two different methods for adapting to model errors appearing at the input and investigates closed loop stability in each case.

Chapter 7 Proposes numerical schemes for solving the control allocation problem. Also presents computer simulations of the designed aircraft control laws in action.

Chapter 8 Concludes the thesis by evaluating the ability to handle important issues like stability, tuning, robustness, input saturation, and disturbance attenuation within the proposed backstepping framework.

1.3 Contributions

The main contributions of this thesis are the following:

- The ideas in Sections 5.1.2 and 5.2.2 on how to benefit from the naturally stabilizing aerodynamic forces using backstepping, rather than cancelling them as in feedback linearization. The resulting control laws rely on less knowledge of the aerodynamic forces than the feedback linearizing designs which have been previously proposed (reviewed in Section 2.5.2).
- The backstepping designs for the two general nonlinear systems in Sections 5.1.2 and 5.2.2.

- The discovery in Section 5.1.3 that the angle of attack control law used by Snell et al. [71], based on feedback linearization and time-scale separation, can also be constructed using backstepping and is in fact optimal w.r.t. a meaningful cost functional.
- The adaptive schemes in Chapter 6 for handling model errors appearing at the control input.
- The computer simulations in Section 7.3 showing the proposed control laws to work satisfactory using realistic aircraft simulation models.

Parts of this thesis have been published previously. The backstepping designs in Chapter 5 originate from

> Ola Härkegård and S. Torkel Glad. A backstepping design for flight path angle control. In *Proceedings of the 39th Conference on Decision and Control*, pages 3570–3575, Sydney, Australia, December 2000.

and

> Ola Härkegård and S. Torkel Glad. Flight control design using backstepping. Technical Report LiTH-ISY-R-2323, Department of Electrical Engineering, Linköpings universitet, SE-581 83 Linköping, Sweden, December 2000. To be presented at the *5th IFAC Symposium "Nonlinear Control Systems" (NOLCOS'01)*, St. Petersburg, Russia.

The results in Chapter 6 on how to adapt to a model error at the input can be found in

> Ola Härkegård and S. Torkel Glad. Control of systems with input nonlinearities and uncertainties: an adaptive approach. Technical Report LiTH-ISY-R-2302, Department of Electrical Engineering, Linköpings universitet, SE-581 83 Linköping, Sweden, October 2000. Submitted to the *European Control Conference, ECC 2001*, Porto, Portugal.

2

Aircraft Primer

In this chapter, we investigate modern fighter aircraft from a control perspective. The aim is to introduce the reader to the aircraft dynamics, the control objectives, and how to assist the pilot to achieve these objectives using automatic control. Much of what is said applies to aircraft in general, not only to fighters.

Section 2.1 discusses the possibilities of using electric control systems to aid the pilot in controlling the aircraft. In Sections 2.2 and 2.3, the control objectives are presented along with the control means at our disposal. In Section 2.4 we turn to the dynamics of flight, and in Section 2.5 some existing approaches to flight control design are presented.

2.1 The Impact of Automatic Control

The interplay between automatic control and manned flight goes back a long time, see Stevens and Lewis [74] for a historic overview. At many occasions their paths have crossed, and progress in one field has provided stimuli to the other.

During the early years of flight technology, the pilot was in direct control of the aircraft control surfaces. These where mechanically connected to the pilot's manual inputs. In modern aircraft, the pilot inputs are instead fed to a control system. Based on the pilot inputs and available sensor information, the control system computes the control surface deflections to be produced. This information is sent

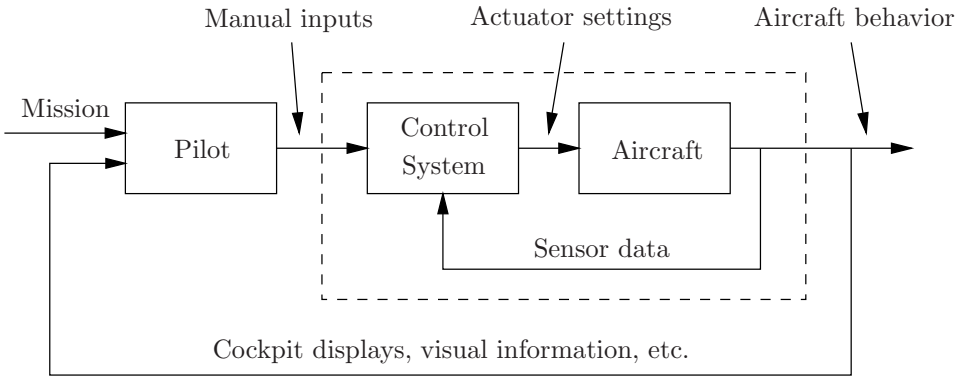


Figure 2.1 *Fly-by-wire from a control perspective.*

through electrical wires to the actuators located at the control surfaces, which in turn realize the desired deflections. Figure 2.1 shows the situation at hand. This is known as *fly-by-wire* technology. What are the benefits of this approach?

Stability Due to the spectacular 1993 mishap, when a fighter aircraft crashed over central Stockholm during a flight show, it is a widely known fact, even to people outside the automatic control community, that many modern aircraft are designed to be unstable in certain modes. A small disturbance would cause the uncontrolled aircraft to quickly diverge from its original position. Such a design is motivated by the fact that it enables faster maneuvering and enhanced performance. However, it also emphasizes the need for reliable control systems, stabilizing the aircraft for the pilot.

Varying dynamics The aircraft dynamics vary with altitude and speed. Thus without a control system assisting him, the pilot himself would have to adjust his joystick inputs to get the same aircraft response at different altitudes and speeds. By "hiding" the true aircraft dynamics inside a control loop, as in Figure 2.1, the varying dynamics can be made transparent to the pilot by designing the control system to make the closed loop dynamics independent of altitude and speed.

Aircraft response Using a control system, the aircraft response to the manual inputs can be selected to fulfill the requirements of the pilots. By adjusting the control law, the aircraft behavior can be tuned much more easily than having to adjust the aircraft design itself to achieve, e.g., a certain rise time or overshoot.

Interpretation of pilot inputs By passing the pilot inputs to a control system, the meaning of the inputs can be altered. In one mode, moving the joystick sideways may control the roll rate, in another mode, it may control the roll angle. This

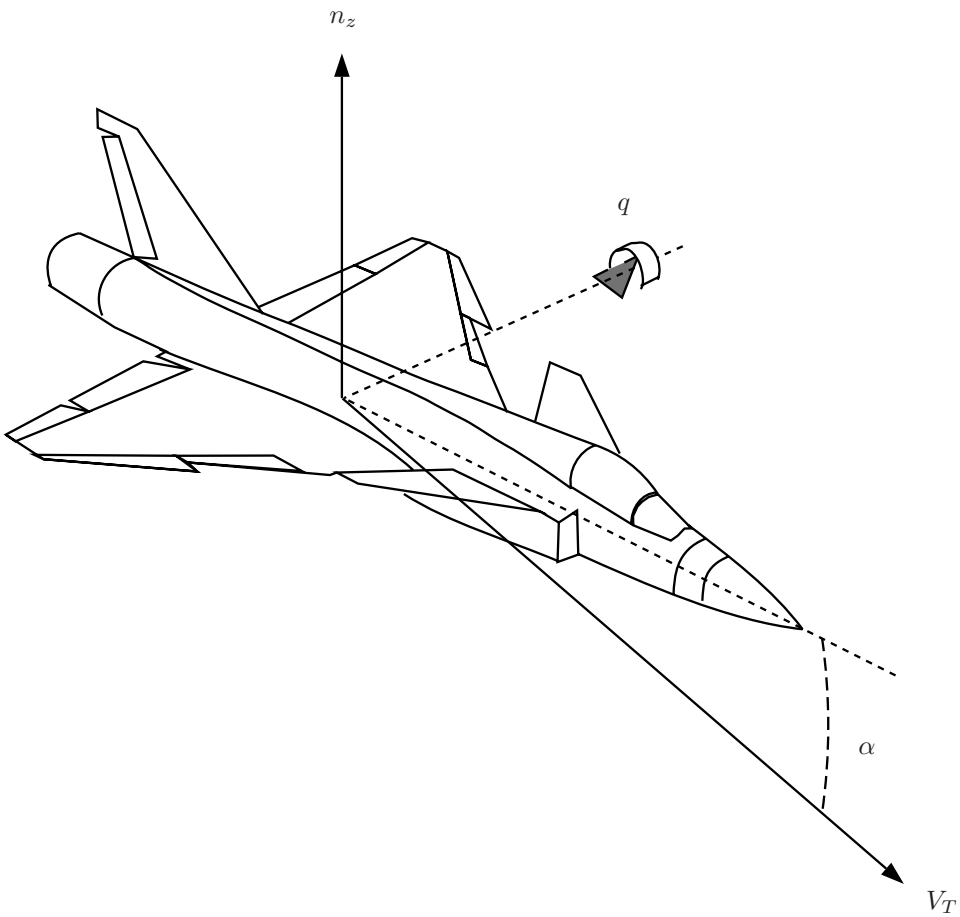


Figure 2.2 *Pitch control objectives.*

paves the way for various autopilot capabilities, e.g., altitude hold, relieving the pilot workload.

2.2 Control Objectives

Given the possibilities using a flight control system, what does the pilot want to control?

In a classical dogfight, whose importance is still recognized, maneuverability is the prime objective. Here, the *normal acceleration*, n_z, or the *pitch rate*, q, make up suitable controlled variables in the longitudinal direction, see Figure 2.2. n_z, also known as the *load factor*, is the acceleration experienced by the pilot directed

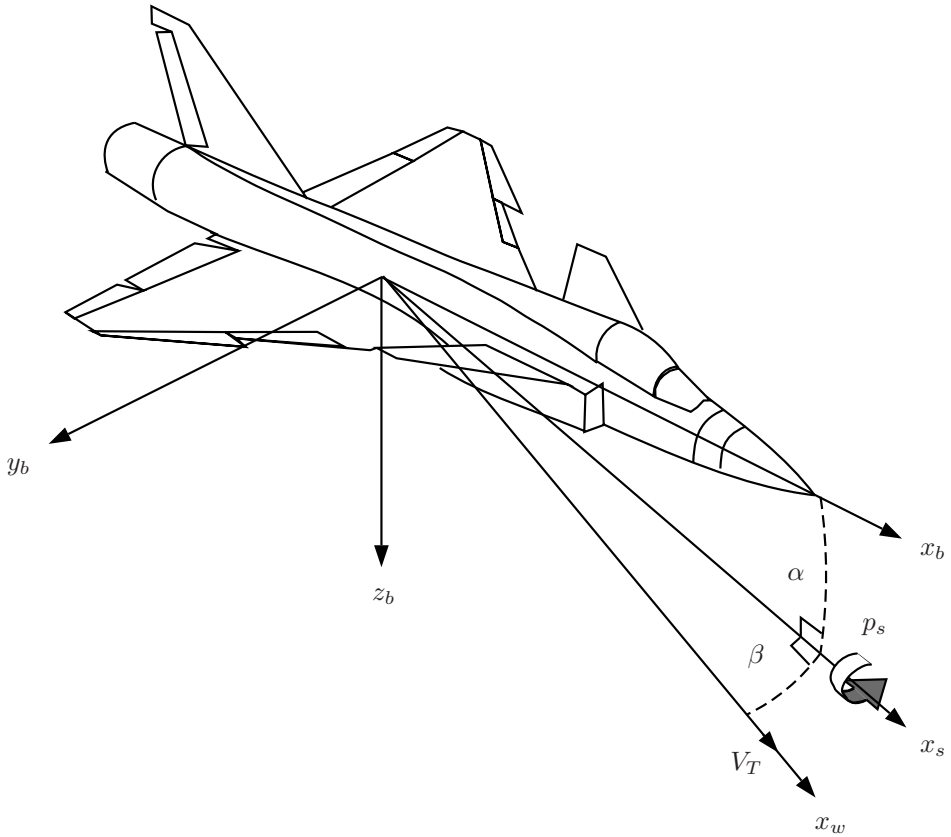


Figure 2.3 *Lateral control objectives and coordinate systems definitions. In the figure, α and β are both positive.*

along his spine. It is expressed as a multiple of the gravitational acceleration, g. n_z is closely coupled to the *angle of attack*, α, which appears naturally in the equations describing the aircraft dynamics, see Section 2.4. Therefore, angle of attack command control designs are also common, in particular for nonlinear approaches.

In the lateral direction, *roll rate* and *sideslip* command control systems are prevalent. The sideslip angle, β, is depicted in Figure 2.3. Typically, $\beta = 0$ is desired so that the aircraft is flying "straight into the wind" with a zero velocity component along the body y-axis, y_b. However, there are occasions where a certain sideslip is necessary, e.g., when landing the aircraft in the presence of side wind.

For the roll rate command system, a choice must be made regarding which axis to roll about, see Figure 2.3. Let us first consider the perhaps most obvious choice,

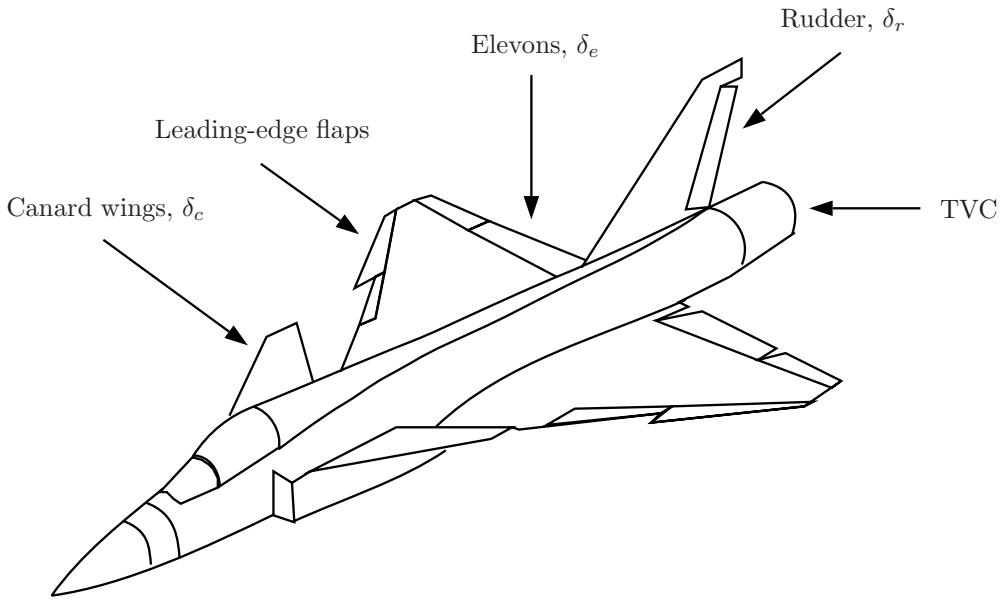


Figure 2.4 *A modern fighter aircraft configuration.*

namely the body x-axis, x_b. Considering a 90 degrees roll, we realize that the initial angle of attack will turn into pure sideslip at the end of the roll and vice versa. At high angles of attack this is not tolerable, since the largest acceptable amount of sideslip during a roll is in the order of 3–5 degrees [14]. To remove this effect, we could instead roll about the wind x-axis, x_w. Then α and β remain unchanged during a roll. This is known as a *velocity-vector roll*. With the usual assumption that a roll is performed at zero sideslip, this is equivalent to a *stability-axis roll*, performed about the stability x-axis, x_s. In this case, the angular velocity p_s is the variable to control.

There also exist situations where other control objectives are of interest. Autopilot functions like *altitude*, *heading*, and *speed* hold are vital to assist the pilot during long distance flight. For firing on-board weapons, the *orientation* of the aircraft is crucial. To benefit from the drag reduction that can be accomplished during close formation flight, the *position* of the wingman relative to the leader must be controlled precisely, preferably automatically to relieve the workload of the wingman pilot [26]. Also, to automate landing the aircraft it may be of interest to control its descent through the *flight path angle*, γ, see Figure 2.7.

2.3 Control Means

To accomplish the control tasks of the previous section, the aircraft must be equipped with actuators providing ways to control the different motions. Figure 2.4 shows a modern fighter aircraft configuration.

Pitch control, i.e., control of the longitudinal motion, is provided by deflecting the elevons and the canard wings symmetrically (right and left control surfaces deflect in the same direction). Conversely, roll control is provided by deflecting the elevons, and possibly also the canard wings, differentially (right and left control surfaces deflect in the opposite directions). Therefore, it is natural to introduce the control inputs

$$\delta_{es} = \frac{\delta_e^{left} + \delta_e^{right}}{2} \qquad \text{symmetrical elevon deflection}$$

$$\delta_{ed} = \frac{\delta_e^{left} - \delta_e^{right}}{2} \qquad \text{differential elevon deflection}$$

$$\delta_{cs} = \frac{\delta_c^{left} + \delta_c^{right}}{2} \qquad \text{symmetrical canard deflection}$$

$$\delta_{cd} = \frac{\delta_c^{left} - \delta_c^{right}}{2} \qquad \text{differential canard deflection}$$

Yaw control, i.e., control of the rotation about the body z-axis, is provided by the rudder. The leading-edge flaps can be used, e.g., to minimize the drag.

Recently, the interest in high angle of attack flight has led to the invention of thrust vectored control (TVC). Deflectable vanes are then mounted at the engine exhaust so that the engine thrust can be directed to produce a force in some desired direction. The NASA High Angle-of-Attack Research Vehicle (HARV) [30] uses this technology.

When convenient, we will let δ represent all the above control surface deflections. Finally, the aircraft speed, or rather the engine thrust force, is governed by the engine throttle setting.

2.4 Aircraft Dynamics

We now turn to the aircraft dynamics, and present the governing equations that tie the variables to be controlled to the control inputs available to us. The presentation, based on the books by Stevens and Lewis [74] and Boiffier [7], is focused on arriving at a model suitable for control design, consisting of a set of first order differential equations. For a deeper insight into the mechanics and aerodynamics behind the model, the reader is referred to the aforementioned books or, e.g., Etkin and Reid [16], McLean [54], or Nelson [57].

2.4.1 Governing physics

We will use the assumptions that Earth is flat and fixed, and that the aircraft body is rigid (as opposed to flexible). This yields a 6 degrees of freedom model (rotation and translation in 3 dimensions). The dynamics can be described by a state space model with 12 states consisting of

- $\mathbf{p} = \begin{pmatrix} p_N & p_E & h \end{pmatrix}^T$, the aircraft position expressed in an Earth-fixed coordinate system;
- $\mathbf{V} = \begin{pmatrix} u & v & w \end{pmatrix}^T$, the velocity vector expressed in the body-axis coordinate system;
- $\Phi = \begin{pmatrix} \phi & \theta & \psi \end{pmatrix}^T$, the Euler angles describing the orientation of the aircraft relative to the Earth-fixed coordinate system;
- $\omega = \begin{pmatrix} p & q & r \end{pmatrix}^T$, the angular velocity of the aircraft expressed in the body-axes coordinate system.

The task of controlling the aircraft position $\mathbf{p}$ is typically left entirely to the pilot, formation flight being a possible exception. The only coupling from $\mathbf{p}$ to the other state variables is through the altitude dependence of the aerodynamic pressure (2.4). Since the altitude varies slower than the rest of the variables, it can be regarded as a constant during the control design. Therefore the position dynamics will be left out here.

The equations governing the remaining three state vectors can be compactly written as

$$\mathbf{F} = m(\dot{\mathbf{V}} + \omega \times \mathbf{V}) \qquad \text{force equation} \qquad (2.1)$$

$$\mathbf{M} = I\dot{\omega} + \omega \times I\omega \qquad \text{moment equation} \qquad (2.2)$$

$$\dot{\Phi} = E(\Phi)\omega \qquad \text{attitude equation} \qquad (2.3)$$

where

$$E(\Phi) = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\theta/\cos\theta & \cos\phi/\cos\theta \end{pmatrix}$$

m is the aircraft mass and I is the aircraft inertial matrix. The force and moment equations follow from applying Newton's second law and the attitude equation spurs from the relation between the Earth-fixed and the body-fixed coordinate systems.

$\mathbf{F}$ and $\mathbf{M}$ represent the sum of the forces and moments, respectively, acting on the aircraft at the center of gravity. These forces and moments spring from three major sources, namely

- *gravity*,
- *engine thrust*, and
- *aerodynamic efforts*.

Introducing

$$\mathbf{F} = \mathbf{F}_G + \mathbf{F}_E + \mathbf{F}_A$$
$$\mathbf{M} = \mathbf{M}_E + \mathbf{M}_A$$

we will now investigate each of these components and express them in the body-fixed coordinate system.

Gravity

Gravity only gives a force contribution since it acts at the aircraft center of gravity. The gravitational force, mg, directed along the normal of the Earth plane, is considered constant over the altitude envelope. This yields

$$\mathbf{F}_G = mg \begin{pmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{pmatrix}$$

Engine thrust

The thrust force produced by the engine is denoted by F_T. Assuming the engine to be positioned so that the thrust acts parallel to the aircraft body x-axis (not using TVC) yields

$$\mathbf{F}_E = \begin{pmatrix} F_T \\ 0 \\ 0 \end{pmatrix}$$

Also assuming the engine to be mounted so that the thrust point lies in the body-axes xz-plane, offset from the center of gravity by Z_{TP} in the body-axes z-direction results in

$$\mathbf{M}_E = \begin{pmatrix} 0 \\ F_T Z_{TP} \\ 0 \end{pmatrix}$$

Aerodynamic efforts

The aerodynamic forces and moments, or the aerodynamic efforts for short, are due to the interaction between the aircraft body and the incoming airflow. The size and direction of the aerodynamic efforts are determined by the amount of air diverted by the aircraft in different directions (see [3] for an enlightening discussion on various explanations to aerodynamic lift). The amount of air diverted by the aircraft is mainly decided by

- the speed and density of the airflow (V_T, ρ);
- the geometry of the aircraft (δ, S, $\bar{c}$, b);
- the orientation of the aircraft relative to the airflow (α, β).

The aerodynamic efforts also depend on other variables, like the angular rates (p, q, r) and the time derivatives of the aerodynamic angles ($\dot{\alpha}$, $\dot{\beta}$), but these effects are not as prominent.

This motivates the standard way of modeling aerodynamic forces and moments:

$$\begin{aligned}\text{Force} &= \bar{q}SC_F(\delta, \alpha, \beta, p, q, r, \dot{\alpha}, \dot{\beta}, M, \dots)\\ \text{Moment} &= \bar{q}SlC_M(\delta, \alpha, \beta, p, q, r, \dot{\alpha}, \dot{\beta}, M, \dots)\end{aligned}$$

The aerodynamic pressure,

$$\bar{q} = \frac{1}{2}\rho(h)V_T^2 \tag{2.4}$$

captures the density dependence and most of the speed dependence, S is the aircraft wing area, and l refers to the length of the lever arm connected to the moment. C_F and C_M are known as *aerodynamic coefficients*. These are difficult to model analytically but can be estimated empirically through wind tunnel experiments and actual flight tests. Typically, each coefficient is written as the sum of several components, each capturing the dependence of one or more of the variables above. These components can be represented in several ways. A common approach is to store them in look-up tables and use interpolation to compute intermediate values. In other approaches one tries to fit the data to some parameterized function.

In the body-axes coordinate system, we have the expressions

$$\mathbf{F}_A = \begin{pmatrix} X \\ \bar{Y} \\ Z \end{pmatrix} \quad \text{where} \quad \begin{aligned} X &= \bar{q}SC_x \\ \bar{Y} &= \bar{q}SC_y \\ Z &= \bar{q}SC_z \end{aligned}$$

$$\mathbf{M}_A = \begin{pmatrix} \bar{L} \\ M \\ N \end{pmatrix} \quad \text{where} \quad \begin{aligned} \bar{L} &= \bar{q}SbC_l && \text{rolling moment} \\ M &= \bar{q}S\bar{c}C_m && \text{pitching moment} \\ N &= \bar{q}SbC_n && \text{yawing moment} \end{aligned}$$

These are illustrated in Figure 2.5. The aerodynamic forces are also commonly expressed in the wind-axes coordinate system (related to the body-fixed coordinate system as indicated in Figure 2.3) where we have that

$$\mathbf{F}_{A,w} = \begin{pmatrix} -D \\ Y \\ -L \end{pmatrix} \quad \text{where} \quad \begin{aligned} D &= \bar{q}SC_D && \text{drag force} \\ Y &= \bar{q}SC_Y && \text{side force} \\ L &= \bar{q}SC_L && \text{lift force} \end{aligned} \tag{2.5}$$

where the lift and side force coefficient, C_L and C_Y, mainly depend on α and β respectively.

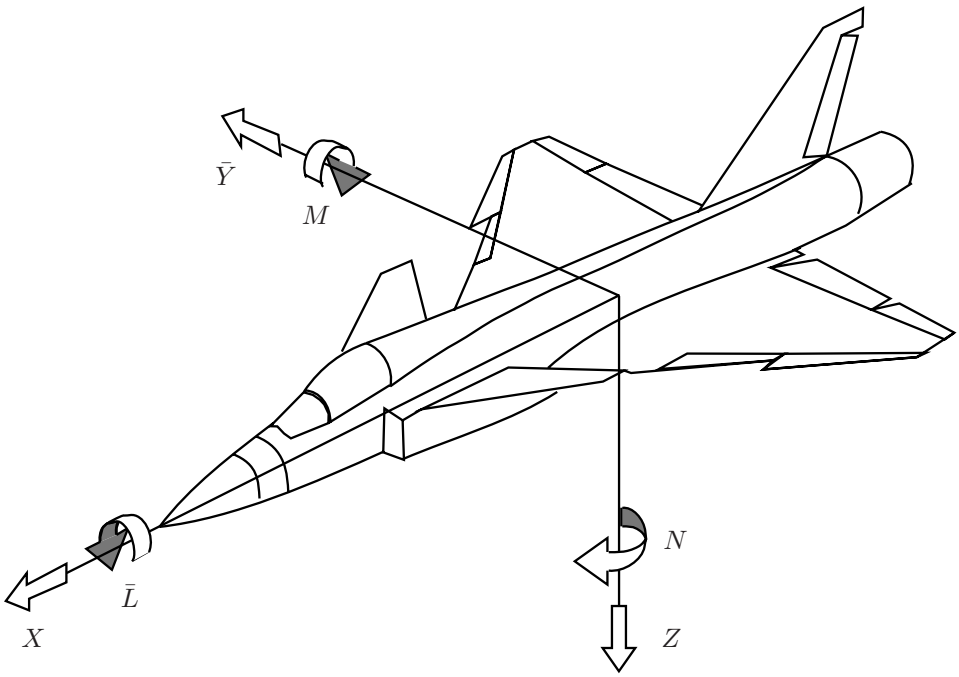


Figure 2.5 *Aerodynamics forces and moments in the body-axes coordinate system.*

Essentially, only the aerodynamic moments are affected when a control surface is deflected. *This is a key feature without which some nonlinear control design methods, including backstepping and dynamic inversion, would not be applicable.* Figure 2.6 shows the lift force and pitching moment coefficients, C_L and C_m, as functions of angle of attack and symmetrical elevon deflection. The aerodata comes from the HIRM model [56].

In Section 2.2, the normal acceleration n_z was introduced. We now have the setup to define n_z more precisely and find its relationship to α. We have that

$$n_z = -\frac{Z}{mg} = -\frac{\bar{q}SC_z(\delta, \alpha, \beta, \dots)}{mg}$$

Given an n_z command, the above equation can be used to solve for a corresponding α command.

2.4.2 Modeling for control

We will now collect the equations from the previous section and write the result in a form suitable for control design, namely as a system of first order scalar differential

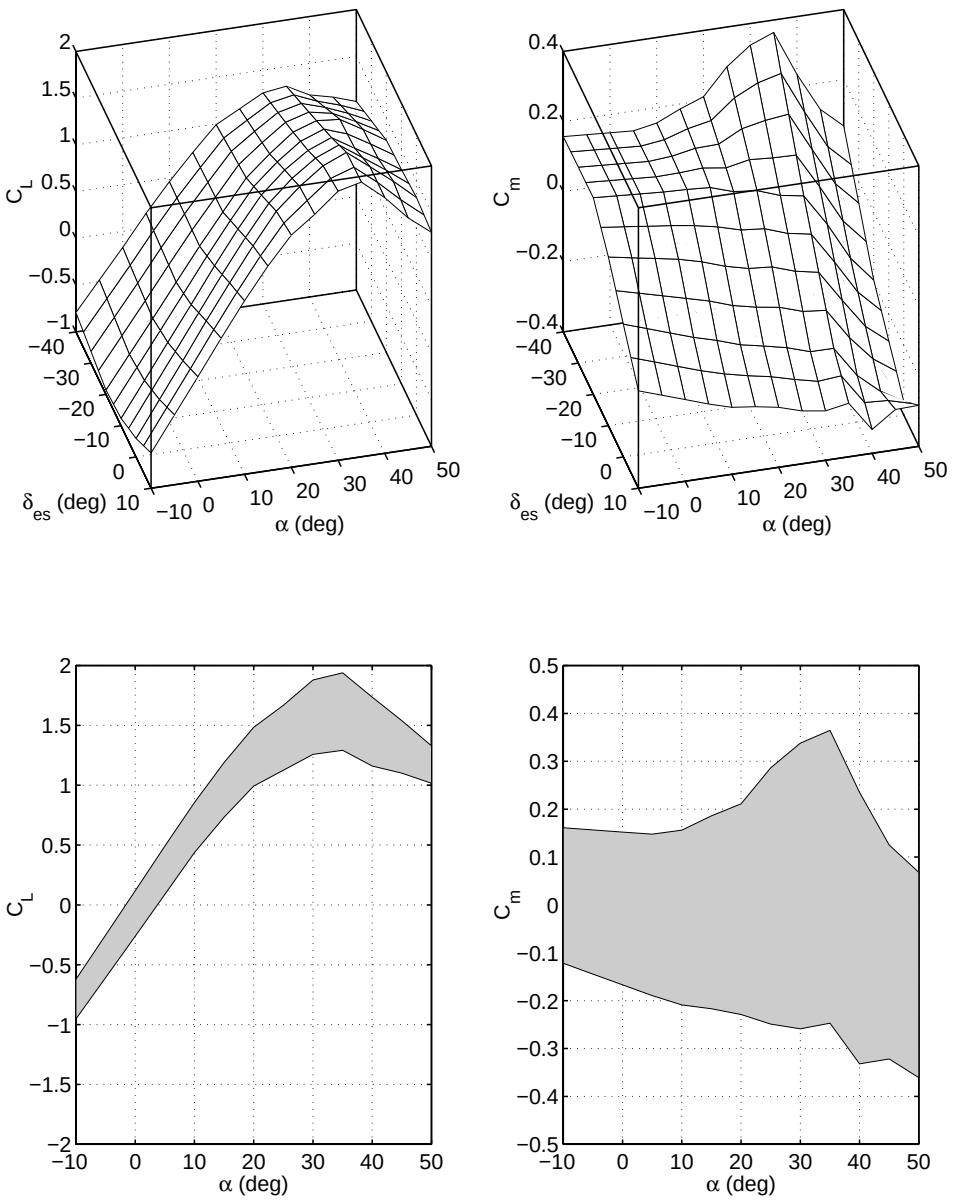


Figure 2.6 *The aerodynamic coefficients C_L and C_m as functions of α and δ_{es}. In the lower figures, the shaded area represents the dependence on δ_{es}. Apparently, elevon deflections primarily produce aerodynamic moments rather than forces.*

equations. Expanding Equations (2.1)–(2.3) yields

Force equations (body-axes)

$$\dot{u} = rv - qw - g\sin\theta + \frac{1}{m}(X + F_T)$$
$$\dot{v} = pw - ru + g\sin\phi\cos\theta + \frac{1}{m}\bar{Y}$$
$$\dot{w} = qu - pv + g\cos\phi\cos\theta + \frac{1}{m}Z$$

Moment equations (body-axes)

$$\dot{p} = (c_1 r + c_2 p)q + c_3\bar{L} + c_4 N \tag{2.6a}$$
$$\dot{q} = c_5 pr - c_6(p^2 - r^2) + c_7(M + F_T Z_{TP}) \tag{2.6b}$$
$$\dot{r} = (c_8 p - c_2 r)q + c_4\bar{L} + c_9 N \tag{2.6c}$$

Attitude equations (body-axes)

$$\dot{\phi} = p + \tan\theta(q\sin\phi + r\cos\theta)$$
$$\dot{\theta} = q\cos\phi - r\sin\phi$$
$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta}$$

Here we have introduced

$$\Gamma c_1 = (I_y - I_z)I_z - I_{xz}^2 \qquad \Gamma c_2 = (I_x - I_y + I_z)I_{xz} \qquad \Gamma c_3 = I_z$$
$$\Gamma c_4 = I_{xz} \qquad c_5 = \frac{I_z - I_x}{I_y} \qquad c_6 = \frac{I_{xz}}{I_y}$$
$$c_7 = \frac{1}{I_y} \qquad \Gamma c_8 = I_x(I_x - I_y) + I_{xz}^2 \qquad \Gamma c_9 = I_x$$

where

$$I = \begin{pmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{xz} & 0 & I_z \end{pmatrix}, \quad \Gamma = I_x I_z - I_{xz}^2$$

In Section 2.2, α, β, and V_T were mentioned as suitable controlled variables. We can rewrite the force equations in terms of these variables by performing the

following change of variables:

$$V_T = \sqrt{u^2 + v^2 + w^2}$$
$$\alpha = \arctan \frac{w}{u}$$
$$\beta = \arcsin \frac{v}{V_T}$$

This gives us

Force equations (wind-axes)

$$\dot{V}_T = \frac{1}{m}(-D + F_T \cos\alpha \cos\beta + mg_1) \tag{2.7a}$$
$$\dot{\alpha} = q - (p\cos\alpha + r\sin\alpha)\tan\beta + \frac{1}{mV_T\cos\beta}(-L - F_T \sin\alpha + mg_2) \tag{2.7b}$$
$$\dot{\beta} = p\sin\alpha - r\cos\alpha + \frac{1}{mV_T}(Y - F_T \cos\alpha \sin\beta + mg_3) \tag{2.7c}$$

where the contributions due to gravity are given by

$$\begin{aligned} g_1 &= g(-\cos\alpha\cos\beta\sin\theta + \sin\beta\cos\theta\sin\phi + \sin\alpha\cos\beta\cos\theta\cos\phi) \\ g_2 &= g(\cos\alpha\cos\theta\cos\phi + \sin\alpha\sin\theta) \\ g_3 &= g(\cos\beta\cos\theta\sin\phi + \sin\beta\cos\alpha\sin\theta - \sin\alpha\sin\beta\cos\theta\cos\phi) \end{aligned} \tag{2.8}$$

See Appendix 2.A for a complete derivation. A pleasant effect of this reformulation is that the nonlinear aerodynamic forces L and Y mainly depend on α and β, respectively. This fact will be used for control design using backstepping.

A very common approach to flight control design is to control longitudinal motions (motions in the body xz-plane) and lateral motions (all other motions) separately. With no lateral motions, the longitudinal equations of motion become

Longitudinal equations

$$\dot{V}_T = \frac{1}{m}(-D + F_T \cos\alpha - mg\sin\gamma) \tag{2.9a}$$
$$\dot{\alpha} = q + \frac{1}{mV_T}(-L - F_T \sin\alpha + mg\cos\gamma) \tag{2.9b}$$
$$\dot{\theta} = q \tag{2.9c}$$
$$\dot{q} = \frac{1}{I_y}(M + F_T Z_{TP}) \tag{2.9d}$$

Here, $\gamma = \theta - \alpha$ is the flight path angle determining the direction of the velocity vector, as depicted in Figure 2.7.

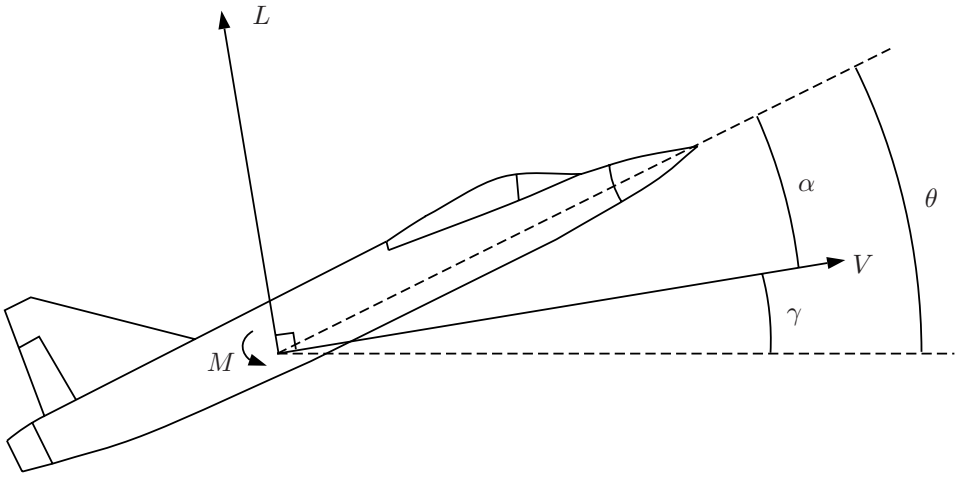


Figure 2.7 *Illustration of the longitudinal aircraft entities.*

2.5 Current Approaches to Flight Control Design

In this section we survey some of the proposed design schemes with the emphasis on nonlinear control designs. Flight control design surveys can also be found in Magni et al. [53] and Huang and Knowles [35].

2.5.1 Gain-scheduling

The prevailing flight control design methodology of today is based on gain-scheduling. The flight envelope that one wants to conquer is partitioned into smaller regions. For each region, a steady state operating point is chosen around which the dynamics are linearized. Linear control tools can then be used to design one control law for each operating point. Between the operating points, the controllers are blended together using gain-scheduling to make the transitions between different regions smooth and transparent to the pilot.

Dynamic pressure or Mach number and altitude together are typical scheduling variables. Note that if the closed loop behavior is designed to be the same throughout the envelope, gain-scheduling can be seen as a way of cancelling the nonlinear behavior due to variations in the scheduling variables. However, the nonlinear effects due to high angles of attack or high angular rates are not dealt with. Whether or not closed loop stability holds when the aircraft state is far from the operating point at which the linearization was performed, depends on the robustness of the control design.

Let us summarize the pros and cons of gain-scheduling.

+ Using several linear models to describe the aircraft dynamics allows the control designer to utilize all the classical tools for control design and robustness and disturbance analysis.

+ The methodology has proven to work well in practice. The Swedish fighter JAS 39 Gripen [63] is a "flying proof" of its success.

– The outlined divide-and-conquer approach is rather tedious since for each region, a controller must be designed. The number of regions may be over 50.

– Only the nonlinear system behavior in speed and altitude is considered. Stability is therefore guaranteed only for low angles of attack and low angular rates.

2.5.2 Dynamic inversion (feedback linearization)

The idea behind gain-scheduling was to provide the pilot with a similar aircraft response irrespectively of the aircraft speed and altitude. This philosophy is even more pronounced in nonlinear dynamic inversion (NDI), which is the term used in the aircraft community for what is known as feedback linearization in control theory. In this thesis, we only deal with feedback linearization through examples and intuitive explanations. For an introduction to feedback linearization theory, the reader is referred to, e.g., Slotine and Li [70] or Isidori [36].

Using dynamic inversion, as the name implies, the natural aircraft dynamics are "inverted" and replaced by the desired linear ones through the wonders of feedback. This includes the nonlinear behavior in speed and altitude as well as the nonlinear effects at high angles of attack and high angular rates.

To make things more concrete, consider the simplified angle of attack dynamics (cf. (2.9b), (2.9d))

$$\dot{\alpha} = q - \frac{1}{mV_T} L(\alpha) \tag{2.10a}$$

$$\dot{q} = \frac{1}{I_y} M(\delta, \alpha, q) \tag{2.10b}$$

The speed is assumed to vary much slower than α and q so that $\dot{V}_T \approx 0$ is a good approximation. Now, introduce

$$z = q - \frac{1}{mV_T} L(\alpha) \tag{2.11}$$

and compute $\dot{z}$:

$$\dot{z} = \frac{1}{I_y} M(\delta, \alpha, q) - \frac{1}{mV_T} L'(\alpha) z$$

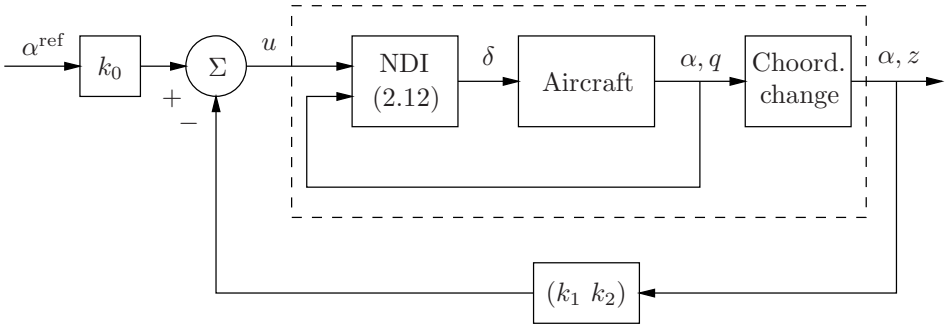


Figure 2.8 *Illustration of a dynamic inversion control law. The inner feedback loop cancels the nonlinear dynamics making the dashed box a linear system, which is controlled by the outer feedback loop.*

Introducing

$$u = \frac{1}{I_y} M(\delta, \alpha, q) - \frac{1}{mV_T} L'(\alpha) z \tag{2.12}$$

we can now rewrite (2.10) as a chain of two integrators:

$$\dot{\alpha} = z$$
$$\dot{z} = u$$

Through this transformation, we have cast a nonlinear control problem onto a linear one. Again, linear control tools can be used to find a control law

$$u = k_0 \alpha^{\text{ref}} - \begin{pmatrix} k_1 & k_2 \end{pmatrix} \begin{pmatrix} \alpha \\ z \end{pmatrix} \tag{2.13}$$

giving a satisfactory response from the commanded value α^{ref} to α. Using (2.11) and (2.12) we can solve for the true control input δ, given that (2.12) is invertible w.r.t. δ. Invertability is lost, e.g., when the control surfaces saturate.

Equation (2.12) can be interpreted as an inner control loop cancelling the nonlinear behavior while (2.13) is an outer control loop providing the system with the desired closed loop characteristics, see Figure 2.8.

There exists a large number of publications devoted to flight control design using feedback linearization. The 1988 paper of Lane and Stengel [48] is an early contribution on the subject containing designs for various control objectives.

Enns et al. [15] outline dynamic inversion as the successor to gain-scheduling as the prevailing design paradigm. This bold statement is supported by discussions regarding the ability to deal with, e.g., handling quality specifications, disturbance attenuation, robustness, and control surface allocation in a dynamic inversion framework.

Robustness analysis has often been pointed out as the Achilles' heel of dynamic inversion. Dynamic inversion relies on the complete knowledge of the nonlinear plant dynamics. This includes knowledge of the aerodynamic efforts, which in practice comes with an uncertainty in the order of 10%. What happens if the true nonlinearities are not completely cancelled by the controller? [78] contains some results regarding this issue.

One way of enhancing the robustness is to reduce the control law dependence on the aerodynamic coefficients. Note that computing δ from (2.12) requires knowledge of the aerodynamic coefficients C_m and C_L as well as $dC_L/d\alpha$ (recall from (2.5) that $L = \bar{q}SC_L$). Snell et al. [71] propose a dynamic inversion design which does not involve $dC_L/d\alpha$, thus making the design more robust. The idea is to use time-scale separation and design separate controllers for the α- and q-subsystems of (2.10). Inspired by singular perturbation theory [42] and cascaded control design, the system is considered to consist of slow dynamics (2.10a) and fast dynamics (2.10b). First, the slow dynamics are controlled. Assume the desired slow dynamics to be

$$\dot{\alpha} = -k_1(\alpha - \alpha^{\text{ref}}), \quad k_1 > 0$$

Then the angle of attack command α^{ref} can be mapped onto a pitch rate command

$$q^{\text{ref}} = -k_1(\alpha - \alpha^{\text{ref}}) + \frac{1}{mV_T}L(\alpha) \tag{2.14}$$

We now turn to the fast dynamics and determine a control law rendering the fast dynamics

$$\dot{q} = -k_2(q - q^{\text{ref}}), \quad k_2 > 0$$

This can be achieved by solving

$$\frac{1}{I_y}M(\delta, \alpha, q) = -k_2(q - q^{\text{ref}}) = -k_2(q + k_1(\alpha - \alpha^{\text{ref}}) - \frac{1}{mV_T}L(\alpha)) \tag{2.15}$$

for δ, which obviously only requires knowledge of C_m and C_L. The controller structure is shown in Figure 2.9.

The remaining question to be answered is whether this time-scale separation approach indeed is valid – is the closed loop system guaranteed to be stable? Snell et al. [71] use linear pole placement arguments to confirm stability. Interestingly enough, in Section 5.1.3 we will show that the above design is a special case of a general backstepping design, and stability can be shown using Lyapunov theory. We also show that using backstepping, the knowledge of the aerodynamic coefficients required to guarantee stability can be further reduced.

Further efforts to robustify a dynamic inversion design against model errors can be found in Adams et al. [1] and Reiner et al. [60, 61]. Here, the idea is to enhance robustness by using a μ synthesis controller in the outer, linear loop in Figure 2.8.

Let us summarize the pros and cons of dynamic inversion.

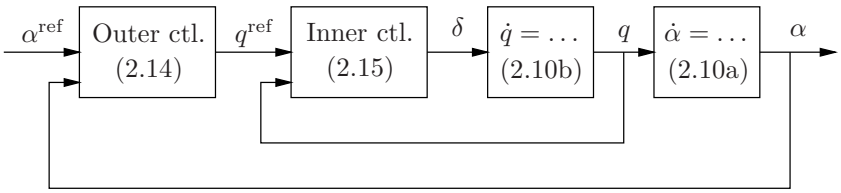


Figure 2.9 *Cascaded dynamic inversion control design based on time-scale separation.*

+ One single controller is used throughout the whole flight regime.
+ Stability is guaranteed even for high angles of attack, provided that the model is accurate at those angles.
+ Closed loop performance can be tuned using linear tools.
– Dynamic inversion relies on precise knowledge of the aerodynamic coefficients to completely cancel the nonlinear dynamics.

2.5.3 Other nonlinear approaches

In addition to dynamic inversion, many other nonlinear approaches have been applied to flight control design. Garrard et al. [23] formulate the angle of attack control problem as a linear quadratic optimization problem. As their aircraft model is nonlinear, in order to capture the behavior at high angles of attack, the arising Hamilton-Jacobi-Bellman equation is difficult to solve exactly. The authors settle for a truncated solution to the HJB equation.

Mudge and Patton [55] consider the problem of pitch pointing, where the objective is to command the pitch angle θ while the flight path angle γ remains unchanged. Eigenstructure assignment is used to achieve the desired decoupling and sliding mode behavior is added for enhanced robustness.

Other approaches deal with the problem of tracking a reference signal, whose future values are also known. Lévine [50] shows that an aircraft is differentially flat if the outputs are properly chosen. This is used to design an autopilot for making the aircraft follow a given trajectory.

Hauser and Jadbabaie [31] design receding horizon control laws for unmanned combat aerial vehicles performing aggressive maneuvers. Over a receding horizon, the aircraft trajectory following properties are optimized on-line. The control laws are implemented and evaluated using the ducted fan at Caltech.

Appendix

2.A Wind-axes Force Equations

This appendix contains the details of the conversion of the aircraft force equation from the body-axes to the wind-axes coordinate system. The result is a standard one, but the derivation, which establishes the relationship between the forces used in the two different representations, is rarely found in textbooks on flight control.

The body-axes force equations are

$$\begin{aligned}
\dot{u} &= rv - qw - g\sin\theta + \frac{1}{m}(X + F_T) \\
\dot{v} &= pw - ru + g\sin\phi\cos\theta + \frac{1}{m}\bar{Y} \\
\dot{w} &= qu - pv + g\cos\phi\cos\theta + \frac{1}{m}Z
\end{aligned}$$

The relation between the variables used in the two coordinate systems is given by

$$\begin{aligned}
\alpha &= \arctan\frac{w}{u} \\
\beta &= \arcsin\frac{v}{V_T} \\
V_T &= \sqrt{u^2 + v^2 + w^2}
\end{aligned}
\qquad \Longleftrightarrow \qquad
\begin{aligned}
u &= V_T\cos\alpha\cos\beta \\
v &= V_T\sin\beta \\
w &= V_T\sin\alpha\cos\beta
\end{aligned}$$

Differentiating we get

$$\begin{aligned}
\dot{V}_T = \frac{u\dot{u} + v\dot{v} + w\dot{w}}{V_T} &= \frac{1}{V_T}\big(u(rv - qw - g\sin\theta + \frac{1}{m}(X + F_T)) \\
&\quad + v(pw - ru + g\sin\phi\cos\theta + \frac{1}{m}\bar{Y}) + w(qu - pv + g\cos\phi\cos\theta + \frac{1}{m}Z)\big) \\
&= \underbrace{g(-\cos\alpha\cos\beta\sin\theta + \sin\beta\sin\phi\cos\theta + \sin\alpha\cos\beta\cos\phi\cos\theta)}_{g_1} \\
&\quad + \frac{1}{m}(F_T\cos\alpha\cos\beta + \underbrace{X\cos\alpha\cos\beta + \bar{Y}\sin\beta + Z\sin\alpha\cos\beta}_{-D})
\end{aligned}$$

$$\begin{aligned}
\dot{\alpha} &= \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} = \frac{V_T \cos\beta(\dot{w}\cos\alpha - \dot{u}\sin\alpha)}{V_T^2 \cos^2\beta} \\
&= \frac{1}{V_T \cos\beta}\big((qV_T\cos\alpha\cos\beta - pV_T\sin\beta + g\cos\phi\cos\theta + \frac{1}{m}Z)\cos\alpha \\
&\quad - (rV_T\sin\beta - qV_T\sin\alpha\cos\beta - g\sin\theta + \frac{1}{m}(X + F_T))\sin\alpha\big) \\
&= q - \tan\beta(p\cos\alpha + r\sin\alpha) \\
&\quad + \frac{1}{mV_T\cos\beta}(m\underbrace{g(\cos\alpha\cos\phi\cos\theta - \sin\alpha\sin\theta)}_{g_2} - F_T\sin\alpha + \underbrace{Z\cos\alpha - X\sin\alpha}_{-L})
\end{aligned}$$

$$\begin{aligned}
\dot{\beta} &= \frac{\dot{v}V_T - v\dot{V}_T}{V_T^2\cos\beta} = \frac{\dot{v}(u^2 + w^2) - v(u\dot{u} + w\dot{w})}{V_T^3\cos\beta} \\
&= \frac{\dot{v}V_T^2\cos^2\beta - V_T^2\sin\beta\cos\beta(\dot{u}\cos\alpha + \dot{w}\sin\alpha)}{V_T^3\cos\beta} \\
&= \frac{1}{V_T}((pw - ru + g\sin\phi\cos\theta + \frac{1}{m}\bar{Y})\cos\beta \\
&\quad - (rv - qw - g\sin\theta + \frac{1}{m}(X + F_T))\cos\alpha\sin\beta \\
&\quad - (qu - pv + g\cos\phi\cos\theta + \frac{1}{m}Z)\sin\alpha\sin\beta) \\
&= p(\underbrace{\sin\alpha\cos^2\beta + \sin\alpha\sin^2\beta}_{\sin\alpha}) - r(\underbrace{\cos\alpha\cos^2\beta + \cos\alpha\sin^2\beta}_{\cos\alpha}) \\
&\quad + \frac{1}{mV_T}(m\underbrace{g(\cos\beta\sin\phi\cos\theta + \cos\alpha\sin\beta\sin\theta - \sin\alpha\sin\beta\cos\phi\cos\theta)}_{g_3} \\
&\quad - F_T\cos\alpha\sin\beta\underbrace{-X\cos\alpha\sin\beta + \bar{Y}\cos\beta - Z\sin\alpha\sin\beta}_{Y})
\end{aligned}$$

Summing up, we have the following transformed system of equations.

$$\begin{aligned}
\dot{V}_T &= \frac{1}{m}(-D + F_T\cos\alpha\cos\beta + mg_1) \\
\dot{\alpha} &= q - \tan\beta(p\cos\alpha + r\sin\alpha) + \frac{1}{mV_T\cos\beta}(-L - F_T\sin\alpha + mg_2) \\
\dot{\beta} &= p\sin\alpha - r\cos\alpha + \frac{1}{mV_T}(Y - F_T\cos\alpha\sin\beta + mg_3)
\end{aligned}$$

The orientation dependent gravitational components are

$$\begin{aligned}
g_1 &= g(-\cos\alpha\cos\beta\sin\theta + \sin\beta\cos\theta\sin\phi + \sin\alpha\cos\beta\cos\theta\cos\phi) \\
g_2 &= g(\cos\alpha\cos\theta\cos\phi + \sin\alpha\sin\theta) \\
g_3 &= g(\cos\beta\cos\theta\sin\phi + \sin\beta\cos\alpha\sin\theta - \sin\alpha\sin\beta\cos\theta\cos\phi)
\end{aligned}$$

The relationship between the aerodynamic forces expressed in the two coordinate systems is given by

$$\begin{aligned} D &= -X\cos\alpha\cos\beta - \bar{Y}\sin\beta - Z\sin\alpha\cos\beta \\ L &= X\sin\alpha - Z\cos\alpha \\ Y &= -X\cos\alpha\sin\beta + \bar{Y}\cos\beta - Z\sin\alpha\sin\beta \end{aligned}$$

These equations are relevant since often the available aerodata relates to the body-axes system while it is the wind-axes forces that appear in the control laws.

3

Backstepping

Lyapunov theory has for a long time been an important tool in linear as well as nonlinear control. However, its use within nonlinear control has been hampered by the difficulties to find a Lyapunov function for a given system. If one can be found, the system is known to be stable, but the task of finding such a function has often been left to the imagination and experience of the designer.

The invention of *constructive* tools for nonlinear control design based on Lyapunov theory, like backstepping and forwarding, has therefore been received with open arms by the control community. Here, a control law stabilizing the system is derived along with a Lyapunov function to prove the stability.

In this chapter, backstepping is presented with the focus on designing state feedback laws. Sections 3.1 and 3.2 contain mathematical preliminaries. Section 3.3 is the core of the chapter where the main backstepping result is presented along with a discussion on the class of systems to which it applies and which choices are left to the designer. In Section 3.4, some related design methods based on Lyapunov theory are outlined, and in Section 3.5 we survey applications to which backstepping has been applied.

3.1 Lyapunov Theory

A basic requirement on a controlled system is that it should end up at the desired equilibrium without taking a too big detour getting there. Let us formalize this requirement in terms of the properties of the desired equilibrium, following Slotine and Li [70].

Definition 3.1 (Lyapunov stability)
Consider the time-invariant system

$$\dot{x} = f(x) \tag{3.1}$$

starting at the initial state $x(0)$. Assume x_e to be an equilibrium point of the system, i.e., $f(x_e) = 0$. We say that the equilibrium point is

- *stable, if for each $\epsilon > 0$ there exists $\delta(\epsilon) > 0$ such that*

$$\|x(0) - x_e\| < \delta \Rightarrow \|x(t) - x_e\| < \epsilon \text{ for all } t \geq 0$$

- *unstable, if it is not stable;*
- *asymptotically stable, if it is stable and in addition there exists $r > 0$ such that*

$$\|x(0) - x_e\| < r \Rightarrow x(t) \rightarrow x_e \text{ as } t \rightarrow \infty$$

- *globally asymptotically stable (GAS), if it is asymptotically stable for all initial states.*

□

Global asymptotic stability (GAS) is what we will strive towards in our control design. We will refer to a control law that yields GAS as *globally stabilizing*, dropping the word "asymptotically" for convenience.

How can we show a certain type of stability? The definitions above involve $x(t)$, the explicit solution to (3.1), which in general cannot be found analytically. Fortunately there are other ways of proving stability.

The Russian mathematician and engineer A. M. Lyapunov came up with the idea of condensing the state vector $x(t)$ into a scalar function $V(x)$, measuring how far from the equilibrium the system is. $V(x)$ can be thought of as representing the energy contained in the system. If $V(x)$ can be shown to continuously decrease, then the system itself must be moving towards the equilibrium.

This approach to showing stability is called Lyapunov's direct method (or second method) and can be found in any introductory textbook on nonlinear control such as those by Slotine and Li [70], Khalil [40], Isidori [36], and Vidyasagar [76]. Lyapunov's original work can be found in [52]. Let us first introduce some useful definitions.

Definition 3.2
A scalar function $V(x)$ is said to be

- *positive definite if $V(0) = 0$ and*

$$V(x) > 0, \quad x \neq 0$$

- *positive semidefinite if $V(0) = 0$ and*

$$V(x) \geq 0, \quad x \neq 0$$

- *negative (semi-)definite if $-V(x)$ is positive (semi-)definite*
- *radially unbounded if*

$$V(x) \to \infty \text{ as } \|x\| \to \infty$$

□

We now state our main theorem for proving stability.

Theorem 3.1 (LaSalle-Yoshizawa)
Let $x = 0$ be an equilibrium point for (3.1). Let $V(x)$ be a scalar, continously differentiable function of the state x such that

- *$V(x)$ is positive definite*
- *$V(x)$ is radially unbounded*
- *$\dot{V}(x) = V_x(x)f(x) \leq -W(x)$ where $W(x)$ is positive semidefinite*

Then, all solutions of (3.1) satisfy

$$\lim_{t\to\infty} W(x(t)) = 0$$

In addition, if $W(x)$ is positive definite, then the equilibrium $x = 0$ is GAS.

Proof See Krstić et al. [46]. □

For showing stability when $\dot{V}(x)$ is only negative semidefinite, the following corollary due to LaSalle is useful.

Corollary 3.1
Let $x = 0$ be the only equilibrium point for (3.1). Let $V(x)$ be a scalar, continously differentiable function of the state x such that

- *$V(x)$ is positive definite*
- *$V(x)$ is radially unbounded*

- $\dot{V}(x)$ *is negative semidefinite*

Let $E = \{x : \dot{V}(x) = 0\}$ *and suppose that no other solution than* $x(t) \equiv 0$ *can stay forever in* E*. Then* $x = 0$ *is GAS.*

Proof See Krstić et al. [46]. □

Note that these results are non-constructive, in the sense that they give no clue about how to find a V satisfying the conditions necessary to conclude GAS. We will refer to a function $V(x)$ satisfying the itemized conditions in Theorem 3.1 as a Lyapunov function for the system.

3.2 Lyapunov Based Control Design

Let us now add a control input and consider the system

$$\dot{x} = f(x, u) \tag{3.2}$$

Given the stability results from the previous section, it would be nice if we could find a control law

$$u = k(x)$$

so that the desired state of the closed loop system

$$\dot{x} = f(x, k(x))$$

becomes a globally asymptotically stable equilibrium point. For simplicity, we will assume the origin to be the goal state. This can always be achieved through a suitable change of coordinates.

A straightforward approach to finding $k(x)$ is to construct a positive definite, radially unbounded function $V(x)$ and then choose $k(x)$ such that

$$\dot{V} = V_x(x)f(x, k(x)) = -W(x) \tag{3.3}$$

where $W(x)$ is positive definite. Then closed loop stability follows from Theorem 3.1. For this approach to succeed, V and W must have been carefully selected, or (3.3) will not be solvable. This motivates the following definition:

Definition 3.3 (Control Lyapunov function (clf))
A smooth, positive definite, radially unbounded function $V(x)$ *is called a control Lyapunov function (clf) for* (3.2) *if for all* $x \neq 0$*,*

$$\dot{V} = V_x(x)f(x, u) < 0 \text{ for some } u$$

□

Given a clf for the system, we can thus find a globally stabilizing control law. In fact, the existence of a globally stabilizing control law is *equivalent* to the existence of a clf. This means that for each globally stabilizing control law, a corresponding clf can be found, and vice versa. This is known as Artstein's theorem [4].

If a clf is known, a particular choice of $k(x)$ is given by Sontag's formula [72] reproduced in (3.5). For a system which is affine in the control input

$$\dot{x} = f(x) + g(x)u \tag{3.4}$$

we can select

$$u = k(x) = -\frac{a + \sqrt{a^2 + b^2}}{b} \tag{3.5}$$

where

$$a = V_x(x)f(x)$$
$$b = V_x(x)g(x)$$

This yields

$$\dot{V} = V_x(x)\big(f(x) + g(x)u\big) = a + b\Big(-\frac{a + \sqrt{a^2 + b^2}}{b}\Big) = -\sqrt{a^2 + b^2} \tag{3.6}$$

and thus renders the origin GAS.

A closely related approach is the one by Freeman and Primbs [22] where u is chosen to minimize the control effort necessary to satisfy

$$\dot{V} \leq -W(x)$$

for some W. The minimization is carried out pointwise in time (and not over some horizon). Using an inequality constraint rather than asking for equality (as in (3.6)) makes it possible to benefit from the system's inherent stability properties. If $f(x)$ alone drives the system (3.4) towards the equilibrium such that

$$\dot{V}|_{u=0} = V_x(x)f(x) < -W(x)$$

it would be a waste of control effort to achieve $\dot{V} = -W(x)$.

3.3 Backstepping

The control designs of the previous section rely on the knowledge of a control Lyapunov function for the system. But how do we find such a function?

Backstepping answers this question in a recursive manner for a certain class of nonlinear systems which show a lower triangular structure. We will first state the main result and then deal with user related issues like which systems can be handled using backstepping, which design choices there are and how they affect the resulting control law.

3.3.1 Main result

This result is standard today and can be found in, e.g, Sepulchre et al. [66] or Krstić et al. [46].

Proposition 3.1 (Backstepping)
Consider the system

$$\dot{x} = f(x, \xi) \tag{3.7a}$$

$$\dot{\xi} = u \tag{3.7b}$$

where $x \in \mathbb{R}^n$, $\xi \in \mathbb{R}$ are state variables and $u \in \mathbb{R}$ is the control input.

Assume that for the subsystem (3.7a), *a virtual control law*

$$\xi = \xi^{des}(x) \tag{3.8}$$

is known such that 0 is a GAS equilibrium of (3.7a). *Let $W(x)$ be a clf for* (3.7a) *such that*

$$\dot{W}|_{\xi=\xi^{des}} = W_x(x) f(x, \xi^{des}(x)) < 0, \ x \neq 0$$

Then, a clf for the augmented system (3.7) *is given by*

$$V(x, \xi) = W(x) + \frac{1}{2}\big(\xi - \xi^{des}(x)\big)^2 \tag{3.9}$$

Moreover, a globally stabilizing control law, satisfying

$$\dot{V} = W_x(x) f(x, \xi^{des}(x)) - \big(\xi - \xi^{des}(x)\big)^2$$

is given by

$$u = \frac{\partial \xi^{des}(x)}{\partial x} f(x, \xi) - W_x(x) \frac{f(x, \xi) - f(x, \xi^{des}(x))}{\xi - \xi^{des}(x)} + \xi^{des}(x) - \xi \tag{3.10}$$

Before presenting the proof, it is worth pointing out that (3.10) is neither the only nor necessarily the best globally stabilizing control law for (3.7). The value of the proposition is that it shows the *existence* of at least one globally stabilizing control law for this type of augmented systems.

Proof We will conduct the proof in a constructive manner to clarify which design choices that can be made during the control law construction.

The key idea is to use the fact that we know how to stabilize the subsystem (3.7a) if we were able to control ξ directly, namely by using (3.8). Therefore, introduce the residual

$$\tilde{\xi} = \xi - \xi^{\text{des}}(x)$$

By forcing $\tilde{\xi}$ to zero, ξ will tend to the desired value ξ^{des} and the entire system will be stabilized.

In terms of $\tilde{\xi}$, the system dynamics (3.7) become

$$\dot{x} = f(x, \tilde{\xi} + \xi^{\text{des}}(x)) \triangleq f(x, \xi^{\text{des}}) + \psi(x, \tilde{\xi})\tilde{\xi} \tag{3.11a}$$

$$\dot{\tilde{\xi}} = u - \frac{\partial \xi^{\text{des}}(x)}{\partial x} f(x, \tilde{\xi} + \xi^{\text{des}}(x)) \tag{3.11b}$$

where

$$\psi(x, \tilde{\xi}) = \frac{f(x, \tilde{\xi} + \xi^{\text{des}}(x)) - f(x, \xi^{\text{des}}(x))}{\tilde{\xi}}$$

In (3.11a) we have separated the desired dynamics from the dynamics due to $\tilde{\xi} \neq 0$.

To find a clf for the augmented system it is natural to take the clf for the subsystem, W, and add a term penalizing the residual $\tilde{\xi}$. Let us select

$$V(x, \tilde{\xi}) = W(x) + \frac{1}{2}\tilde{\xi}^2$$

and find a globally stabilizing control law by making $\dot{V}$ negative definite.

$$\begin{aligned} \dot{V} &= W_x(x)\Big[f(x, \xi^{\text{des}}(x)) + \psi(x, \tilde{\xi})\tilde{\xi}\Big] + \tilde{\xi}\Big[u - \frac{\partial \xi^{\text{des}}(x)}{\partial x} f(x, \tilde{\xi} + \xi^{\text{des}}(x))\Big] \\ &= W_x(x) f(x, \xi^{\text{des}}(x)) + \tilde{\xi}\Big[W_x(x)\psi(x, \tilde{\xi}) + u - \frac{\partial \xi^{\text{des}}(x)}{\partial x} f(x, \tilde{\xi} + \xi^{\text{des}}(x))\Big] \end{aligned} \tag{3.12}$$

The first term is negative definite according to the assumptions. The second part, and thereby $\dot{V}$, can be made negative definite by choosing

$$u = -W_x(x)\psi(x, \tilde{\xi}) + \frac{\partial \xi^{\text{des}}(x)}{\partial x} f(x, \tilde{\xi} + \xi^{\text{des}}(x)) - \tilde{\xi}$$

This yields

$$\dot{V} = W_x(x) f(x, \xi^{\text{des}}(x)) - \tilde{\xi}^2$$

which proves the sought global asymptotic stability. □

The key idea in backstepping is to let certain states act as "virtual controls" of others. The same idea can be found in cascaded control design and singular perturbation theory [42].

The origin of backstepping is not quite clear due to its simultaneous and often implicit appearance in several papers in the late 1980's. However, it is fair to say that backstepping has been brought into the spotlight to a great extent thanks to the work of Professor Petar V. Kokotović and his coworkers.

The 1991 Bode lecture at the IEEE CDC, held by Kokotović [43], was devoted to the evolving subject and in 1992, Kanellakopoulos et al. [39] presented a mathematical "toolkit" for designing control laws for various nonlinear systems using backstepping. During the following years, the textbooks by Krstić et al. [46], Freeman and Kokotović [21], and Sepulchre et al. [65] were published. The progress of backstepping and other nonlinear control tools during the 1990's were surveyed by Kokotović [41] at the 1999 IFAC World Congress in Beijing.

Let us now deal with some issues related to practical control design using backstepping.

3.3.2 Which systems can be handled?

Input nonlinearities

An immediate extension of Proposition 3.1 is to allow for an input mapping to be present in (3.7b):

$$\dot{\xi} = g(x, \xi, u) \triangleq \tilde{u}$$

$\tilde{u}$ can now be selected according to (3.10) whereafter u can be found given that

$$g(x, \xi, u) = \tilde{u}$$

can be solved for u. If this is possible, we say that g is invertible w.r.t. u.

Feedback form systems

Also, we see that the system (3.7) can be further augmented. Assume that u is not the actual control input to the system, but merely another state variable obeying the dynamics

$$\dot{u} = v \tag{3.13}$$

Then (3.10) becomes a virtual control law, which along with the clf (3.9) can be used to find a globally stabilizing control law for (3.7) augmented by (3.13).

Now, either v is yet another state variable, in which case the backstepping procedure is repeated once again, or v is indeed the control input, in which case we are done.

Thus, by recursively applying backstepping, globally stabilizing control laws can be constructed for systems of the following lower triangular form, known as *pure-feedback form systems*:

$$\begin{aligned}
\dot{x} &= f(x, \xi_1) \\
\dot{\xi}_1 &= g_1(x, \xi_1, \xi_2) \\
&\vdots \\
\dot{\xi}_i &= g_i(x, \xi_1, \ldots, \xi_i, \xi_{i+1}) \\
&\vdots \\
\dot{\xi}_m &= g_m(x, \xi_1, \ldots, \xi_m, u)
\end{aligned} \tag{3.14}$$

For the design to succeed, a globally stabilizing virtual control law, $\xi_1 = \xi_1^{\text{des}}(x)$, along with a clf, must be known for the x subsystem. In addition, $g_i,\ i = 1, \ldots, m-1$ must be invertible w.r.t. ξ_{i+1} and g_m must be invertible w.r.t. u.

Systems for which the "new" variables enter in an affine way, are known as *strict-feedback form systems*:

$$\begin{aligned}
\dot{x} &= a(x) + b(x)\xi_1 \\
\dot{\xi}_1 &= a_1(x, \xi_1) + b_1(x, \xi_1)\xi_2 \\
&\vdots \\
\dot{\xi}_i &= a_i(x, \xi_1, \ldots, \xi_i) + b_i(x, \xi_1, \ldots, \xi_i)\xi_{i+1} \\
&\vdots \\
\dot{\xi}_m &= a_m(x, \xi_1, \ldots, \xi_m) + b_m(x, \xi_1, \ldots, \xi_m)u
\end{aligned}$$

Strict-feedback form systems are nice to deal with and often used for deriving results related to backstepping. Firstly, the invertability condition imposed above is satisfied given that $b_i \neq 0$. Secondly, if (3.7a) is affine in ξ, the control law (3.10) reduces to

$$u = \frac{\partial \xi^{\text{des}}(x)}{\partial x}(a(x) + b(x)\xi) - W_x(x)b(x) + \xi^{\text{des}}(x) - \xi$$

Dynamic backstepping

Even for certain systems which do not fit into a lower triangular feedback form, there exist backstepping designs. Fontaine and Kokotović [18] consider a two dimensional system where both states are affected by the control input:

$$\begin{aligned}
\dot{x}_1 &= \psi(x_1) + x_2 + \phi(u) \\
\dot{x}_2 &= u
\end{aligned}$$

Their approach is to first design a globally stabilizing virtual control law for the x_1-subsystem, considering $\eta = x_2 + \phi(u)$ as the input. Then backstepping is used to convert this virtual control law into a realizable one in terms of u. Their design results in a dynamic control law, and hence the term *dynamic backstepping* is used.

3.3.3 Which design choices are there?

The derivation of the backstepping control law (3.10) leaves a lot of room for variations. Let us now exploit some of these.

Dealing with nonlinearities

A trademark of backstepping is that is allows us to benefit from "useful" nonlinearities, naturally stabilizing the system. This can be done by choosing the virtual control laws properly. The following example demonstrates this fundamental difference to feedback linearization.

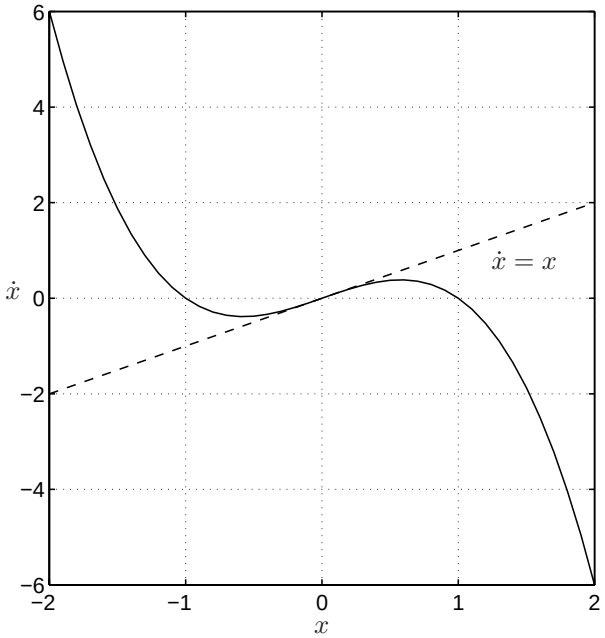


Figure 3.1 *The dynamics of the uncontrolled system $\dot{x} = -x^3 + x$. The linear term acts destabilizing around the origin.*

Example 3.1 (A useful nonlinearity I)
Consider the system

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

where $x = 0$ is the desired equilibrium. The uncontrolled dynamics, $\dot{x} = -x^3 + x$, are plotted in Figure 3.1. For the system to be stable, the sign of $\dot{x}$ should be opposite that of x for all x. This holds for large values of x where the cubic term $-x^3$ dominates the dynamics, but near the origin, the linear term x dominates and destabilizes the system.
Thus, to make the origin GAS, only the linear dynamics need to be counteracted by the control input. This can be achieved by selecting

$$u = -x \tag{3.15}$$

A clf is given by

$$W = \frac{1}{2}x^2 \tag{3.16}$$

which yields

$$\dot{W} = -x(-x^3 + x + u) = -x^4$$

proving the origin to be GAS according to Theorem 3.1.
Applying feedback linearization would have rendered the control law

$$u = x^3 - kx, \quad k > 1 \tag{3.17}$$

Obviously, this control law does not recognize the beneficial cubic nonlinearity but counteracts it, thus wasting control effort. Also, the feedback linearizing design is dangerous from a robustness perspective – what if the true system dynamics are $\dot{x} = -0.9x^3 + x + u$ and (3.17) is applied...

Weighting the clf

When constructing the combined clf (3.9), we can choose any weighted sum of the two terms,

$$V = cW + \frac{1}{2}(\xi - \xi^{\text{des}})^2, \ c > 0$$

In our designs, we will use the weight c to cancel certain terms in Equation (3.12). A technical hint is to put the weight on W since it yields nicer expressions.

Non-quadratic clf

Although quadratic clf:s are frequently used in backstepping, they do not always make up the best choice as the following example demonstrates.

Example 3.2 (A useful nonlinearity II)
Consider the system in Example 3.1 augmented by an integrator:

$$\begin{aligned}\dot{x}_1 &= -x_1^3 + x_1 + x_2\\ \dot{x}_2 &= u\end{aligned}$$

To benefit from the useful nonlinearity $-x_1^3$, let us reuse (3.15) as our virtual control law, i.e.,

$$x_2^{\text{des}}(x_1) = -x_1$$

As it turns out, the choice of clf for the x_1-subsystem will strongly affect the resulting control law. To see this we first reuse (3.16) and pick

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

Introducing

$$\tilde{x}_2 = x_2 - x_2^{\text{des}}(x_1) = x_2 + x_1$$

we can rewrite the system as

$$\begin{aligned}\dot{x}_1 &= -x_1^3 + \tilde{x}_2 \\ \dot{\tilde{x}}_2 &= u - x_1^3 + \tilde{x}_2\end{aligned}$$

Following the proof of Proposition 3.1, we add a quadratic term to W, to penalize the deviation from the suggested virtual control law:

$$V(x_1, x_2) = W(x_1) + \frac{1}{2}\big(x_2 - x_2^{\text{des}}(x_1)\big)^2 = \frac{1}{2}x_1^2 + \frac{1}{2}\tilde{x}_2^2$$

Differentiating w.r.t. time yields

$$\begin{aligned}\dot{V} &= x_1(-x_1^3 + \tilde{x}_2) + \tilde{x}_2(u - x_1^3 + \tilde{x}_2) \\ &= -x_1^4 + \tilde{x}_2(x_1 + u - x_1^3 + \tilde{x}_2)\end{aligned}$$

To render $\dot{V}$ negative definite, u must clearly dominate the $\tilde{x}_2$ term using a control input of, e.g., $-3\tilde{x}_2$. In addition, since the mixed terms between x_1 and $\tilde{x}_2$ are indefinite, there seems to be no other choice than to cancel them using the control law

$$u = x_1^3 - x_1 - 3\tilde{x}_2 = x_1^3 - 4x_1 - 3\tilde{x}_2$$

We note that this control law does not recognize the fact that x_1-subsystem is naturally stabilized for high values of x_1 but instead counteracts this effect, thereby wasting control effort.
So how should we pick $W(x_1)$ to avoid this cancellation? One idea is not to specify the clf beforehand, but instead let the choice of W be decided by the backstepping design. Thus, we let W be any function satisfying Definition 3.3. As before, use

$$V(x_1, x_2) = W(x_1) + \frac{1}{2}\tilde{x}_2^2$$

and compute its time derivative.

$$\begin{aligned}\dot{V} &= W'(x_1)(-x_1^3 + \tilde{x}_2) + \tilde{x}_2(u - x_1^3 + \tilde{x}_2) \\ &= -W'(x_1)x_1^3 + \tilde{x}_2(W'(x_1) + u - x_1^3 + \tilde{x}_2)\end{aligned}$$

We now use our extended design freedom and select a W so that the indefinite mixed terms cancel each other. This is satisfied by

$$W'(x_1) = x_1^3, \quad W(x_1) = \frac{1}{4}x_1^4$$

which indeed is a valid choice. We now have

$$\dot{V} = -x_1^6 + \tilde{x}_2(u + \tilde{x}_2)$$

Clearly, the control u no longer has to cancel the cubic x_1 term but can be chosen linear in x_1 and x_2.

$$u = -3\tilde{x}_2 = -3x_1 - 3x_2$$

renders $\dot{V} = -x_1^6 - 2\tilde{x}_2^2$ negative definite and thus makes the origin GAS.

This refinement of backstepping is due to Krstić et al. [45]. The technique of designing a non-quadratic clf will be used for flight control design in Chapter 5, where some of the aerodynamic forces also have the property of being nonlinear but still stabilizing.

Implicit residuals

In (3.9) the deviation from the desired virtual control law is penalized through the difference $\xi - \xi^{\text{des}}(x)$. Another way of forcing ξ towards ξ^{des} is to instead penalize the *implicit residual*

$$\alpha(\xi) - \alpha(\xi^{\text{des}}(x)) \triangleq \alpha(\xi) - \alpha^{\text{des}}(x)$$

where $\alpha(\xi)$ is an invertible and strictly monotone mapping. This leads to the clf

$$V(x, \xi) = W(x) + \frac{1}{2}\big(\alpha(\xi) - \alpha^{\text{des}}(x)\big)^2$$

An equivalent way of reaching this clf is to replace ξ by $\alpha(\xi)$ in the system description (3.7):

$$\begin{aligned} \dot{x} &= g(x, \alpha) \\ \dot{\alpha} &= \alpha'(\xi)u \end{aligned}$$

Here, $g(x, \alpha(\xi)) = f(x, \xi)$ has been introduced. Applying (3.10) now gives us

$$u = \frac{1}{\alpha'(\xi)}\Big[\frac{\partial \alpha^{\text{des}}(x)}{\partial x} g(x, \alpha) - W_x(x)\frac{g(x, \alpha) - g(x, \alpha^{\text{des}}(x))}{\alpha - \alpha^{\text{des}}(x)} + \alpha^{\text{des}}(x) - \alpha\Big] \tag{3.18}$$

Example 3.3
Consider the system

$$\begin{aligned} \dot{x}_1 &= x_1^3 + x_2^5 + x_2 \\ \dot{x}_2 &= u \end{aligned}$$

A clf for the x_1-subsystem is given by $W(x_1) = \frac{1}{2}x_1^2$ which gives

$$\dot{W} = x_1(x_1^3 + x_2^5 + x_2)$$

In terms of $\alpha = x_2^5 + x_2$, a globally stabilizing virtual control law is easy to find. Pick, e.g., $\alpha^{\text{des}}(x_1) = -2x_1^3$ which yields $\dot{W} = -x_1^4$ negative definite. Inserting this into (3.18) gives the control law

$$u = \frac{1}{5x_2^4 + 1}\Big[-6x_1^2(x_1^3 + x_2^5 + x_2) - x_1 - 2x_1^3 - x_2^5 - x_2\Big]$$

In [2], this technique was used for speed control of a switched reluctance motor where it was convenient to formulate the virtual control law in terms of the square current i^2.

Optimal backstepping

In linear control, one often seeks control laws that are optimal in some sense, due to their ability to suppress external disturbances and to function despite model errors, as in the case of $\mathcal{H}_\infty$ and linear quadratic control [79].

It is therefore natural that efforts have been made to extend these designs to nonlinear control. The difficulty lies in the Hamilton-Jacobi-Bellman equation that needs to be solved in order to find the control law.

A way to surpass this problem is to only require the desired optimality to hold *locally* around the origin, where the system can be approximated by its linearization. In the global perspective, one settles for optimality according to some cost functional that the designer cannot rule over precisely. This is known as inverse optimality, which is the topic of Chapter 4.

Contributions along this line can be found for strict-feedback form systems. Ezal et al. [17] use backstepping to construct controllers which are locally $\mathcal{H}_\infty$-optimal. Löfberg [51] designs backstepping controllers which are locally optimal according to a linear quadratic performance index.

One advantage of using an optimality based approach is that the designer then specifies an optimality criterion rather than virtual control laws and the clf:s themselves. This enhances the resemblance with linear control.

3.4 Related Lyapunov Designs

Besides state feedback backstepping, several other constructive nonlinear control designs have been developed during the last decade. We will now outline some of these.

3.4.1 Forwarding

The backstepping philosophy applies to systems of the form (3.7). Another class of nonlinear systems for which one can also construct globally stabilizing control laws are those that can be written

$$\dot{x} = f(x, u) \tag{3.19a}$$

$$\dot{\xi} = g(x, u) \tag{3.19b}$$

A clf and a globally stabilizing control law for the x-subsystem (3.19a) are assumed to be known. The question is how to augment this control law to also stabilize the integrator state ξ in (3.19b). This problem, which can be seen as a dual to the one in backstepping, can be solved using so called *forwarding* [67].

By combining feedback (3.7) and feedforward (3.19) systems, *interlaced systems* can be constructed. Using backstepping in combination with forwarding, such systems can also be systematically stabilized [66].

3.4.2 Adaptive, robust, and observer backstepping

So far we have only considered the case where all the state variables are available for feedback and where the model is completely known. For the non-ideal cases where this is not true, there are other flavors of backstepping to resort to.

For systems with parametric uncertainties, there exists *adaptive backstepping* [46]. Here, a parameter estimate update law is designed such that closed loop stability is guaranteed when the parameter estimate is used by the controller. In Section 6.3 we will see how this technique can be used to estimate and cancel unknown additive disturbances on the control input.

Robust backstepping [21] designs exist for systems with imperfect model information. Here, the idea is to select a control law such that a Lyapunov function decreases for all systems comprised by the given model uncertainty.

In cases where not all the state variables can be measured, the need for observers arises. The separation principle valid for linear systems does not hold for nonlinear systems in general. Therefore, care must be taken when designing the feedback law based on the state estimates. This is the topic of *observer backstepping* [39, 46].

3.5 Applications of Backstepping

Although backstepping theory has a rather short history, numerous practical applications can be found in the literature. This fact indicates that the need for a nonlinear design methodology handling a number of practical problems, as discussed in the previous section, has existed for a long time. We now survey some publications regarding applied backstepping. This survey is by no means complete, but is intended to show the broad spectrum of engineering disciplines in which backstepping has been used.

Backstepping designs can be found for a wide variety of electrical motors [2, 10, 11, 33, 34]. Turbocharged diesel engines are considered in [20, 37] while jet engines are the subject of [45].

In [25, 75], backstepping is used for automatic ship positioning. In [75], the controller is made locally $\mathcal{H}_\infty$-optimal based on results in [17].

Robotics is another field where backstepping designs can be found. Tracking control in considered in [38] and [9] where the latter is a survey of approaches valid for various assumptions regarding the knowledge of the model.

There also exist a few papers, combining flight control and backstepping. [68] treats formation flight control of unmanned aerial vehicles. [69] and [73] use backstepping to design flight control laws which are adaptive to changes in the aerodynamic forces and moments due to, e.g., actuator failures. Also, the Lyapunov functions used contain a term penalizing the integral of the tracking error, enhancing the robustness.

4

Inverse Optimal Control

This chapter is preparatory for the upcoming control designs. The tools we develop in this chapter will be used to show that the control laws derived in Chapter 5 each minimize a certain cost functional. This route of first deriving a control law and then determining which cost it minimizes, and thus in which sense it is optimal, is known as *inverse optimal control*.

The material in this chapter will be presented in a rather intuitive manner. A mathematically strict treatment of the subject can be found in, e.g., Sepulchre et al. [65]. In Section 4.1 the general infinite horizon optimal control problem is introduced. In Section 4.2, systems which are affine in the control input are considered, and some standard inverse results are derived for cost functionals which are quadratic in the input. Finally, the well known gain margin result of optimal control is shown in Section 4.3.

4.1 Optimal Control

A general idea within control design is to select a control law which is optimal in some sense. Given a dynamic system

$$\dot{x} = f(x, u)$$

where $x \in \mathbb{R}^n$ is the state vector and $u \in \mathbb{R}^m$ is the control input, we seek the control law $u(x)$ that minimizes the cost functional

$$J = \int_0^\infty L(x,u)dt$$

By choosing L properly, the system is guaranteed to reach steady state as $t \to \infty$. J, representing the cost to get there, the *cost-to-go*, will depend on the initial state of the system, $x(0)$. We therefore write this cost $J(x)$.

The optimal control law is denoted by $u^*(x)$. When this optimal control law is applied, $J(x)$ will decrease along the trajectory, since the cost-to-go must continuously decrease by the principle of optimality [6]. This means that $V(x) = J(x)$ is a Lyapunov function for the controlled system. At steady state it must hold that $V = 0$. Hence, the following holds:

$$V(x(0)) = \int_0^\infty L(x,u^*)dt = -\big[\underbrace{V(x(\infty))}_{=0} - V(x(0))\big] = -\int_0^\infty \dot{V}(x)dt$$

Clearly, when the optimal control law is used, L and $-\dot{V}$ coincide. This motivates the Hamilton-Jacobi-Bellman (HJB) equation

$$0 = \min_u \big[L(x,u) + V_x(x)f(x,u)\big] \tag{4.1}$$

for finding the optimal control law u^* along with a Lyapunov function $V(x)$ for the controlled system.

4.2 Inverse Optimal Control

It is well known that in general, it is not feasible to solve the HJB equation (4.1). We therefore restrict our discussion to dynamic systems of the form

$$\dot{x} = f(x) + g(x)u \tag{4.2}$$

For these systems, the HJB equation is greatly simplified if L is chosen quadratic in u according to

$$L(x,u) = q(x) + u^T R(x) u$$

where $q(x) > 0$, $x \neq 0$ and $R(x)$ is symmetric matrix, positive definite for all x. Inserting this into (4.1) yields

$$0 = \min_u \big[q(x) + u^T R(x)u + V_x(x)(f(x) + g(x)u)\big] \tag{4.3}$$

The equation is solved in two steps. First we find the minimizing u, and then we solve for equality to zero. The minimization can be done by completion of squares:

$$\begin{aligned} &q + u^T Ru + V_x f + V_x g u = \\ &\qquad q + V_x f + \big[u + \frac{1}{2}R^{-1}(V_x g)^T\big]^T R\big[u + \frac{1}{2}R^{-1}(V_x g)^T\big] - \frac{1}{4}V_x g R^{-1}(V_x g)^T \end{aligned}$$

The control input u only appears in the "square", positive definite term. The minimum therefore occurs when this term is set to zero, which is achieved by

$$u^* = -k(x) = -\frac{1}{2}R^{-1}(V_x g)^T \tag{4.4}$$

What remains is to insert this control law into (4.3). This gives us

$$0 = q + V_x f - \frac{1}{4}V_x g R^{-1}(V_x g)^T \tag{4.5}$$

Equations (4.4) and (4.5) provide the connection between the cost functional, given by $q(x)$ and $R(x)$, and the optimal control strategy, in terms of $k(x)$ and $V(x)$. As for the general problem in the previous section, it is in general not feasible to solve for $k(x)$ and $V(x)$ given $q(x)$ and $R(x)$ of the designer's choice. However, we see that the reverse task is simpler. Given a control law $k(x)$ and a clf $V(x)$ (corresponding to a Lyapunov function for the controlled system), $q(x)$ and $R(x)$, determining the cost functional that is minimized, can be found by solving

$$k(x) = \frac{1}{2}R^{-1}(x)(V_x(x)g(x))^T \tag{4.6}$$

$$q(x) = -V_x(x)f(x) + \frac{1}{2}V_x(x)g(x)k(x) \tag{4.7}$$

For a single input system we can explicitly solve for $R(x)$:

$$R(x) = \frac{V_x(x)g(x)}{2k(x)} \tag{4.8}$$

Example 4.1 (Exploiting useful nonlinearities is optimal)
Let us return to the system considered in Example 3.2 and show that the derived backstepping control law is indeed optimal w.r.t. a meaningful cost functional. The original dynamics,

$$\begin{aligned}\dot{x}_1 &= -x_1^3 + x_1 + x_2\\ \dot{x}_2 &= u\end{aligned}$$

can be written in the form (4.2) with

$$f(x) = \begin{pmatrix} -x_1^3 + x_1 + x_2 \\ 0 \end{pmatrix}, \quad g(x) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

In Example 3.2,

$$u = -k(x) = -3(x_1 + x_2)$$

was shown to be globally stabilizing using the clf

$$V(x) = \frac{1}{4}x_1^4 + \frac{1}{2}(x_1 + x_2)^2$$

which satisfies

$$V_x(x) = \begin{pmatrix} x_1^3 + x_1 + x_2 & x_1 + x_2 \end{pmatrix}$$

Inserting this into (4.7) and (4.8) yields

$$\begin{aligned} R(x) &= \frac{x_1 + x_2}{2 \cdot 3(x_1 + x_2)} = \frac{1}{6} \\ q(x) &= -(x_1^3 + x_1 + x_2)(-x_1^3 + x_1 + x_2) + \frac{1}{2}(x_1 + x_2) \cdot 3(x_1 + x_2) \\ &= x_1^6 + \frac{1}{2}(x_1 + x_2)^2 \end{aligned}$$

Thus, the suggested control law minimizes the cost functional

$$J = \int_0^\infty (x_1^6 + \frac{1}{2}(x_1 + x_2)^2 + \frac{1}{6}u^2)dt$$

4.3 Robustness of Optimal Control

Besides optimal control being an intuitively appealing approach, the resulting control laws inherently possess certain robustness properties [24]. One important property regards the gain margin.

Assume that the prescribed optimal control input (4.4) cannot be produced exactly, but that the actual control input is

$$u = \Gamma(x)u^* \tag{4.9}$$

where $\Gamma(x) > 0$ is a scalar, see Figure 4.1. Actuator saturation, for example, can be modeled as gain reduction, $\Gamma(x) < 1$. Are optimal controllers robust to such changes in the gain? The control law (4.9) is globally stabilizing provided that

$$\dot{V} = V_x f + V_x g u = V_x f + \Gamma V_x g u^*$$

is negative definite. From the assumptions and (4.7) we know that

$$-q = V_x f + \frac{1}{2} V_x g u^*$$

is negative definite. Combining these two equations yields

$$\dot{V} = -q + (\Gamma - \frac{1}{2})V_x g u^* = -q(x) - (\Gamma(x) - \frac{1}{2}) \cdot \underbrace{\frac{1}{2} V_x(x) g(x) R^{-1}(x) (V_x(x) g(x))^T}_{\text{positive (semi-)definite}}$$

Apparently, $\dot{V}$ is negative definite (at least) for all $\Gamma(x) \geq \frac{1}{2}$. Thus, all state feedback control laws which solve an optimal control problem of the type considered in Section 4.2, have a gain margin of $[\frac{1}{2}, \infty]$.

Note that the actual tolerable gain reduction may be more than 50%. In Example 3.2, any control law $u = -k\tilde{x}_2$ where $k > 1$ makes $\dot{V}$ negative definite and hence is globally stabilizing. The selected control law $u = -3\tilde{x}_2$ thus has a gain margin of $]\frac{1}{3}, \infty]$.

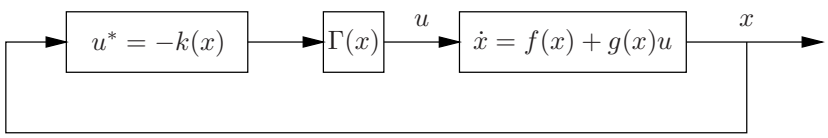


Figure 4.1 *The optimal control law $u^* = -k(x)$ remains globally stabilizing for any gain perturbation $\Gamma(x) \geq \frac{1}{2}$.*

5

Backstepping Designs for Flight Control

In the previous chapters we have introduced the aircraft dynamics, the backstepping design procedure and inverse optimality tools for evaluating state feedback laws. We now have the toolbox we need to live up to the title of the thesis, and do *flight control design using backstepping*. This chapter is the core of the thesis.

We will design flight control laws for two different objectives – for general maneuvering (control of α, β, and p_s) and for flight path angle control (control of γ). The two presentations in Sections 5.1 and 5.2 follow the same outline. First, the relevant dynamics from Chapter 2 are reproduced and the assumptions needed for making the control design feasible are stated. The flight control problem of interest is then viewed as a more general nonlinear control problem and backstepping is used to derive globally stabilizing control laws, whose properties are investigated. We finally return to the flight control context and investigate which practical consequences applying the derived control laws leads to.

5.1 General Maneuvering

We begin by designing flight controllers for general purpose maneuvering, useful, e.g., in dogfight situations where maneuverability is very important in order to get a first shot opportunity. Sharp turns imply high angles of attack and a prime feature of our design is that stability is guaranteed for all angles of attack, including

post-stall, given that enough pitching moment can be produced.

5.1.1 Objectives, dynamics, and assumptions

Following the discussion in Section 2.2 on relevant variables to be controlled for this type of general maneuvering we set up the following objectives:

- Fore-and-aft control stick deflections should demand angle of attack.
- Side-to-side control stick deflections should demand stability axis roll rate.
- The sideslip should be kept zero at all times.

The controlled variables are illustrated in Figure 2.3. With these objectives, the task of the control system is to make

$$\begin{aligned} \alpha &= \alpha^{\text{ref}} \\ \beta &= 0 \\ p_s &= p_s^{\text{ref}} \end{aligned}$$

a globally asymptotically stable equilibrium. Speed control is assumed to be handled separately and will not be dealt with.

The relevant dynamics consist of the angular rate dynamics in (2.6) and the α and β dynamics in (2.7b) and (2.7c). Trading the scalar moment equations (2.6) for the compact vector description (2.2) yields

$$\dot{\alpha} = q - (p\cos\alpha + r\sin\alpha)\tan\beta + \frac{1}{mV_T\cos\beta}(-L - F_T\sin\alpha + mg_2) \tag{5.1a}$$

$$\dot{\beta} = p\sin\alpha - r\cos\alpha + \frac{1}{mV_T}(Y - F_T\cos\alpha\sin\beta + mg_3) \tag{5.1b}$$

$$\mathbf{M} = I\dot{\omega} + \omega \times I\omega \tag{5.1c}$$

where

$$\begin{aligned} g_2 &= g(\cos\alpha\cos\theta\cos\phi + \sin\alpha\sin\theta) \\ g_3 &= g(\cos\beta\cos\theta\sin\phi + \sin\beta\cos\alpha\sin\theta - \sin\alpha\sin\beta\cos\theta\cos\phi) \end{aligned}$$

For backstepping to be applicable, the system dynamics must comply with the pure-feedback form (3.14). We therefore need the following assumption:

A1. Control surface deflections only produce aerodynamic moments, and not forces. Also neglecting the dependence on the angular rates, the aerodynamic force coefficients can be written

$$\begin{aligned} &\text{lift force coefficient: } C_L(\alpha) \\ &\text{side force coefficient: } C_Y(\beta) \end{aligned}$$

whose characteristics are shown in Figure 5.1.

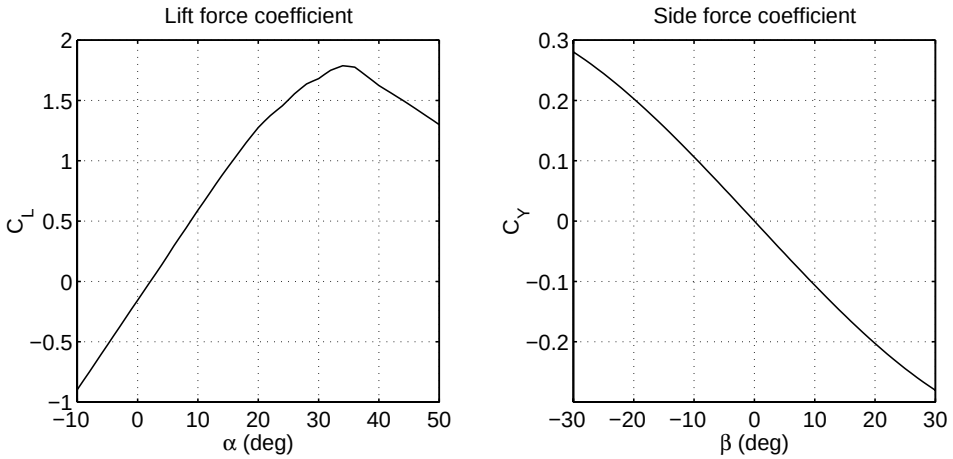


Figure 5.1 *Typical lift force coefficient vs. angle of attack and side force coefficient vs. sideslip characteristics.*

To simplify the design we also make the following assumptions:

A2. The speed, altitude, and orientation of the aircraft vary slowly compared to the controlled variables. Therefore, their time derivatives can be neglected.

A3. Longitudinal and lateral commands are assumed not to be applied simultaneously. Hence, when α is controlled, β and p_s are considered constant. Conversely, when β and p_s are controlled, α is considered constant.

A4. The control surface actuator dynamics are assumed to be fast enough to be disregarded.

Assumption A3 certainly does not hold in practice but allows us to decouple the control design. As we will show using simulations in Chapter 7, the resulting control laws work well in practice.

Since the roll rate to be controlled, p_s, is expressed in the stability-axes coordinate system, we need to establish the relationship to the body-axes angular velocity, ω. The stability-axes angular velocity,

$$\omega_s = \begin{pmatrix} p_s & q_s & r_s \end{pmatrix}^T$$

is related to the body-axes angular velocity

$$\omega = \begin{pmatrix} p & q & r \end{pmatrix}^T$$

through the transformation

$$\omega_s = R_{sb}\omega, \quad R_{sb} = \begin{pmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix} \tag{5.2}$$

Note that the transformation matrix R_{sb} satisfies $R_{sb}^{-1} = R_{sb}^T$.

Inspecting the dynamics (5.1), we see that it is more convenient to work with ω_s rather than ω. Introducing

$$u = \begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}^T = \dot{\omega}_s$$

we can rewrite the aircraft dynamics (5.1) as

$$\dot{p}_s = u_1 \tag{5.3a}$$

$$\dot{\alpha} = q_s - p_s \tan\beta + \frac{1}{mV_T\cos\beta}(-L(\alpha) - F_T\sin\alpha + mg_2) \tag{5.3b}$$

$$\dot{q}_s = u_2 \tag{5.3c}$$

$$\dot{\beta} = -r_s + \frac{1}{mV_T}(Y(\beta) - F_T\cos\alpha\sin\beta + mg_3) \tag{5.3d}$$

$$\dot{r}_s = u_3 \tag{5.3e}$$

These are the dynamics we will use for the control design.

The relationship between u, which will be considered as the control input during the control design, and the actual control input, δ, can be found by combining (5.1c) and (5.2). Under assumption A3 above, α, and thereby also R_{sb}, are considered constant while realizing u_1 and u_3 which relate to lateral control. This yields

$$u = \dot{\omega}_s = R_{sb}\dot{\omega} = R_{sb}I^{-1}(\mathbf{M} - \omega \times I\omega) \tag{5.4}$$

Solving for the net moment, $\mathbf{M}$, which depends on the control surface deflections, δ, we get

$$\mathbf{M}(\delta) = IR_{sb}^T u + \omega \times I\omega \tag{5.5}$$

We will postpone the discussion on how to practically solve for δ given u until Chapter 7.

5.1.2 Backstepping control design

The nonlinear control problem

To begin with, note the structural similarities between the angle of attack dynamics (5.3b)–(5.3c) and the sideslip dynamics (5.3d)–(5.3e). Both these second order systems can be written

$$\begin{aligned} \dot{w}_1 &= f(w_1, y) + w_2 \\ \dot{w}_2 &= u \end{aligned} \tag{5.6}$$

General system (5.6)	α dynamics (5.3b)–(5.3c)	β dynamics (5.3d)–(5.3e)
$\mathcal{R}$	α^{ref}	0
w_1	α	β
w_2	q_s	$-r_s$
u	u_2	$-u_3$
y	$p_s, \beta, V_T, h, \theta, \phi$	$\alpha, V_T, h, \theta, \phi$
$f(w_1, y)$	$f_\alpha(\alpha, y_\alpha)$	$f_\beta(\beta, y_\beta)$

Table 5.1 *The relationships between the general nonlinear system* (5.6) *and the angle of attack and sideslip dynamics in* (5.3).

where y represents the influence of variables that we regard as constant, and $w_1 = \mathcal{R}$ is the desired equilibrium. We assume that the reference, $\mathcal{R}$, is constant, which is the best we can do without making any assumptions regarding the maneuvers that the pilot wants to perform. Table 5.1 summarizes the relationships between (5.6) and the original aircraft dynamics (5.3). Here,

$$f_\alpha(\alpha, y_\alpha) = -p_s \tan\beta + \frac{1}{mV_T \cos\beta}(-L(\alpha) - F_T \sin\alpha + mg_2) \tag{5.7}$$

$$f_\beta(\beta, y_\beta) = \frac{1}{mV_T}(Y(\beta) - F_T \cos\alpha \sin\beta + mg_3) \tag{5.8}$$

have been introduced.

For notational convenience it is favorable to make the origin the desired equilibrium. We therefore define

$$\begin{aligned} x_1 &= w_1 - \mathcal{R} \\ x_2 &= w_2 + f(\mathcal{R}, y) \\ \Phi(x_1) &= f(x_1 + \mathcal{R}, y) - f(\mathcal{R}, y) \end{aligned} \tag{5.9}$$

which yields $\Phi(0) = 0$. In these coordinates, the dynamics (5.6) become

$$\dot{x}_1 = \Phi(x_1) + x_2 \tag{5.10a}$$

$$\dot{x}_2 = u \tag{5.10b}$$

The relationships between (5.10) and the original aircraft dynamics (5.3) can be found in Table 5.2. If the aircraft is assumed to be built symmetrically such that the side force, Y, is zero for zero sideslip, we get

$$f_\beta(0, y_\beta) = \frac{1}{V_T} g \cos\theta \sin\phi \tag{5.11}$$

using (2.8).

General system (5.10)	α dynamics (5.3b)–(5.3c)	β dynamics (5.3d)–(5.3e)
x_1	$\alpha - \alpha^{\text{ref}}$	β
x_2	$q_s + f_\alpha(\alpha^{\text{ref}}, y_\alpha)$	$-r_s + f_\beta(0, y_\beta)$
u	u_2	$-u_3$
$\Phi(x_1)$	$f_\alpha(\alpha, y_\alpha) - f_\alpha(\alpha^{\text{ref}}, y_\alpha)$	$f_\beta(\beta, y_\beta) - f_\beta(0, y_\beta)$

Table 5.2 *The relationships between the translated general nonlinear system* (5.10) *and the angle of attack and sideslip dynamics in* (5.3).

In the remainder of this section, we will use backstepping to construct globally stabilizing state feedback control laws for the system (5.10) assuming a general nonlinearity Φ. In Section 5.1.3 we will return to the flight control context and express the control laws in the original coordinates.

The backstepping design

In the case of α and β control, Φ will contain the negative lift force, $-L$, and the side force, Y. As seen from Figure 5.1, the gradients of these contributions are negative (note the sign on the lift force) in large parts of the operating range. The only exception is the post-stall behavior of the lift force. The key property of backstepping is that it allows us to benefit from these inherently stabilizing forces and not cancel them. If we do not have to cancel them, we need less information about them, which is appealing from a robustness point of view.

Step 1: In the spirit of backstepping, we start by finding a globally stabilizing virtual control law for the for the x_1-subsystem (5.10a). Not to constrain ourselves at this stage, we pick a general control law

$$x_2^{\text{des}} = -\Psi(x_1) \tag{5.12}$$

The idea is to let the necessary demands on Ψ be revealed by the design below. Temporarily pick the clf

$$W(x_1) = \frac{1}{2}x_1^2$$

To make $x_1 = 0$ GAS we must make $\dot{W}$ negative definite. Thus, we select a Ψ satisfying

$$\dot{W}|_{x_2 = x_2^{\text{des}}} = \big(\Phi(x_1) - \Psi(x_1)\big)x_1 < 0, \quad x_1 \neq 0 \tag{5.13}$$

This condition is natural, since for $x_1 = 0$ to be GAS, the desired net x_1 dynamics, $\dot{x}_1 = \Phi(x_1) - \Psi(x_1)$ must lie in the second and fourth quadrants.

Step 2: Continue by introducing the residual

$$\tilde{x}_2 = x_2 - x_2^{\text{des}} = x_2 + \Psi(x_1)$$

and rewrite the dynamics (5.10) in terms of x_1 and $\tilde{x}_2$.

$$\begin{aligned} \dot{x}_1 &= \Phi(x_1) - \Psi(x_1) + \tilde{x}_2 \\ \dot{\tilde{x}}_2 &= u + \Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2\big) \end{aligned} \tag{5.14}$$

Remembering the benefits of using a non-quadratic clf (cf. Example 3.2), we select

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

as the clf for the total system (5.14), where F is any valid clf for the x_1-subsystem. Specifically this means that

$$\dot{F}(x_1)|_{x_2 = x_2^{\text{des}}} = F'(x_1)\big(\Phi(x_1) - \Psi(x_1)\big) = -U(x_1) \tag{5.15}$$

where $U(x_1)$ is positive definite. Differentiating V w.r.t. time we get

$$\begin{aligned} \dot{V} &= F'(x_1)\big[\Phi(x_1) - \Psi(x_1) + \tilde{x}_2\big] + \tilde{x}_2\big[u + \Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2\big)\big] \\ &= -U(x_1) + \tilde{x}_2\big[F'(x_1) + u + \Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1)\big) + \Psi'(x_1)\tilde{x}_2\big] \end{aligned}$$

We can reduce the complexity of the second term by selecting F such that the x_1 terms inside the brackets cancel each other. This is achieved by

$$F'(x_1) = -\Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1)\big), \quad F(0) = 0$$

Inserting this into (5.15) yields

$$U(x_1) = \Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1)\big)^2$$

For $U(x_1)$ to be positive definite, Ψ must satisfy

$$\Psi'(x_1) > 0, \quad x_1 \neq 0 \tag{5.16}$$

Note that $\Phi(x_1) \neq \Psi(x_1)$ holds according to (5.13). An intuitive interpretation of this condition is that the virtual control law (5.12) must provide negative feedback everywhere, even locally. We now have that

$$\dot{V} = -U(x_1) + \tilde{x}_2\big[u + \Psi'(x_1)\tilde{x}_2\big]$$

To make $\dot{V}$ negative definite, u must dominate the positive definite, destabilizing term $\Psi'(x_1)\tilde{x}_2^2$. If $\Psi'(x)$ has an upper bound, u can be selected linear in $\tilde{x}_2$. Choosing

$$u = -k\tilde{x}_2 = -k\big(x_2 + \Psi(x_1)\big) \tag{5.17}$$

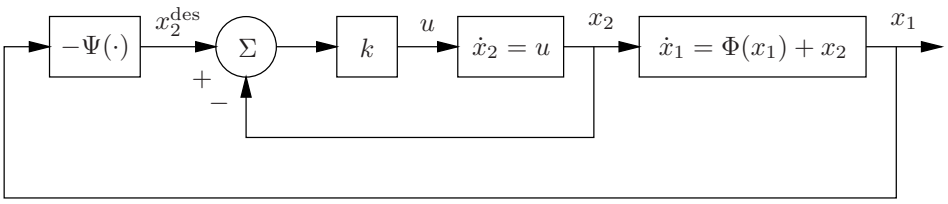


Figure 5.2 *The nonlinear system 5.10 can be globally stabilized through a cascaded control structure.*

where

$$k > \max_{x_1} \Psi'(x_1) \tag{5.18}$$

renders

$$\dot{V} = -U(x_1) - (k - \Psi'(x_1))\tilde{x}_2^2$$

negative definite, thus making the origin of (5.10) GAS. Note the cascaded structure of the control law as illustrated in Figure 5.2. This way of viewing the control law motivates the condition (5.18). The condition apparently states that the inner control loop must have a higher feedback gain than the outer loop.

Before we conclude, let us investigate which system nonlinearities Φ can be handled using the control law (5.17). If $\Psi'(x_1)$ is upper bounded by k as in (5.18), then the growth rate of Ψ must also be bounded in the sense that

$$\frac{\Psi(x_1)}{x_1} \leq \max_{x_1} \Psi'(x_1) < k$$

and hence, Ψ must be confined to the sectors shown in Figure 5.3(b). Dividing (5.13) by x_1^2 and inserting this inequality yields

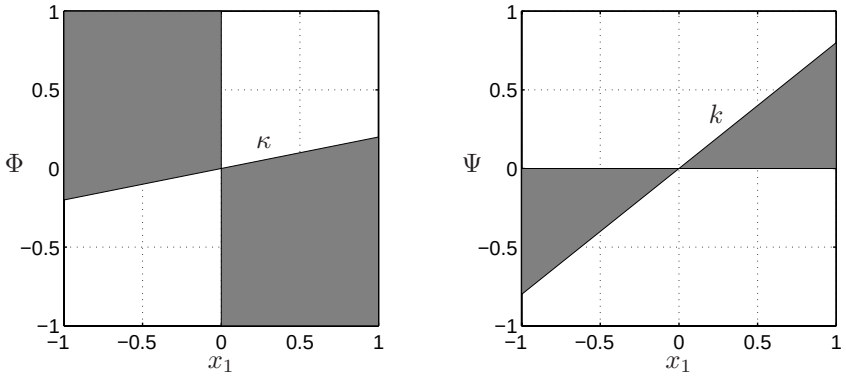
$$\frac{\Phi(x_1)}{x_1} < \frac{\Psi(x_1)}{x_1} < k$$

Thus, for the control law (5.17) to be applicable, the growth of the system nonlinearity Φ must also be linearly upper bounded as depicted in Figure 5.3(a). Conversely, given an upper bound on $\Phi(x_1)/x_1$ we can always find a Ψ and a k such that (5.17) is globally stabilizing.

Let us summarize our findings as a proposition.

Proposition 5.1
Consider the system

$$\dot{x}_1 = \Phi(x_1) + x_2 \tag{5.19a}$$
$$\dot{x}_2 = u \tag{5.19b}$$



(a) System nonlinearity (b) Virtual control law

Figure 5.3 *Φ and Ψ must be sector bounded for a globally stabilizing control law of the form* (5.17) *to exist.*

where $\Phi(0) = 0$. Assume there exists a constant κ such that

$$\frac{\Phi(x_1)}{x_1} \leq \kappa \text{ for all } x_1 \neq 0 \tag{5.20}$$

Then, a globally stabilizing control law

$$u = -k\big(x_2 + \Psi(x_1)\big) \tag{5.21}$$

can be found, where

$$\big(\Phi(x_1) - \Psi(x_1)\big)x_1 < 0, \quad x_1 \neq 0 \tag{5.22}$$

and

$$0 < \Psi'(x_1) < k \tag{5.23}$$

Moreover, a clf is given by

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

which satisfies

$$\dot{V} = -\Psi'(x_1)\big(\Phi(x_1) - \Psi(x_1)\big)^2 - (k - \Psi'(x_1))\tilde{x}_2^2$$

□

Inverse optimality

Proposition 5.1 gives us a family of control laws which all globally stabilize the system of interest. Before discussing which choices of Ψ might be of interest, let us further examine the properties of the derived control law. Using the tools of Chapter 4 we will show that (5.21) is actually optimal with respect to a meaningful cost functional, provided that k is chosen properly.

The control input enters the system (5.10) affinely and thus the tools of Chapter 4 can be used. However, we will use the transformed system (5.14) to compute the cost functional that is minimized since the expressions become simpler. Comparing (5.14) with (4.2), we have that

$$f(x) = \begin{pmatrix} \Phi(x_1) - \Psi(x_1) + \tilde{x}_2 \\ \Psi'(x_1)(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2) \end{pmatrix}, \quad g(x) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

We also have that

$$V_x = \begin{pmatrix} -\Psi'(x_1)(\Phi(x_1) - \Psi(x_1)) & \tilde{x}_2 \end{pmatrix}$$

Inserting this into (4.7) and (4.8) yields

$$\begin{aligned} R(x) &= \frac{\tilde{x}_2}{2k\tilde{x}_2} = \frac{1}{2k} \\ q(x) &= \Psi'(x_1)(\Phi(x_1) - \Psi(x_1))(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2) \\ &\quad + \tilde{x}_2\Psi'(x_1)(\Phi(x_1) - \Psi(x_1) + \tilde{x}_2) - \frac{1}{2}\tilde{x}_2 \cdot k\tilde{x}_2 \\ &= \Psi'(x_1)(\Phi(x_1) - \Psi(x_1))^2 + (\frac{k}{2} - \Psi'(x_1))\tilde{x}_2^2 \end{aligned}$$

Apparently, to make $q(x)$ positive definite, which is required for the cost functional to be "meaningful", k should be chosen such that

$$k > 2 \cdot \max_{x_1} \Psi'(x_1)$$

Note that this lower limit for inverse optimality is twice the limit in (5.23) regarding global stability. This is natural considering the 50% gain reduction robustness of all optimal controllers, cf. Chapter 4.

Proposition 5.2
The control law (5.21) *is optimal w.r.t. a meaningful cost functional for*

$$k > 2 \cdot \max_{x_1} \Psi'(x_1)$$

The control law then minimizes

$$\int_0^\infty \Big(\Psi'(x_1)(\Phi(x_1) - \Psi(x_1))^2 + (\frac{k}{2} - \Psi'(x_1))(x_2 + \Psi(x_1))^2 + \frac{1}{2k}u^2 \Big) dt$$

□

Two special cases

So how should Ψ be chosen? Let us investigate two special choices. One has control law simplicity in focus while the other aims at linearizing the dynamics.

Corollary 5.1 (Linear control)
Consider the system (5.19). *A globally stabilizing control law, linear in the states, is given by*

$$u = -k_2(x_2 + k_1 x_1)$$

where

$$k_2 > k_1 > \max\{0, \kappa\}$$

with κ as defined in (5.20). *In addition, for $k_2 > 2k_1$ the control law minimizes a meaningful cost functional given by*

$$\int_0^\infty \Big(k_1(\Phi(x_1) - k_1 x_1)^2 + (\frac{k_2}{2} - k_1)(x_2 + k_1 x_1)^2 + \frac{1}{2k_2} u^2 \Big) dt$$

Proof Selecting $\Psi(x_1) = k_1 x_1$ and $k = k_2$ yields $u = -k_2(x_2 + k_1 x_1)$ and leads to the conditions

- (5.22): $(\Phi(x_1) - k_1 x_1)x_1 \leq (\kappa - k_1)x_1^2 < 0, \quad x_1 \neq 0 \quad \Longleftrightarrow \quad k_1 > \kappa$
- (5.23): $0 < k_1 < k_2$

The cost functional follows directly from Proposition 5.2. □

Note how little information about the system nonlinearity Φ this control law is dependent of. Only an upper bound on its growth rate, κ, is needed. In particular, if Φ is known to lie in the second and fourth quadrants only, thus intuitively being useful for stabilizing x_1, we do not need any further information, since then $\kappa < 0$ and the parameter restriction $k_1 > 0$ becomes active.

Corollary 5.2 (Linearizing control)
Consider the system (5.19). *A globally stabilizing control, partially linearizing the dynamics, is given by*

$$u = -k_2(x_2 + k_1 x_1 + \Phi(x_1))$$

where

$$k_1 > \max\{0, -\min_{x_1} \Phi'(x_1)\} \tag{5.24}$$

and

$$k_2 > k_1 + \max_{x_1} \Phi'(x_1)$$

provided that such upper and lower bounds on Φ' *exist. In addition, for* $k_2 > 2(k_1 + \max_{x_1} \Phi'(x_1))$ *the control law minimizes a meaningful cost functional given by*

$$\int_0^\infty \Big((k_1 + \Phi'(x_1))k_1^2 x_1^2 + (\frac{k_2}{2} - k_1 - \Phi'(x_1))(x_2 + k_1 x_1 + \Phi(x_1))^2 + \frac{1}{2k_2} u^2 \Big) dt$$

Proof Selecting $\Psi(x_1) = k_1 x_1 + \Phi(x_1)$ and $k = k_2$ yields $u = -k_2(x_2 + k_1 x_1 + \Phi(x_1))$ and the conditions

- (5.22): $(\Phi(x_1) - k_1 x_1 - \Phi(x_1))x_1 = -k_1 x_1^2 < 0, \quad x_1 \neq 0 \iff k_1 > 0$
- (5.23): The first part of the inequality becomes

$$0 < k_1 + \Phi'(x_1), \quad x_1 \neq 0 \iff k_1 > -\min_{x_1 \neq 0} \Phi'(x_1)$$

  The second part of the inequality becomes

$$k_1 + \Phi'(x_1) < k_2, \quad x_1 \neq 0 \iff k_2 > k_1 + \max_{x_1 \neq 0} \Phi'(x_1)$$

The cost functional follows directly from Proposition 5.2. □

This control law corresponds to choosing the virtual control law

$$x_2^{\text{des}} = -\Phi(x_1) - k_1 x_1$$

which, as in feedback linearization, cancels the nonlinear dynamics of x_1, $\Phi(x_1)$, and replaces them with linear dynamics, $-k_1 x_1$. Condition (5.24) states that it is alright to cancel the natural dynamics, $\Phi(x_1)$, as long as the new dynamics, $-k_1 x$, provide at least the same amount of negative feedback. This is sound also from an optimality point of view. Intuitively, it must be suboptimal to waste feedback effort in order to slow down the natural system dynamics.

5.1.3 Flight control laws

Let us now return to the flight control context and apply the control laws derived in the previous section to the aircraft dynamics in (5.3). The boxed control laws below are the ones that will be implemented and evaluated in Chapter 7.

Angle of attack control

Let us first apply the linear control law in Corollary 5.1. Using Tables 5.1 and 5.2 for translation from x to the proper aircraft variables yields

$$u_2 = -k_{\alpha,2}(q_s + k_{\alpha,1}(\alpha - \alpha^{\text{ref}}) + f_\alpha(\alpha^{\text{ref}}, y_\alpha)) \tag{5.25}$$

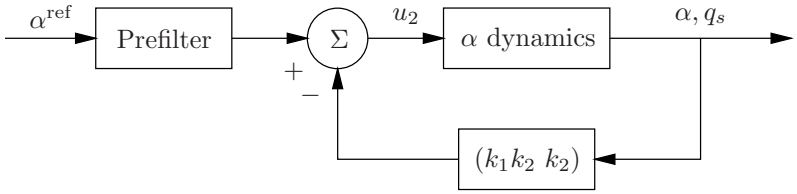


Figure 5.4 *The backstepping control law* (5.25) *moves the dependence on* C_L *outside the control loop, thereby enhancing the robustness.*

with f_α from (5.7). The control law is illustrated in Figure 5.4. Although implementing this control law requires knowledge of the lift force, and thereby the lift force coefficient, C_L, we note that the lift force dependent computation is performed in the prefilter *outside* the feedback loop. Therefore, imperfect knowledge of C_L does not jeopardize closed loop stability but only shifts the equilibrium.

The parameters should satisfy

$$k_{\alpha,2} > 2 \cdot k_{\alpha,1}, \quad k_{\alpha,1} > \max\{0, \kappa_\alpha\}$$

for the control law to be globally stabilizing and also minimize a meaningful cost functional. Here,

$$\kappa_\alpha = \max_{x_1} \frac{\Phi(x_1)}{x_1} = \max_{\alpha, \alpha^{\text{ref}}, y_\alpha} \frac{f_\alpha(\alpha, y_\alpha) - f_\alpha(\alpha^{\text{ref}}, y_\alpha)}{\alpha - \alpha^{\text{ref}}}$$

The maximum occurs when α^{ref} is chosen to be the point where f_α has the highest positive slope and α is selected infinitely close to α^{ref}. Then, the fraction above turns into the derivative w.r.t. α, i.e.,

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

In practice, f_α in (5.7) is dominated by the lift force term. The contribution to κ_α from the lift force becomes

$$\max_{\alpha, y_\alpha} -\frac{1}{m V_T \cos\beta} \frac{\partial L}{\partial \alpha}(\alpha) = \max_{\beta, V_T, h} \frac{\rho(h) V_T S}{2m \cos\beta} \cdot \max_{\alpha} -\frac{dC_L}{d\alpha}(\alpha)$$

recalling from (2.5) that

$$L = \bar{q} S C_L = \frac{1}{2} \rho(h) V_T^2 S C_L$$

Since $dC_L/d\alpha$ is negative in the post-stall region, see Figure 5.1, κ_α will be positive. This means that the higher speed V_T, and the larger sideslip β one wants the control

law to handle, the higher κ_α becomes, and the higher the control law parameters $k_{\alpha,1}$ and $k_{\alpha,2}$ must be chosen.

To solve this, we impose that $y_\alpha \in \Omega_{y_\alpha}$, where $\Omega_{y_\alpha} \subset \mathbb{R}^6$ is selected to represent the flight envelope of interest. For instance, a practically valid assumption may be that the sideslip is always less than 10 degrees. The final expression for κ_α then becomes

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

Let us now instead apply the linearizing control law in Corollary 5.2. Using Table 5.2 we get

$$u_2 = -k_{\alpha,2}(q_s + k_{\alpha,1}(\alpha - \alpha^{\text{ref}}) + f_\alpha(\alpha, y_\alpha)) \tag{5.26}$$

The only difference to (5.25) is that now f_α takes α as its first argument rather than α^{ref}. This causes the feedback loop to depend on C_L, and robustness against model errors in C_L becomes more difficult to analyze.

Somewhat surprisingly, the control law in (5.26) is identical[1] to the one in (2.15), which was derived using dynamic inversion and time-scale separation arguments. Using our Lyapunov based backstepping approach, we have thus shown this control law to be not only globally stabilizing, but also inverse optimal w.r.t. a meaningful cost functional, according to Corollary 5.2.

Sideslip regulation

Applying the linear control law in Corollary 5.1, using Table 5.2 along with (5.8) and (5.11) yields

$$u_3 = k_{\beta,2}(-r_s + k_{\beta,1}\beta + \frac{1}{V_T} g \cos\theta \sin\phi) \tag{5.27}$$

The parameters restrictions

$$k_{\beta,2} > 2 \cdot k_{\beta,1}, \quad k_{\beta,1} > \max\{0, \kappa_\beta\}$$

ensure the control law to be globally stabilizing and to be optimal w.r.t. a meaningful cost functional. Here,

$$\kappa_\beta = \max_{x_1} \frac{\Phi(x_1)}{x_1} = \max_{\beta, y_\beta} \frac{f_\beta(\beta, y_\beta) - f_\beta(0, y_\beta)}{\beta}$$

The first two terms of f_β in (5.8) both give negative contributions to κ_β. Since these typically are superior to the gravity contribution, $\kappa_\beta < 0$ holds and the parameter restrictions above simply reduce to

$$k_{\beta,2} > 2 \cdot k_{\beta,1} > 0$$

Thus, precise knowledge of the side force is not necessary to implement the globally stabilizing control law (5.27).

[1] The slight difference is due to that (2.15) was derived using somewhat simplified dynamics.

Stability axis roll control

Finally, controlling the stability axis roll is straightforward. Given the dynamics from (5.3a) and the roll rate command p_s^{ref}, simply assign

$$u_1 = k_{p_s}(p_s^{\text{ref}} - p_s) \tag{5.28}$$

where $k_{p_s} > 0$. This corresponds to ordinary proportional control.

5.1.4 Practical issues

Let us now turn to some practically relevant issues regarding the application of the derived flight control laws.

Tuning

How should the control law parameters k_{p_s}, $k_{\alpha,1}$, $k_{\alpha,2}$, $k_{\beta,1}$, and $k_{\beta,2}$ be selected? For roll control, k_{p_s} can be chosen to satisfy a given requirement on the roll time constant, which becomes $1/k_{p_s}$.

For α control, the closed loop system will not be linear since the nonlinear lift force L is not cancelled by the control law (5.25). However, since the control law is linear in α and q_s, it is tempting to still use linear techniques. A natural procedure is to linearize the angle of attack dynamics (5.3b)–(5.3c) around a suitable operating point and then select $k_{\alpha,1}$ and $k_{\alpha,2}$ to achieve some desired linear closed loop behavior locally around the operating point.

For β regulation, the situation is the same. Here, $k_{\beta,1}$ and $k_{\beta,2}$, determining the control law (5.27), can be selected by choosing some desired closed loop behavior using a linearization of the sideslip dynamics (5.3d)–(5.3e).

Saturation

The control laws (5.25), (5.27), and (5.28) all can handle a certain amount of gain reduction and still remain stabilizing. Thus, even in cases where actuator saturation makes the moment equation (5.5) infeasible, the control laws remain stabilizing within certain bounds. The maximum moment that can be produced depends on the aircraft state, see Figure 2.6. This makes it difficult to determine the exact part of the state space (in terms of α, β, etc.) where stability is guaranteed, and we will not further pursue this issue.

5.2 Flight Path Angle Control

We now turn to flight path angle control. This is not a standard autopilot function, but may still be of interest, e.g., for controlling the ascent or descent of an unmanned aerial vehicle.

5.2.1 Objectives, dynamics, and assumptions

The objective is for the flight path angle, depicted in Figure 2.7, to follow a given command, γ^{ref}. In other words, we want to make

$$\gamma = \gamma^{\text{ref}}$$

a globally asymptotically stable equilibrium.

We consider only the longitudinal motion of the aircraft, assuming that the roll and sideslip angles are zero. Again, speed control is assumed to be handled separately. The relevant dynamics are then given by (2.9b)–(2.9d). Using the definition of the flight path angle, $\gamma = \theta - \alpha$, yields

$$\begin{aligned} \dot{\gamma} &= \frac{1}{mV_T}(L + F_T \sin\alpha - mg\cos\gamma) \\ \dot{\theta} &= q \\ \dot{q} &= \frac{1}{I_y}(M + F_T Z_{TP}) \end{aligned} \tag{5.29}$$

Preparatory for the backstepping design, we make the following assumptions and simplifications.

A1. The lift force coefficient, C_L, is assumed to be a function of α alone.

A2. The time derivatives of the aircraft speed and altitude are neglected.

A3. The contribution to $\dot{\gamma}$ due to gravity is replaced by its value at the desired equilibrium, γ^{ref}. This means $\cos\gamma \approx \cos\gamma^{\text{ref}}$ is used. The idea is to make $\dot{\gamma}$ a function of $\alpha = \theta - \gamma$ alone.

A4. The control surface actuator dynamics are assumed to be fast enough to be disregarded.

Also introducing

$$u = \dot{q} = \frac{1}{I_y}(M + F_T Z_{TP}) \tag{5.30}$$

the aircraft dynamics (5.29) become

$$\dot{\gamma} = \frac{1}{mV_T}(L(\alpha) + F_T \sin\alpha - mg\cos\gamma^{\text{ref}}) \tag{5.31a}$$

$$\dot{\theta} = q \tag{5.31b}$$

$$\dot{q} = u \tag{5.31c}$$

The control law will be derived considering u as the input. The relationship to the true control input, δ, affecting the pitching moment, M, is given by

$$M(\delta) = I_y u - F_T Z_{TP} \tag{5.32}$$

In Chapter 7 we discuss how to solve for δ given u.

5.2.2 Backstepping control design

The nonlinear control problem

We begin by posing the control problem above as a more general nonlinear control problem. We first define new variables such that the origin becomes the desired equilibrium.

$$\begin{aligned} x_1 &= \gamma - \gamma^{\text{ref}} \\ x_2 &= \theta - (\gamma^{\text{ref}} + \alpha_0) \\ x_3 &= q \end{aligned} \tag{5.33}$$

$$\Phi(x_2 - x_1) = \Phi(\alpha - \alpha_0) = \frac{1}{mV_T}(L(\alpha) + F_T \sin\alpha - mg\cos\gamma^{\text{ref}})$$

α_0 is the angle of attack at steady state, solving $\dot{\gamma} = \Phi(0) = 0$. This gives us the dynamics

$$\dot{x}_1 = \Phi(x_2 - x_1) \tag{5.34a}$$

$$\dot{x}_2 = x_3 \tag{5.34b}$$

$$\dot{x}_3 = u \tag{5.34c}$$

considering the reference γ^{ref} as a constant.

The nonlinearity Φ is decided by the lift force, see Figure 5.1, and the contribution due to the thrust force. For $\alpha \lesssim 90$ degrees, a valid conclusion is therefore that Φ belongs to the first and third quadrants only. Mathematically, this can be written

$$x\Phi(x) > 0 \iff x\Phi(-x) < 0, \quad x \neq 0 \tag{5.35}$$

In the forthcoming design, we will show that this scarce information suffices to construct a globally stabilizing control law.

The backstepping design

The key idea of the forthcoming design is the following. For $x_2 = 0$, we get $\dot{x}_1 = \Phi(-x_1)$, which acts stabilizing since $\Phi(-x_1)$ is $\Phi(x_1)$ mirrored about the y-axis, and thus lies in the second and fourth quadrants. Using backstepping, we will show how to utilize this inherent stability property.

Below, k_i are constants which parameterize the control law, while c_i are "dummy constants" whose values will be assigned during the derivation to simplify various expressions.

Step 1: As usual, start by considering only the x_1-subsystem (5.34a). To find a globally stabilizing virtual control law, we use the clf

$$V_1 = \frac{1}{2}x_1^2 \tag{5.36}$$

Considering x_2 as our virtual control input

$$\begin{aligned}\dot{V}_1 &= x_1\Phi(x_2 - x_1)\\ &= x_1\Phi(-(1+k_1)x_1 + x_2 + k_1x_1)\\ &= x_1\Phi(-(1+k_1)x_1) < 0, \quad x_1 \neq 0\end{aligned}$$

can be achieved by selecting

$$x_2 = x_2^{\text{des}} = -k_1x_1, \quad k_1 > -1 \tag{5.37}$$

The fact that $k_1 = 0$ is a valid choice means that x_1 feedback is not necessary for the sake of stabilization. However, it provides an extra degree of freedom for tuning the closed loop performance.

Step 2: Since we cannot control x_2 directly, we continue by introducing the deviation from the virtual control law.

$$\tilde{x}_2 = x_2 - x_2^{\text{des}} = x_2 + k_1x_1$$

Including the x_2 dynamics (5.34b) we get

$$\begin{aligned}\dot{x}_1 &= \Phi(\xi)\\ \dot{\tilde{x}}_2 &= x_3 + k_1\Phi(\xi)\end{aligned}$$

where

$$\xi = -(1+k_1)x_1 + \tilde{x}_2 \tag{5.38}$$

has been introduced. We will also need

$$\dot{\xi} = -(1+k_1)\Phi(\xi) + x_3 + k_1\Phi(\xi) = -\Phi(\xi) + x_3$$

A regular backstepping design would proceed by expanding the clf (5.36) with a term penalizing $\tilde{x}_2$. We do this, but also add a term $F(\xi)$ as an extra degree of freedom, where F is required to be positive definite. Hence,

$$V_2 = \frac{c_1}{2}x_1^2 + \frac{1}{2}\tilde{x}_2^2 + F(\xi), \quad c_1 > 0$$

We compute its time derivative to find a new virtual control law, x_3^{des}.

$$\begin{aligned}\dot{V}_2 &= c_1x_1\Phi(\xi) + \tilde{x}_2(x_3 + k_1\Phi(\xi)) + F'(\xi)(-\Phi(\xi) + x_3)\\ &= (c_1x_1 + k_1\tilde{x}_2 - F'(\xi))\Phi(\xi) + (\tilde{x}_2 + F'(\xi))x_3^{\text{des}} + (\tilde{x}_2 + F'(\xi))(x_3 - x_3^{\text{des}})\end{aligned}$$

Although it may not be transparent, we can again find a stabilizing function independent of Φ. Choosing

$$x_3^{\text{des}} = -k_2\tilde{x}_2, \quad k_2 > 0 \tag{5.39}$$

$$F'(\xi) = c_2\Phi(\xi), \quad F(0) = 0, \quad c_2 > 0 \tag{5.40}$$

where (5.40) is an implicit but perfectly valid choice of F, yields

$$\dot{V}_2|_{x_3=x_3^{\text{des}}} = (\underbrace{c_1 x_1 + (k_1 - k_2 c_2)\tilde{x}_2}_{(k_1-k_2c_2)\xi})\Phi(\xi) - c_2\Phi(\xi)^2 - k_2\tilde{x}_2^2$$

To make the first term negative definite using (5.35), we select c_1 to make the factor in front of $\Phi(\xi)$ proportional to $-\xi$, see (5.38). This is achieved by

$$c_1 = -(1+k_1)(k_1 - k_2c_2), \quad k_2c_2 > k_1 \tag{5.41}$$

With this choice,

$$\dot{V}_2|_{x_3=x_3^{\text{des}}} = (k_1 - k_2c_2)\xi\Phi(\xi) - c_2\Phi(\xi)^2 - k_2\tilde{x}_2^2$$

becomes negative definite. The benefit of using the extra term $F(\xi)$ shows up in Equation (5.41). $F(\xi) \equiv 0$ (or equally, $c_2 = 0$) leads to $c_1 = -(1+k_1)k_1 > 0$ and the severe restriction $-1 < k_1 < 0$ implying positive feedback from x_1.

Step 3: The final backstepping iteration begins with introducing the third residual

$$\tilde{x}_3 = x_3 - x_3^{\text{des}} = x_3 + k_2\tilde{x}_2$$

We also update the system description

$$\begin{aligned}
\dot{x}_1 &= \Phi(\xi) \\
\dot{\tilde{x}}_2 &= \tilde{x}_3 - k_2\tilde{x}_2 + k_1\Phi(\xi) \\
\dot{\tilde{x}}_3 &= u + k_2(\tilde{x}_3 - k_2\tilde{x}_2 + k_1\Phi(\xi))
\end{aligned} \tag{5.42}$$

Furthermore,

$$\dot{\xi} = -\Phi(\xi) + \tilde{x}_3 - k_2\tilde{x}_2$$

V_3 is constructed by adding a term penalizing $\tilde{x}_3$ to V_2.

$$V_3 = c_3V_2 + \frac{1}{2}\tilde{x}_3^2, \quad c_3 > 0$$

We get

$$\begin{aligned}
\dot{V}_3 &= c_3\big[\underbrace{(k_1 - k_2c_2)\xi\Phi(\xi)}_{\text{negative definite}} - c_2\Phi(\xi)^2 - k_2\tilde{x}_2^2 + \tilde{x}_3(\tilde{x}_2 + c_2\Phi(\xi))\big] \\
&\quad + \tilde{x}_3\big[u + k_2(\tilde{x}_3 - k_2\tilde{x}_2 + k_1\Phi(\xi))\big] \\
&\leq -c_2c_3\Phi^2(\xi) - k_2c_3\tilde{x}_2^2 + \tilde{x}_3\big[u + k_2\tilde{x}_3 + (c_3 - k_2^2)\tilde{x}_2 + (k_1k_2 + c_2c_3)\Phi(\xi)\big]
\end{aligned}$$

once again using (5.35). Select $c_3 = k_2^2$ to cancel the $\tilde{x}_2\tilde{x}_3$ cross-term and try yet another linear control law.

$$u = -k_3\tilde{x}_3, \quad k_3 > k_2 \tag{5.43}$$

is a natural candidate and with this we investigate the resulting clf time derivative.

$$\dot{V}_3 \leq -k_2^2 c_2 \Phi^2(\xi) - k_2^3 \tilde{x}_2^2 - (k_3 - k_2)\tilde{x}_3^2 + (k_1 k_2 + k_2^2 c_2)\tilde{x}_3 \Phi(\xi)$$

In order to investigate the impact of the last cross-term, we complete the squares.

$$\begin{aligned}\dot{V}_3 \leq & -k_2^3 \tilde{x}_2^2 - (k_3 - k_2)(\tilde{x}_3 - \frac{k_1 k_2 + k_2^2 c_2}{2(k_3 - k_2)}\Phi(\xi))^2 \\ & - (k_2^2 c_2 - \frac{(k_1 k_2 + k_2^2 c_2)^2}{4(k_3 - k_2)})\Phi^2(\xi)\end{aligned}$$

$\dot{V}_3$ is negative definite provided that the $\Phi^2(\xi)$ coefficient is negative, which is true for

$$k_3 > k_2(1 + \frac{(k_1 + k_2 c_2)^2}{4 k_2 c_2}) \tag{5.44}$$

We now pick c_2 to minimize this lower limit under the constraints $c_2 > 0$ and $k_2 c_2 > k_1$.

For $k_1 \leq 0$, we can make $k_1 + k_2 c_2$ arbitrarily small whereby Equation (5.44) reduces to

$$k_3 > k_2$$

i.e., the same restriction as in (5.43). For $k_1 > 0$ the optimal strategy can be shown to be selecting c_2 arbitrarily close to the bound k_1/k_2. This yields

$$k_3 > k_2(1 + k_1) \tag{5.45}$$

Let us summarize this lengthy control law derivation, which resulted in a globally stabilizing control law (5.43) for the system (5.34), under the parameter restrictions in (5.37), (5.39), (5.43), and (5.45).

Proposition 5.3
Consider the system

$$\begin{aligned}\dot{x}_1 &= \Phi(x_2 - x_1) \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u\end{aligned} \tag{5.46}$$

where $x\Phi(x)$ is positive definite. A globally stabilizing control law is given by

$$u = -k_3(x_3 + k_2(x_2 + k_1 x_1)) \tag{5.47}$$

where

$$\begin{aligned}&k_1 > -1 \\ &k_2 > 0 \\ &k_3 > \begin{cases} k_2 & k_1 \leq 0 \\ k_2(1 + k_1) & k_1 > 0 \end{cases}\end{aligned} \tag{5.48}$$

□

The cascaded structure of the control law is illustrated in Figure 5.5.

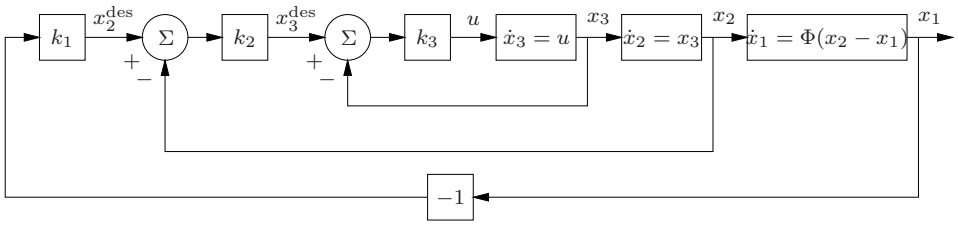


Figure 5.5 *The nonlinear system* (5.46) *can be globally stabilized through a cascaded control structure.*

Robustness

For this design we refrain from investigating inverse optimality due to the complicated expressions involved. Regardless of this, it is clear that the control law (5.47) gives a certain gain margin. E.g., in the case of $k_1 > 0$, we can afford a gain reduction of

$$\Gamma > \frac{k_2(1+k_1)}{k_3} \tag{5.49}$$

at the input (see Figure 4.1) without violating $\gamma k_3 > k_2(1+k_1)$ from (5.48).

Backstepping vs. feedback linearization

It is rewarding to compare the preceding backstepping design with a control design based on feedback linearization. Such a design makes the open loop system a chain of integrators by defining new coordinates according to

$$\begin{aligned}
\dot{x}_1 &= \Phi(x_2 - x_1) = \tilde{x}_2 \\
\dot{\tilde{x}}_2 &= \Phi'(x_2 - x_1)(x_3 - \tilde{x}_2) = \tilde{x}_3 \\
\dot{\tilde{x}}_3 &= \Phi''(x_2 - x_1)(x_3 - \tilde{x}_2)^2 + \Phi'(x_2 - x_1)(u - \tilde{x}_3) = \tilde{u}
\end{aligned}$$

We can now select

$$\tilde{u} = -\begin{pmatrix} k_1 & k_2 & k_3 \end{pmatrix} \begin{pmatrix} x_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \end{pmatrix}$$

to achieve any desired linear relationship between $\tilde{u}$ and x_1.

However, solving for the actual control input u to be produced, we get

$$u = \tilde{x}_3 + \frac{\tilde{u} - \Phi''(x_2 - x_1)(x_3 - \tilde{x}_2)^2}{\Phi'(x_2 - x_1)} \tag{5.50}$$

Two things are worth noting about this expression. Firstly, it depends not only on Φ (through $\tilde{x}_2$), but also on its first and second derivatives, which therefore must be known. In the aircraft control case, this corresponds to very accurate knowledge of the lift force, see (5.33). Since $L(\alpha)$ in practice comes with certain model error, especially at high angles of attack, the estimates of $L'(\alpha)$ and $L''(\alpha)$ may be poor. This means that the nonlinear system behavior cannot be cancelled completely. Unfortunately, it is difficult to analyze the robustness of (5.50), i.e., how incomplete cancellation of the nonlinearities affects the controlled system.

Secondly, Φ' is in the denominator of (5.50) implying that the control law has a singularity where $\Phi' = 0$. In the aircraft case, this occurs around the stall angle, where the lift force no longer increases with α, see Figure 5.1. Thus, global stability cannot be achieved using feedback linearization.

Our backstepping design did not suffer from any of the problems above, since all we required from Φ was for $x\Phi(x)$ to be positive definite, see (5.35).

5.2.3 Flight control law

We now return to the flight control context. Expressing (5.47) in the original coordinates using (5.33) gives us

$$u = -k_3(q + k_2(\theta + k_1(\gamma - \gamma^{\text{ref}}) - \gamma^{\text{ref}} - \alpha_0)) \tag{5.51}$$

This control law is globally stabilizing provided that k_1, k_2, and k_3 satisfy (5.48). Recall that α_0 is the angle of attack at steady state, solving $\dot{\gamma}$=0 in (5.31a).

5.2.4 Practical issues

We now investigate some practical issues regarding the application of this control law.

Tuning

For selecting the controller parameters, k_1, k_2, and k_3, we can use the same strategy as for tuning the general maneuvering control laws, cf. Section 5.1.4. This means linearizing the dynamics (5.31) around a suitable operating point and selecting the controller parameters to achieve some desired linear closed loop behavior locally around this operating point.

Saturation

As shown in Section 5.2.2, the control law (5.51) remains stabilizing in the presence of a gain certain reduction, given by (5.49) in the case $k_1 > 0$. Thus, even for a certain amount of control surface saturation, such that the desired pitching moment in (5.32) cannot be produced, the closed loop system is stable. As noted

in Section 5.1.4, the maximum moment, M, depends not only on the control surface deflections, δ, but also on the angle of attack, see Figure 2.6. Again, this makes is difficult to determine the part of the state space within which stability is guaranteed.

6

Adapting to Input Nonlinearities and Uncertainties

The flight control laws developed in Chapter 5 consider the angular accelerations as the control input, u. To find the corresponding control surface deflections, δ, the mapping from δ to u must be completely known. Typically this is not the case in practice. For example, the aerodynamic moment coefficients suffer from inevitable model errors, and the moment contributions from, e.g., the engine thrust, may not be measurable. Hence, it is necessary to add integral action in some form to the control laws to reach the desired equilibrium despite such model imperfections. This is the topic of the chapter.

In Section 6.1, we further illustrate the problem, and in Section 6.2, we present its mathematical formulation. Two solutions, based on adaptive backstepping and nonlinear observer techniques, respectively, are proposed in Sections 6.3 and 6.4. The two solutions are evaluated in Section 6.5 using a water tank example, and in Section 6.6 the adaptive schemes to be used for flight control are explicitly stated.

6.1 Background

Many of today's constructive nonlinear control design methods assume the control input to enter the system dynamics affinely, i.e., for the model to be of the form

$$\dot{x} = f(x) + g(x)u$$

In many practical cases this is not true. A common solution, see, e.g., [37, 47], is to find some other entity, a virtual control input v, that *does* enter the dynamics linearly, and that depends on the true control input u through a static mapping. Using, e.g., backstepping or feedback linearization, a globally stabilizing control law $v = k(x)$ can then be derived. These virtual control inputs are often physical entities like forces, torques, or flows, while the true input might be the deflection of a control surface in a flight control case or the throttle setting in an engine control case.

The remaining problem, how to find which actual control input u to apply, is often very briefly discussed, typically assuming that the mapping from u to v is completely known and invertible. In this chapter we investigate the case where the mapping is only partially known. It might be that the true mapping is too complex to identify, or that other sources than u contribute to v. Friction might for example reduce the net torque in a robot control case. Here, we will pragmatically model the discrepancy between the model and the true mapping as a constant bias. We propose two different ways of adapting to the bias, and for each case, the issue of closed loop stability is investigated.

6.2 Problem Formulation

Consider a single input system of the form

$$\begin{aligned} \dot{x} &= f(x) + Bv \\ v &= g(x, u) \end{aligned} \tag{6.1}$$

$x \in \mathbb{R}^n$ is the measurable state vector and $u \in \mathbb{R}$ is the true control input.

$$B = \begin{pmatrix} 0 & \cdots & 0 & 1 \end{pmatrix}^T \tag{6.2}$$

is such that only the last state, x_n, is directly affected by the control input through

$$\dot{x}_n = f_n(x) + g(x, u)$$

Assume that the mapping, $g(x, u)$, from the true control input, u, to the virtual control input, v, is not completely known but only a model, $\hat{g}(x, u)$, such that

$$g(x, u) = \hat{g}(x, u) + e$$

The model error, $e = g(x, u) - \hat{g}(x, u)$, is modeled as a constant. This pragmatic assumption may be more or less realistic but allows us to correct for biases and reach the correct equilibrium at steady state. With this we can rewrite (6.1) as

$$\dot{x} = f(x) + B(w + e) \tag{6.3a}$$

$$w = \hat{g}(x, u) \tag{6.3b}$$

w is the part of the virtual control input, v, that we are truly in control of.

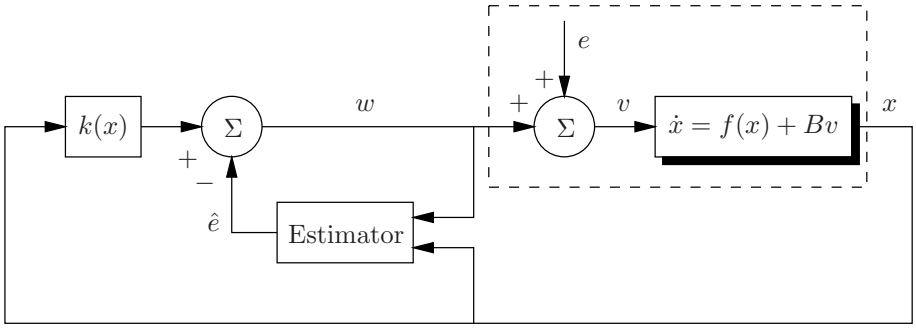


Figure 6.1 *Illustration of the certainty equivalence controller* (6.7)*. The shadowed box represents the dynamic system to be controlled. The dashed box makes up the system that the control designer actually faces, since e is not known.*

From a preceding control design, a control law

$$v = k(x) \tag{6.4}$$

is assumed to be known such that the origin is a GAS equilibrium of

$$\dot{x} = f(x) + Bk(x)$$

We also assume that a Lyapunov function $V(x)$ for the closed loop system is known, such that

$$\dot{V}(x) = V_x(x)(f(x) + Bk(x)) = -W(x) \tag{6.5}$$

where $W(x)$ is positive definite.

Given e, (6.4) could be realized by solving

$$w = v - e \iff \hat{g}(x, u) = k(x) - e \tag{6.6}$$

for u. How do we deal with the fact that e is not available? A straightforward solution is to rely on one of the corner stones of adaptive control and use the certainty equivalence [46] of (6.6). This means that we replace the unknown parameter e by an estimate $\hat{e}$ and form

$$w = \hat{g}(x, u) = k(x) - \hat{e} \tag{6.7}$$

Figure 6.1 illustrates the approach. The strategy is intuitively appealing but leads to two important questions:

- How do we estimate e?
- Can we retain global stability using $\hat{e}$ for feedback?

Two approaches to the problem will be pursued. In Section 6.3, we will use standard adaptive backstepping techniques to find an estimator that will guarantee closed loop stability without having to adjust the control law (6.7). In Section 6.4, the starting point is that a converging estimator is given. The question then is how to adjust the control law to retain stability. This approach is due to the author, but was inspired by the observer backstepping techniques introduced by Kanellakopoulos et al. [39].

6.3 Adaptive Backstepping

Adaptive backstepping [46] deals with the unknown parameter e by extending the Lyapunov function $V(x)$ with a term penalizing the estimation error $\tilde{e} = e - \hat{e}$:

$$V_a(x, \tilde{e}) = V(x) + \frac{1}{2\gamma}\tilde{e}^2, \quad \gamma > 0$$

By cleverly selecting the update rule

$$\dot{\hat{e}} = \tau(x, \hat{e})$$

closed loop stability can be guaranteed. To see this we investigate $\dot{V}_a$ when (6.7) is used as feedback. Using (6.2) and (6.5) we get

$$\begin{aligned} \dot{V}_a &= V_x(x)\big(f(x) + B(k(x) - \hat{e} + e)\big) - \frac{1}{\gamma}\tilde{e}\,\tau(x, \hat{e}) \\ &= -W(x) + \tilde{e}\Big(\frac{\partial V(x)}{\partial x_n} - \frac{1}{\gamma}\tau(x, \hat{e})\Big) \end{aligned} \tag{6.8}$$

The first term is negative definite according to the assumptions, while the second, mixed term is indefinite. Since $\tilde{e}$ is not available, the best we can do is to cancel the second term by selecting

$$\tau(x, \hat{e}) = \tau(x) = \gamma\frac{\partial V(x)}{\partial x_n} \tag{6.9}$$

The resulting closed loop system becomes

$$\begin{aligned} \dot{x} &= f(x) + B(k(x) + \tilde{e}) \\ \dot{\tilde{e}} &= -\gamma\frac{\partial V(x)}{\partial x_n} \end{aligned} \tag{6.10}$$

which satisfies

$$\dot{V}_a(x, \tilde{e}) = -W(x)$$

Despite $\dot{V}_a$ only being negative semidefinite, the origin, $x = 0$, $\tilde{e} = 0$, is GAS according to Corollary 3.1. Since only $x = 0$ solves $\dot{V}_a = 0$, $\tilde{e} = 0$ must also hold for $\dot{x}$ to be zero.

If V is quadratic in x_n, $V = \ldots + \frac{1}{2}x_n^2$, the control law (6.7) becomes

$$w = k(x) - \gamma \int_0^t x_n(s)ds$$

In this case, estimating e and using the estimate for feedback corresponds to adding integral action from x_n.

6.4 Observer Based Adaption

In adaptive backstepping, the estimator was a consequence of assigning a negative Lyapunov time derivative. In this section, we first design an estimator and then investigate how to possibly adjust the control law (6.7).

6.4.1 The general case

The idea is to regard e as an unknown but constant state variable. Augmenting the original dynamics (6.3a) with this extra state yields

$$\begin{aligned} \dot{x} &= f(x) + B(w + e) \\ \dot{e} &= 0 \end{aligned}$$

Although this system is nonlinear, we can design an observer for e with linear error dynamics, since the nonlinearity, f, is a function of the measurable states x only [44].

We will use x_n alone to measure of the goodness of the estimate. In the presence of measurement noise, it would be advantageous to utilize the entire state vector, which is also fully possible. The nonlinear observer becomes

$$\frac{d}{dt}\begin{pmatrix} \hat{x}_n \\ \hat{e} \end{pmatrix} = \begin{pmatrix} f_n(x) + w + \hat{e} \\ 0 \end{pmatrix} + \begin{pmatrix} k_1 \\ k_2 \end{pmatrix}(x_n - \hat{x}_n) \tag{6.11}$$

With this, the dynamics of the estimation error

$$\epsilon = \begin{pmatrix} x_n - \hat{x}_n \\ e - \hat{e} \end{pmatrix}$$

become linear:

$$\dot{\epsilon} = \begin{pmatrix} -k_1 & 1 \\ -k_2 & 0 \end{pmatrix}\epsilon = A_\epsilon \epsilon \tag{6.12}$$

For any asymptotically stabilizing observer gains k_1 and k_2, making A_ϵ Hurwitz[1], we can find a positive definite matrix P such that

$$\frac{d}{dt}\epsilon^T P\epsilon = -\epsilon^T Q\epsilon \leq -q\tilde{e}^2$$

by solving the Riccati equation

$$A_\epsilon^T P + PA_\epsilon = -Q, \quad Q = qI, \quad q > 0$$

according to basic linear systems theory, see Rugh [62]. A_ϵ is Hurwitz if and only if

$$k_1 > 0, \quad k_2 > 0$$

To investigate the closed loop stability, we combine the original Lyapunov function $V(x)$ with $\epsilon^T P\epsilon$ and form

$$V_o(x,\epsilon) = V(x) + \epsilon^T P\epsilon$$

We also augment the control law (6.7) with an extra term, l, to be decided, to compensate for using $\hat{e}$ for feedback.

$$w = k(x) + l(x,\hat{e}) - \hat{e} \tag{6.13}$$

yields

$$\begin{aligned}\dot{V}_o &= V_x(x)\big(f(x) + B(k(x) + l(x,\hat{e}) - \hat{e} + e)\big) - \epsilon^T Q\epsilon \\ &\leq -W(x) + \frac{\partial V(x)}{\partial x_n}(l(x,\hat{e}) + \tilde{e}) - q\tilde{e}^2\end{aligned}$$

By choosing

$$l(x,\hat{e}) = l(x) = -\lambda\frac{\partial V(x)}{\partial x_n}, \quad \lambda > 0 \tag{6.14}$$

we can complete the squares.

$$\dot{V}_o \leq -W(x) - \lambda(\frac{\partial V(x)}{\partial x_n} - \frac{1}{2\lambda}\tilde{e})^2 - (q - \frac{1}{4\lambda})\tilde{e}^2$$

To achieve GAS, we must satisfy $q - \frac{1}{4\lambda} > 0$, which can always be done once λ in (6.14) has been selected, since q is at our disposal.

Let us summarize our discussion.

Proposition 6.1 (Observer based adaption)
Consider the system

$$\begin{aligned}\dot{x}_i &= f_i(x), \quad i = 1,\ldots,n-1 \\ \dot{x}_n &= f_n(x) + w + e\end{aligned} \tag{6.15}$$

[1] A matrix is said to be Hurwitz if all its eigenvalues are in the open left half plane.

where $x \in \mathbb{R}$ is the measurable state vector, $w \in \mathbb{R}$ is the control input, and $e \in \mathbb{R}$ is an unknown constant. Let $w = k(x) - e$ be a globally stabilizing control law and let $V(x)$ be a Lyapunov function for the controlled system such that $\dot{V}(x)$ is negative definite.

Then, the control law

$$w = k(x) - \lambda \frac{\partial V(x)}{\partial x_n} - \hat{e}, \quad \lambda > 0$$

where $\hat{e}$ is produced by the observer

$$\frac{d}{dt}\begin{pmatrix}\hat{x}_n \\ \hat{e}\end{pmatrix} = \begin{pmatrix}f_n(x) + w + \hat{e} \\ 0\end{pmatrix} + \begin{pmatrix}k_1 \\ k_2\end{pmatrix}(x_n - \hat{x}_n), \quad k_1 > 0,\ k_2 > 0$$

is also globally stabilizing. □

An interesting feature is displayed by computing an explicit expression of the estimate produced. Using (6.12) we have that

$$\hat{e}(t) = e - \tilde{e} = e - \begin{pmatrix}0 & 1\end{pmatrix} e^{A_\epsilon t}\epsilon(0)$$

This means that $\hat{e}$ evolves independently of the control input $u(t)$ and the state trajectory $x(t)$. This is not true for the adaptive backstepping estimate.

6.4.2 The optimal control case

Let us consider the case where the original, unattainable control law (6.4) solves an optimal control problem of the form

$$V(x) = \min_v \int_0^\infty q(x) + r(x)v^2 dt \tag{6.16}$$

where $r(x) \leq r_0$ for all x. According to (4.6) it then holds that

$$k(x) = -\frac{1}{2r(x)}\frac{\partial V(x)}{\partial x_n}$$

We recall from Section 4.3 that a fundamental property of control laws minimizing a criterion like (6.16) is that they have a gain margin of $[\frac{1}{2}, \infty]$. This inherent robustness means that we do not need to modify the certainty equivalence control law (6.7) to retain stability, since $l(x)$ in Equation (6.14) is proportional to $k(x)$. To show this we make the split

$$k(x) = \underbrace{-\frac{1}{2r(x)}\frac{\partial V(x)}{\partial x_n} + \lambda\frac{\partial V(x)}{\partial x_n}}_{\tilde{k}(x)} \underbrace{-\lambda\frac{\partial V(x)}{\partial x_n}}_{l(x)}$$

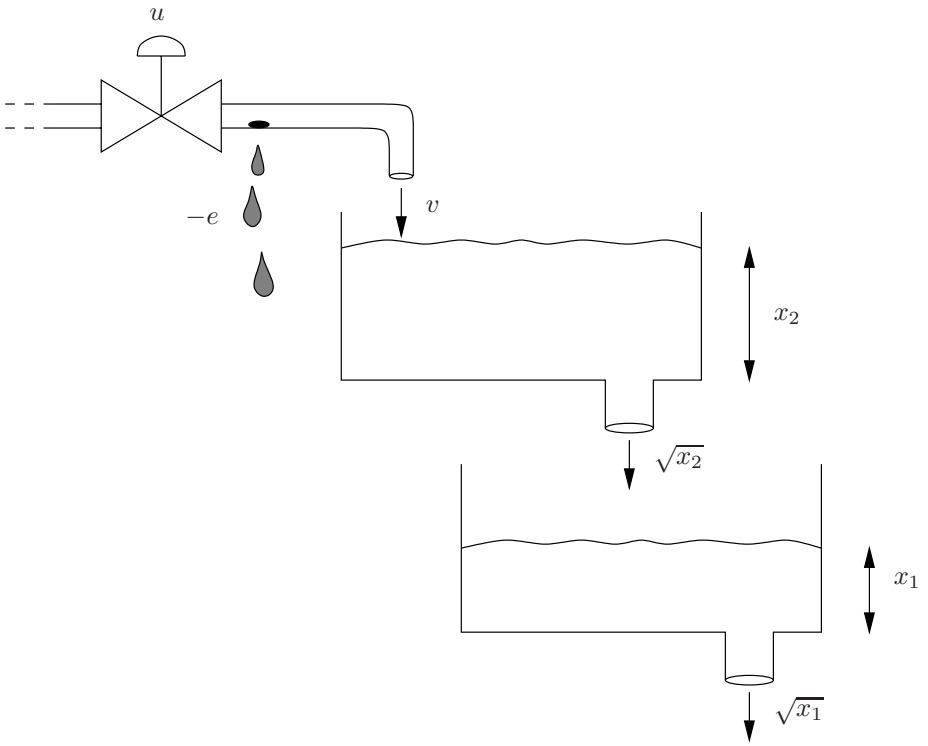


Figure 6.2 *Two tanks connected in series.*

Now,

$$v = \tilde{k}(x) = -(\frac{1}{2r(x)} - \lambda)\frac{\partial V(x)}{\partial x_n}$$

is guaranteed to be globally stabilizing, i.e., to make $W(x)$ in (6.5) negative definite given that the gain reduction due to λ is less than 50%. Thus,

$$\frac{1}{2r(x)} - \lambda \geq \frac{1}{2} \cdot \frac{1}{2r(x)} \iff \lambda \leq \frac{1}{4r_0} \leq \frac{1}{4r(x)}$$

must hold which does not contradict the only previous requirement from (6.14) that $\gamma > 0$.

An intuitive interpretation of this result is that some of the optimal control effort can be sacrificed in order to compensate for using the estimate $\hat{e}$ for feedback.

6.5 A Water Tank Example

Let us apply the two strategies to a practical example to investigate their pros and cons. Consider the two tanks in Figure 6.2. The control goal is to achieve a

certain water level r in the bottom tank. Using Bernoulli's equation and setting all constants to unity, the system dynamics become

$$\dot{x}_1 = -\sqrt{x_1} + \sqrt{x_2}$$
$$\dot{x}_2 = -\sqrt{x_2} + v$$

where x_1 = water level of the lower tank, x_2 = water level of the upper tank, and v = incoming water flow. v is produced by changing the aperture of the valve of the input pipe.

We assume the dynamics of the valve to be very fast compared to the dynamics of the tanks, so that the relationship between the commanded aperture radius, u, and the water flow, v, can be regarded as static. Assuming some external water supply to keep a constant pressure, v will be proportional to the aperture opening area, which in turn depends on u^2. Again setting all constants to unity we would have $v = u^2$. In order to be able to account for a possible model error in this static relationship and for other sources contributing to the net inflow, e.g., leakage, we assign the model

$$v = u^2 + e$$

in accordance with (6.3a).

The first step is to find a globally stabilizing control law $v = k(x)$. We do this using an ad hoc Lyapunov approach. At the desired steady state, $x_1 = x_2 = r$. Therefore consider the control Lyapunov function

$$V(x) = \frac{a}{2}(x_1 - r)^2 + \frac{1}{2}(x_2 - r)^2, \quad a > 0$$

Compute its time derivative:

$$\dot{V}(x) = a(x_1 - r)(-\sqrt{x_1} + \sqrt{x_2}) + (x_2 - r)(-\sqrt{x_2} + k(x))$$

By collecting the beneficial terms and cancelling the indefinite ones, one finds that

$$k(x) = \sqrt{r} + \frac{a}{\sqrt{x_2} + \sqrt{r}}(r - x_1) + b(r - x_2), \quad a > 0,\ b \geq 0 \tag{6.17}$$

yields $\dot{V}(x) = -W(x)$ where

$$W(x) = a(x_1 - r)(\sqrt{x_1} - \sqrt{r}) + (x_2 - r)(\sqrt{x_2} - \sqrt{r}) + b(x_2 - r)^2$$

is positive definite.

Let us now evaluate the expressions involved with the two approaches for adapting to the leakage. The adaptive backstepping update rule (6.9) for estimating e becomes

$$\dot{\hat{e}} = \gamma \frac{\partial V(x)}{\partial x_2} = \gamma(x_2 - r), \quad \gamma > 0 \tag{6.18}$$

$k(x)$	Adaptive backstepping	Observer based adaption
$a = 1$ $b = 0.5$	$\gamma = 0.3$	$k_1 = 1$ $k_2 = 0.5$ $\lambda = 0$

Table 6.1 *Controller parameter values used in the simulations.*

With this, the implicit control law (6.7) becomes

$$w = u^2 = k(x) + \gamma \int_0^t (r - x_2(s))ds$$

Using the observer based approach, the estimator can be designed according to (6.11). For the actual implementation, we can rewrite this as

$$\frac{d}{dt}\begin{pmatrix}\hat{x}_n \\ \hat{e}\end{pmatrix} = \begin{pmatrix}-k_1 & 1 \\ -k_2 & 0\end{pmatrix}\begin{pmatrix}\hat{x}_n \\ \hat{e}\end{pmatrix} + \begin{pmatrix}1 \\ 0\end{pmatrix}(-\sqrt{x_2} + w)$$

which can be implemented using, e.g., SIMULINK. The implicit control law (6.13) becomes

$$w = u^2 = k(x) + \lambda(r - x_2) - \hat{e}, \quad \lambda > 0$$

If $b > 0$ was selected in the control law (6.17), we do not have to add the term $\lambda(r - x_2)$ for the sake of stability, since it can be seen as a part of $k(x)$ already. As in the optimal control case treated in Section 6.4.2, closed loop is then guaranteed using the original certainty equivalence control law (6.7) without any modification.

In the simulations, the parameter values for the two adaptive controllers were selected according to Table 6.1. The initial water level, which is also fed to the observer, is 1 in both tanks. The control goal is to for x_1 to reach the reference level $r = 4$ and maintain this despite the leakage $e = -3$ starting at $t = 25$ s. Figure 6.3 shows the actual control input and the water level of the lower tank when no adaption is used. Figures 6.4 and 6.5 show the results of applying adaptive backstepping and observer based adaption, respectively.

There is a striking difference between the initial behaviors of the two leakage estimates. As pointed out in Section 6.4.1, the observer $\hat{e}$ estimate evolves independently of u and x. Since $\epsilon(0) = 0$, the estimation error remains zero until the leakage starts. The adaptive backstepping estimate on the other hand depends on the integral of $r - x_2$ over time, causing an oscillatory behavior due to the initial error of the upper tank water level.

Also, in the presence of actuator saturation, adaptive backstepping will suffer from the windup problems that generally occur when using integral action in the feedback loop. This is avoided with the observer based approach if the observer is fed with the true, saturated value of the control input.

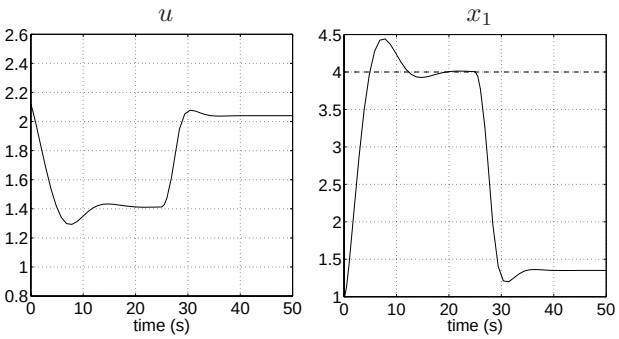


Figure 6.3 *No adaption, pure state feedback.*

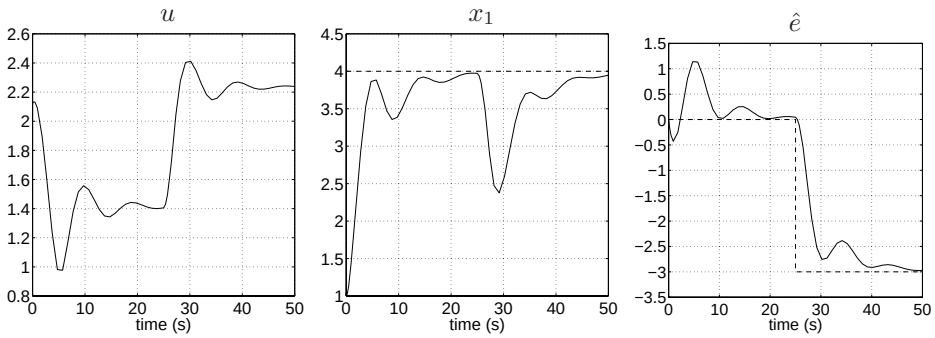


Figure 6.4 *Adaptive backstepping.*

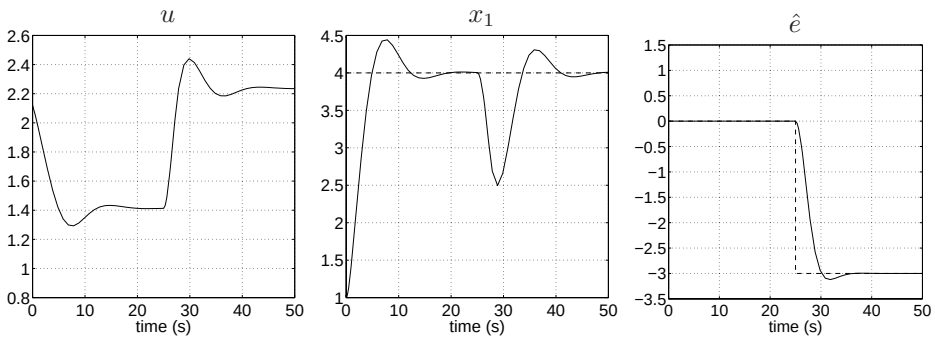


Figure 6.5 *Observer based adaption.*

6.6 Application to Flight Control

Considering the appealing properties of the observer based adaption, as demonstrated in the previous section, this will be our choice of adaptive scheme for flight control. All of the state feedback laws to be used along with the observers, (5.25), (5.27), (5.28), and (5.51), can afford a certain amount of gain reduction and still be globally stabilizing, as shown in Chapter 5. As shown in Section 6.4.2, this means that the certainty equivalence control law (6.7) is globally stabilizing.

Let us now state the resulting observers when Proposition 6.1 is applied to the flight dynamics used for control design in Chapter 5.

6.6.1 General maneuvering

The relevant dynamics are given by (5.3). To handle unmodeled nonlinearities and uncertainties in the mapping (5.4) from δ to u, we redefine (5.3a), (5.3c), and (5.3e) as

$$\begin{aligned}\dot{p}_s &= u_1 + e_1\\ \dot{q}_s &= u_2 + e_2\\ \dot{r}_s &= u_3 + e_3\end{aligned}$$

Comparing these dynamics to (6.15) yields the observers

$$\begin{aligned}\frac{d}{dt}\begin{pmatrix}\hat{p}_s\\ \hat{e}_1\end{pmatrix} &= \begin{pmatrix}u_1 + \hat{e}_1\\ 0\end{pmatrix} + L_1(p_s - \hat{p}_s)\\ \frac{d}{dt}\begin{pmatrix}\hat{q}_s\\ \hat{e}_2\end{pmatrix} &= \begin{pmatrix}u_2 + \hat{e}_2\\ 0\end{pmatrix} + L_2(q_s - \hat{q}_s)\\ \frac{d}{dt}\begin{pmatrix}\hat{r}_s\\ \hat{e}_3\end{pmatrix} &= \begin{pmatrix}u_3 + \hat{e}_3\\ 0\end{pmatrix} + L_3(r_s - \hat{r}_s)\end{aligned} \tag{6.19}$$

L_1, L_2, and L_3 are 2×1-vectors whose entries must be positive for the estimates to converge.

6.6.2 Flight path angle control

The dynamics are given by (5.31). As above, (5.31c) is redefined as

$$\dot{q} = u + e$$

to handle unmodeled nonlinearities and uncertainties in the mapping (5.30) from δ to u. Using Proposition 6.1, an observer for e is given by

$$\frac{d}{dt}\begin{pmatrix}\hat{q}\\ \hat{e}\end{pmatrix} = \begin{pmatrix}u + \hat{e}\\ 0\end{pmatrix} + L(q - \hat{q}) \tag{6.20}$$

where L is a 2×1-vector with positive entries.

7

Implementation and Simulation

The control designs of the preceding chapters were based on a number of pragmatic, simplifying assumptions regarding the aircraft dynamics and the types of model errors and uncertainties to be handled. Using computer simulations we will now evaluate the control laws experimentally.

The aircraft models used for simulation are presented in Section 7.1. Some implementation details are covered in Section 7.2, while Section 7.3 is devoted to the actual computer simulations.

7.1 Aircraft Simulation Models

Due to the complexity of the dynamics and the military nature of the field, there exist only a few available aircraft simulation models. Three of these are listed in Table 7.1.

7.1.1 GAM/ADMIRE

The Generic Aerodata Model (GAM) contains aerodynamic data for a small fighter aircraft, not unlike JAS 39 Gripen [63]. The model was produced and made available by Saab AB, Linköping, Sweden. A model description is given by Backström [5], and the package can be downloaded from [64]. A disadvantage of the GAM is

Simulation model	Publicly available	Fighter aircraft dynamics
GAM/ADMIRE	Yes	Yes
HIRM	Not in general	Yes
FDC	Yes	No

Table 7.1 *Three existing environments for aircraft simulation.*

that aerodynamic data is only recorded for $-10^\circ \leq \alpha \leq 30^\circ$ and $-20^\circ \leq \beta \leq 20^\circ$. In this flight envelope, the aerodynamic efforts are not significantly nonlinear, see Figure 2.6. Basic facts regarding the GAM can be found in Appendix 7.A.

To perform flight simulations based on the GAM, there exists a SIMULINK wrapper called ADMIRE. We will use the GAM/ADMIRE environment to evaluate the general maneuvering control laws developed in Section 5.1. The GAM was recently used by Cronander [12] to evaluate a pitch rate controller based on dynamic inversion.

7.1.2 HIRM

The High Incidence Research Model (HIRM) was developed by DERA[1] of the United Kingdom. The model is based on aerodynamic data from wind tunnel tests and drop tests of a small-scale model. The HIRM was then derived by scaling up these data to create an aircraft of F-18 proportions, see Appendix 7.A. Aerodynamic data exist for a wide range of angles of attack and sideslip angles, $-50^\circ \leq \alpha \leq 120^\circ$ and $-50^\circ \leq \beta \leq 50^\circ$.

The HIRM was used as a benchmark fighter aircraft model in the robust flight control design challenge [53] initiated by GARTEUR[2]. A technical description of the HIRM, which is implemented as SIMULINK model, can be found in [56].

We will use the HIRM to evaluate the flight path angle control law developed in Section 5.2. DERA is gratefully acknowledged for granting permission to use the model for this purpose.

7.1.3 FDC

The Flight Dynamics and Control toolbox (FDC) by Rauw [59] is a SIMULINK toolbox for general flight dynamics and control analysis. The toolbox comes with aerodata from a small, non-military aircraft, but the modular structure of the toolbox allows the user to plug in external aerodata of his or her choice. Although promising, FDC has not been used for simulation since for the GAM and HIRM aerodata, well functioning interfaces already exist.

[1] Defence Evaluation and Research Agency
[2] Group for Aeronautical Research and Technology in EURope

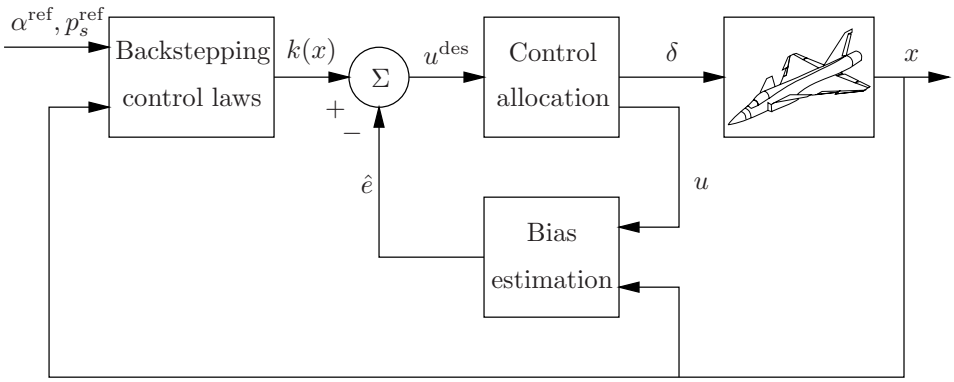


Figure 7.1 *Controller configuration.*

7.2 Controller Implementation

The controller configuration is shown in Figure 7.1. The backstepping block contains the state feedback control laws derived in Chapter 5, while the bias estimator block contains the nonlinear observers from Chapter 6, which are used to estimate and adapt to unmodeled moments acting on the aircraft. The control allocation block will be discussed in Section 7.2.1. Its function is to translate the desired angular acceleration,

$$u^{\text{des}} = k(x) - \hat{e}$$

into actual control surface deflections, δ. The signal u, which is fed to the estimator, is the actual angular acceleration, which differs from u^{des} when u^{des} is not feasible. By feeding the actual value u to the estimator we avoid wind-up problems.

The observers used are compactly stated in Section 6.6. Let us for convenience also gather the control laws from Chapter 5. For general maneuvering the control laws are made up by (5.25), (5.27), and (5.28). Using $k(x)$ to denote these control laws, we get

$$k(x) = \begin{pmatrix} k_1(x) & k_2(x) & k_3(x) \end{pmatrix}^T$$

where

$$\begin{aligned} k_1(x) &= k_{p_s}(p_s^{\text{ref}} - p_s) \\ k_2(x) &= -k_{\alpha,2}(q_s + k_{\alpha,1}(\alpha - \alpha^{\text{ref}}) + f_\alpha(\alpha^{\text{ref}}, y_\alpha)) \\ k_3(x) &= k_{\beta,2}(-r_s + k_{\beta,1}\beta + \frac{1}{V_T} g \cos\theta \sin\phi) \end{aligned}$$

with f_α from (5.7). The flight path angle control law (5.51) reads

$$k(x) = -k_3(q + k_2(\theta + k_1(\gamma - \gamma^{\text{ref}}) - \gamma^{\text{ref}} - \alpha_0))$$

Let us make two remarks regarding the implementation of these controllers.

- In the control designs in Chapter 5, the force effects due to the control surface deflections and the angular rates were neglected. Here, these effects are included when f_α and α_0 are computed in order to improve the accuracy.
- α_0, the steady state angle of attack, is obtained by solving $\dot{\gamma} = 0$ in (5.31a) at each time step, using current values of the variables involved. Although this ad hoc approach to determine α_0 is not guaranteed to converge, successful simulations provide an alibi.

7.2.1 Control allocation

The control designs in Chapters 5 and 6 considered the angular accelerations of the aircraft as the control inputs, see (5.4) and (5.30). In (5.5) and (5.32) we solved for the actual moments to be produced. In most nonlinear aircraft designs it is assumed that the control surface deflections, δ, affect these moments linearly. In practice this is not true, since the aerodynamic moments produced by the control surfaces also suffer from stall effects similar to the ones that are well known for the lift force. This can be seen in Figure 2.6 where the pitching coefficient, C_m, tends to saturate for low and high values of the symmetrical elevon deflection, δ_{es}.

Here, we will take into account the nonlinear mapping from δ to the aerodynamic moments and propose numerical algorithms for finding the proper δ given the moments to be produced. The implementations used are due to Press et al. [58].

General maneuvering

The equation to be solved for δ is given by (5.5):

$$\mathbf{M}(\delta) = IR_{sb}^T u^{\text{des}} + \omega \times I\omega$$

Using the definition of $\mathbf{M}$ from Section 2.4, $\mathbf{M}(\delta)$ can be translated into the desired aerodynamic moment coefficients,

$$C^{\text{des}} = \begin{pmatrix} C_l^{\text{des}} & C_m^{\text{des}} & C_n^{\text{des}} \end{pmatrix}^T$$

Introducing

$$C(\delta) = \begin{pmatrix} C_l(\delta) & C_m(\delta) & C_n(\delta) \end{pmatrix}^T \tag{7.1}$$

the control allocation problem now becomes to find δ such that

$$C(\delta) = C^{\text{des}} \tag{7.2}$$

For simplicity we will only make use of the elevons and the rudder and ignore the canard wings, see Figure 2.4. Thus, $\delta = (\delta_{es},\ \delta_{ed},\ \delta_r)$. To handle cases where these

control surfaces saturate, and (7.2) cannot be satisfied, we reformulate the control allocation problem as an optimization problem:

$$\delta = \arg\min_{\delta \in \Omega_\delta} J(\delta)$$
$$\text{where } J(\delta) = \|C(\delta) - C^{\text{des}}\|^2$$

where Ω_δ refers to the set of feasible control surface settings.

Nonlinear optimization is a complex matter and in general no guarantee can be given that the global optimum is found. However, since in our case the optimization is performed at each sample time, we can use the solution at one time step as the initial guess at the next time step. This strategy of each time starting very close to the optimal solution yields very good performance.

In the implementation, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) variable metric method [13], a quasi-Newton method, was chosen. This takes $J(\delta)$ and the gradient $\nabla J(\delta)$ as arguments, where the latter can be numerically computed. It recursively estimates the inverse Hessian of J and performs a line search in this direction to update δ. Reusing the inverse Hessian at the next time step significantly improves the performance.

Flight path angle control

The equation to be solved for δ is given by (5.32):

$$M(\delta) = I_y u^{\text{des}} - F_T Z_{TP}$$

where M corresponds to the aerodynamic pitching moment to be produced. Recalling that $M = \bar{q} S \bar{c} C_m$ we can explicitly solve for the desired pitching coefficient:

$$C_m^{\text{des}} = \frac{1}{\bar{q} S \bar{c}} (I_y u^{\text{des}} - F_T Z_{TP})$$

Again we ignore the canard wings for simplicity, and only make use of the symmetrical elevon deflection, δ_{es}. The control allocation problem then becomes solving

$$C_m(\delta_{es}) = C_m^{\text{des}} \iff C_m(\delta_{es}) - C_m^{\text{des}} = 0$$

for δ_{es}. Unlike above, we will use this formulation for finding δ_{es} and not use an optimization framework. In this scalar case, actuator saturation can be handled separately.

The pitching coefficient, C_m, is usually a complicated function of the arguments involved. However, given measurements of α, q, etc., C_m becomes a close to monotone function of δ_{es}, see Figure 2.6. The only exception is when the aforementioned stall effects occur and the moment produced saturates. However, given the aerodata, these regions can be manually removed by virtually saturating δ_{es} when C_m saturates.

Many numerical solvers are well adapted to finding the zero of a monotone scalar function. For the implementation, the Van Wijngaarden-Dekker-Brent method

[8, 19] was chosen. This method is a happy marriage between bisection, which ensures convergence, and inverse quadratic interpolation, which provides superlinear convergence in the best-case scenario.

7.3 Simulation

We now turn to the actual simulations performed. Before showing the actual plots in Section 7.3.3, Section 7.3.1 gives an account of the conditions surrounding the simulations, and the controller parameters used are listed in Section 7.3.2.

7.3.1 Conditions

The simulations are performed under the following general conditions:

- The simulation models, GAM/ADMIRE and HIRM, are both equipped with sensor dynamics and actuator dynamics mimicing those encountered in a real aircraft.
- No noise is added to the models, i.e., we assume perfect sensing (besides the dynamics of the sensors) and no external disturbances like wind gusts.
- The dynamics of the aircraft model are the fully nonlinear dynamics introduced in Section 2.4 and not the simplified ones used for control design in Chapter 5 (see Sections 5.1.1 and 5.2.1). Specifically this means that the control surface deflections affect not only the aerodynamic moments but also the aerodynamic forces.
- The aerodynamic moment coefficients used for control allocation as described in Section 7.2.1 agree with the ones in the aircraft models as far as possible. Since, e.g., $\dot{\alpha}$ and $\dot{\beta}$ are not measured, the effects of these are neglected in the control allocation schemes.
- Speed control is handled by separate controllers. For the HIRM, the speed controller by Larsson [49] is used, and for the GAM/ADMIRE, a simple PI-controller has been implemented.
- The sensor information used by the flight control system is updated at a frequence of 50 Hz for the GAM/ADMIRE and 80 Hz for the HIRM. The control surface demands are sent to the aircraft at the same frequency.

7.3.2 Controller parameters

The controller parameters used for the simulations are given in Tables 7.2 and 7.3. The state feedback parameters k_{p_s}, $k_{\alpha,1}$, $k_{\alpha,2}$, $k_{\beta,1}$, and $k_{\beta,2}$ have been chosen according to the guidelines in Sections 5.1.4 and 5.2.4, to achieve suitable linear dynamics around the initial states of the flight cases presented below. The observer gains L_1, L_2, and L_3 place the poles of the error dynamics (6.12) in $-8 \pm i$, while L gives the observer poles $-2 \pm i$.

p_s control		α control		β control	
k_{p_s}	2.0	$k_{\alpha,1}$	2.0	$k_{\beta,1}$	2.0
		$k_{\alpha,2}$	5.0	$k_{\beta,2}$	5.0
L_1	$\begin{pmatrix}16.0 & 65.0\end{pmatrix}^T$	L_2	$\begin{pmatrix}16.0 & 65.0\end{pmatrix}^T$	L_3	$\begin{pmatrix}16.0 & 65.0\end{pmatrix}^T$

Table 7.2 *General maneuvering control parameters.*

γ control	
k_1	0.4
k_2	1.2
k_3	2.1
L	$\begin{pmatrix}4.0 & 5.0\end{pmatrix}^T$

Table 7.3 *Flight path angle control parameters.*

7.3.3 Simulation results

General maneuvering

To evaluate the general maneuvering control laws from Section 5.1 we use the GAM/ADMIRE environment. The simulations are performed at an initial speed of 0.5 Mach and at an altitude of 1000 m. The assessment maneuvers are the following:

M1. Roll rate demand, $p_s^{\text{ref}} = 150$ deg/s. See Figure 7.2.

M2. Angle of attack demand, $\alpha^{\text{ref}} = 15$ deg. See Figure 7.3.

M3. M1 and M2 performed simultaneously. See Figure 7.4. During this maneuver, both α and p_s vary rapidly which means that assumption A3 in Section 5.1.1 is violated.

Let us make some comments regarding the simulation results.

- During M1 and M2, the controlled variables, p_s, α, and β all follow their reference trajectories (given by the dashed curves) well.
- The increase in the aircraft speed, V_T, during M1 is due to our strategy of rolling about the stability x-axis, rather than the body x-axis, as discussed in Section 2.2. Since β is small, the stability x-axes coincides with the velocity vector. Initially, $\alpha = 2.8$ deg is necessary to have the lift force make up for gravity. But after rolling 180 deg without changing α, the same amount of lift force is directed towards Earth, which causes the aircraft to dive and the speed to increase.
- In M2, we see that p_s and β are not completely unchanged despite the maneuver being constrained to the body xz-plane. The small perturbations are

caused by the feedback from the bias estimators. This is also the reason for the small perturbations of the δ_{ed} and δ_r input signals.

- Shifting our attention to M3 we see that the resulting aircraft trajectory is not quite the superposition of the M1 and M2 trajectories, in terms of the controlled variables. The roll rate response is still satisfactory, but α and β oscillate more. The reason is (at least) twofold:

  1. According to assumption A3 in Section 5.1.1, the β controller assumes α to be constant, and vice versa. Since this is not the case here, the roll axis will not coincide exactly with the stability axis as desired, see Figure 2.3. As outlined in Section 2.2, this means α and β are no longer decoupled, but during the roll, part of the angle of attack turns into sideslip and vice versa.
  2. During the initial phase of the step, the parameter estimates, $\hat{e}_1$, $\hat{e}_2$, and $\hat{e}_3$ oscillate. The 0.2 peak in $\hat{e}_2$ causes the α controller to believe some external source is contributing to the pitching moment. Consequently, the pitching moment produced by δ_{es} is reduced, which leads to a reduction in q and that the increase in α is temporarily stopped. Efforts have been made to tune the observers to avoid these oscillations but the ultimate solution is yet to be found.

Flight path angle control

The flight path angle control law from Section 5.2 is evaluated using the HIRM. Here, the initial aircraft state is level flight at Mach 0.3 at an altitude of 1524 m (5000 ft). A model error in the pitch coefficient, C_m, of -0.03 is introduced on purpose to examine the robustness of the controller. This is the same error that was used for evaluating the controllers in the GARTEUR robust flight control design challenge [53]. The following assessment maneuver is used:

M4. Two consecutive flight path angle demands, $\gamma^{\text{ref}} = 25$ deg followed by $\gamma^{\text{ref}} = 15$ deg. See Figure 7.5.

Let comment on the simulation results.

- Overall, the flight path angle, γ, follows its reference trajectory well.
- The small initial dip of γ is caused by the model error introduced, which makes the true pitching moment less than the controller expects. Soon however, the model error is estimated by the observer, as seen in the $\hat{e}$ plot, and the controller compensates for the error and brings γ back to zero after 4 seconds.
- The maximum angle of attack is 38 deg which is greater than the HIRM stall angle, see Figure 5.1. In accordance with the GAS property of the control law shown in Section 5.2.2, this does not cause any stability problems.

- During the steep ascent just after 5 s, the elevons saturate at -40 deg, which means that the desired angular acceleration, u^{des}, cannot be produced. The actual u can be computed by numerically differentiating q from sensor data. Doing so, the maximum gain reduction during the saturation period is given by

$$\min_t \frac{u}{u^{\text{des}}} \approx 0.55$$

In Section 5.2.2, the backstepping control law was shown to have a certain amount of gain margin. Using the parameter values of Table 7.3, the bound (5.49) becomes

$$\frac{k_2(1+k_1)}{k_3} = 0.8$$

Thus, despite violating (5.49), the system converges to the desired state[3]. This indicates that the parameter restrictions (5.48) are sufficient but not always necessary for closed loop stability.

[3]One should also note that the robustness results in Chapter 5 only regard the state feedback laws, not including the observer feedback.

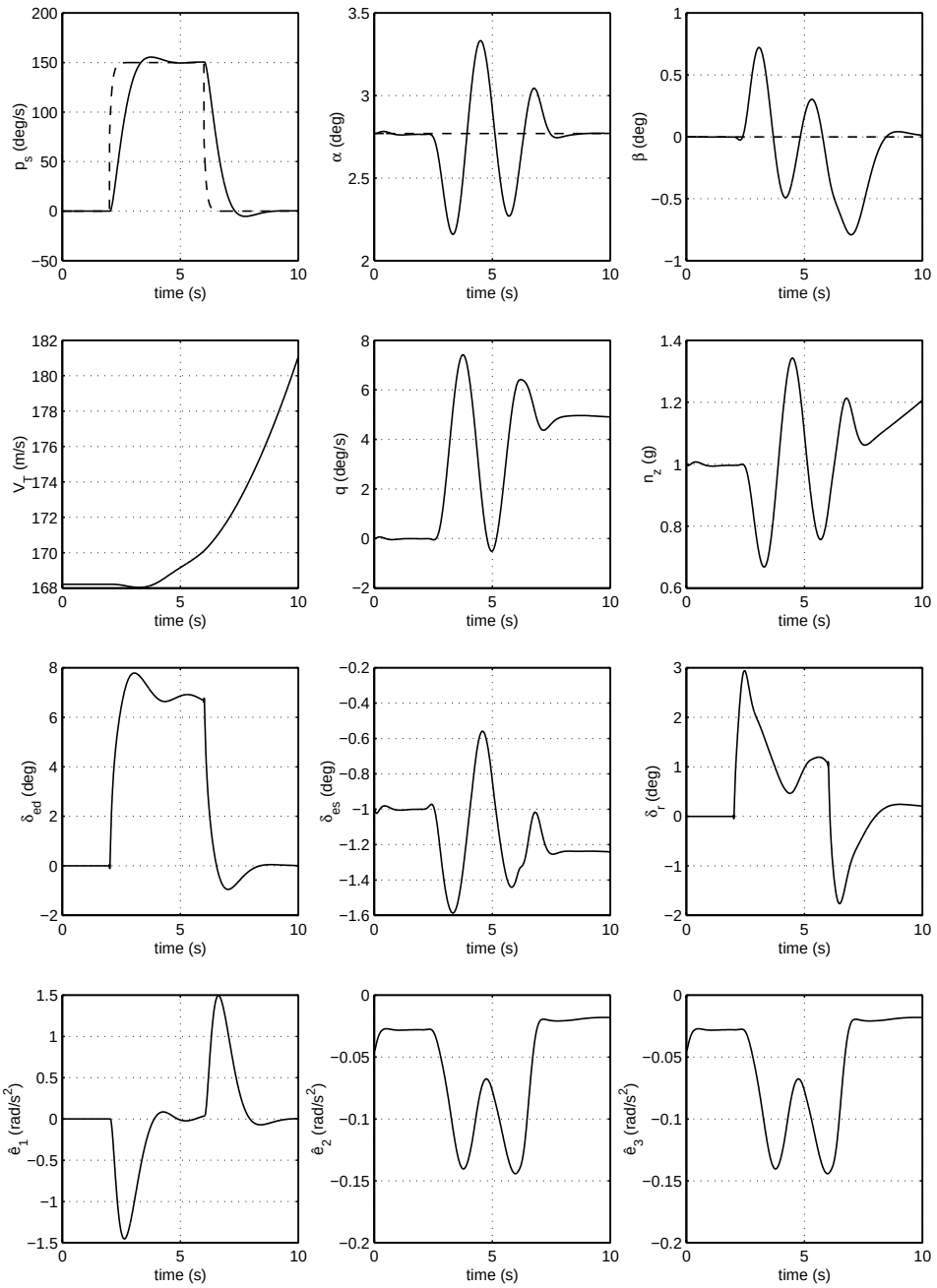


Figure 7.2 *Assessment maneuver M1: roll rate demand,* $p_s^{ref} = 150°/s$.

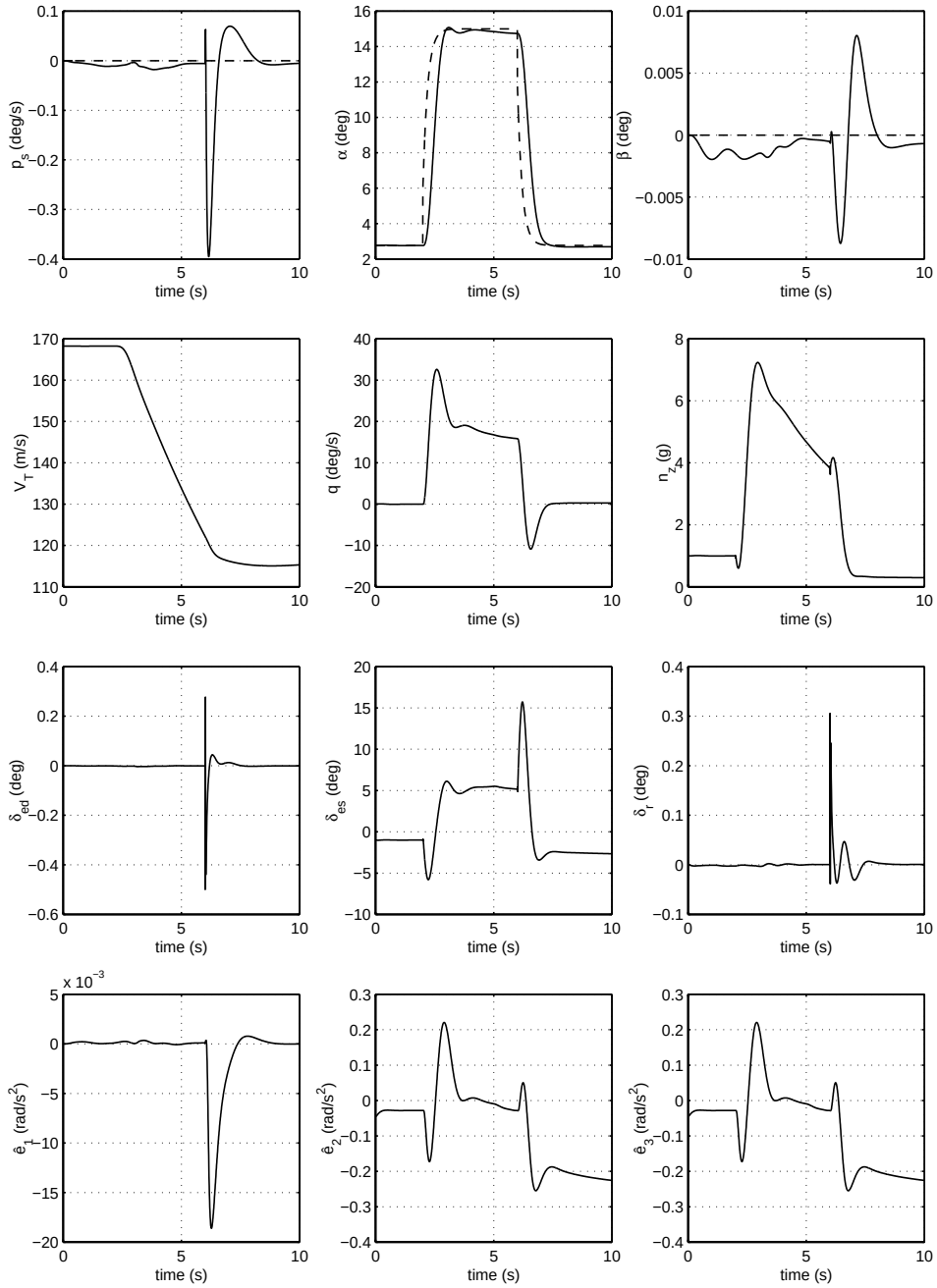


Figure 7.3 *Assessment maneuver M2: angle of attack demand,* $\alpha^{ref} = 15^\circ$.

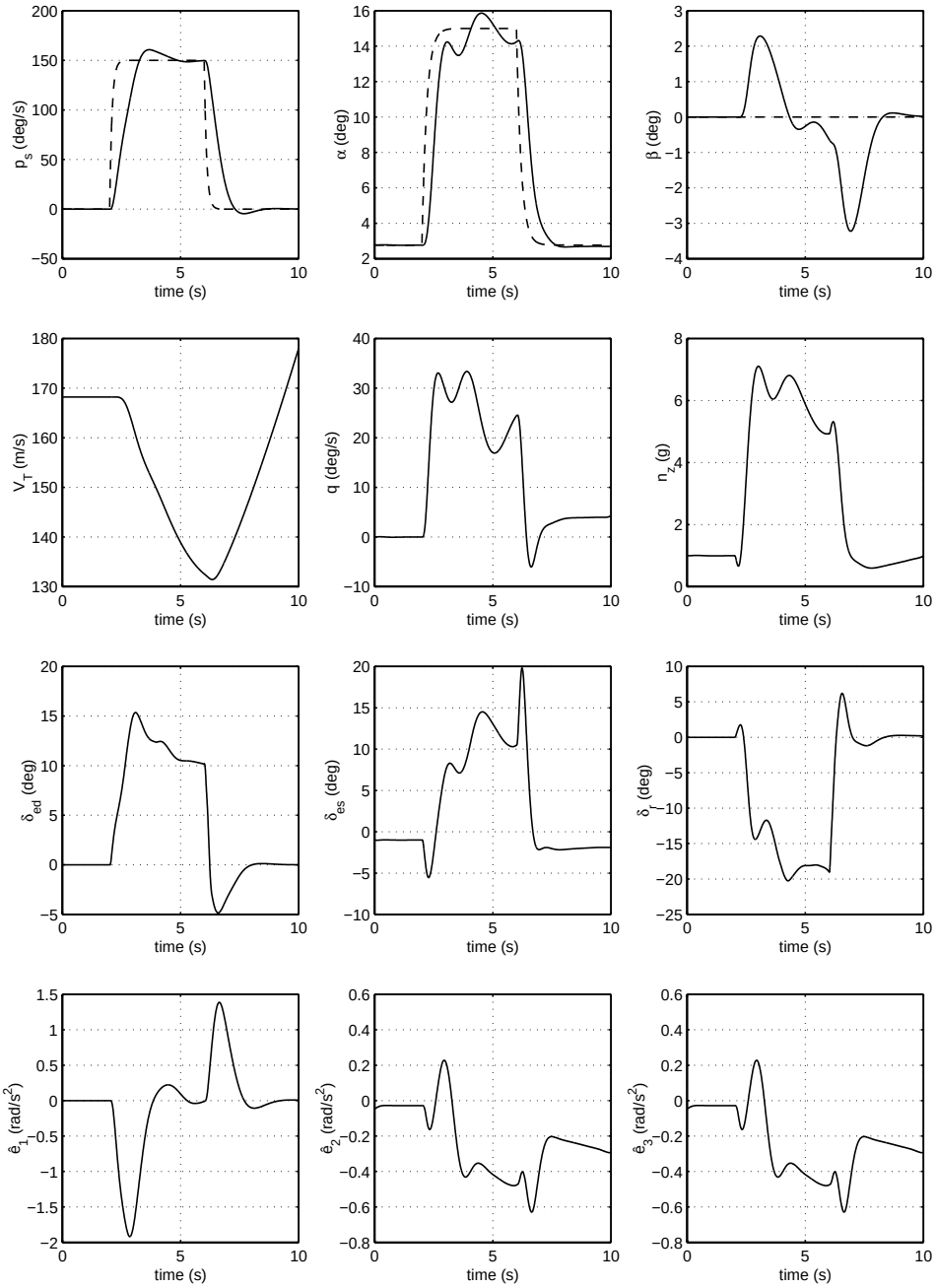


Figure 7.4 *Assessment maneuver M3: M1 and M2 combined.*

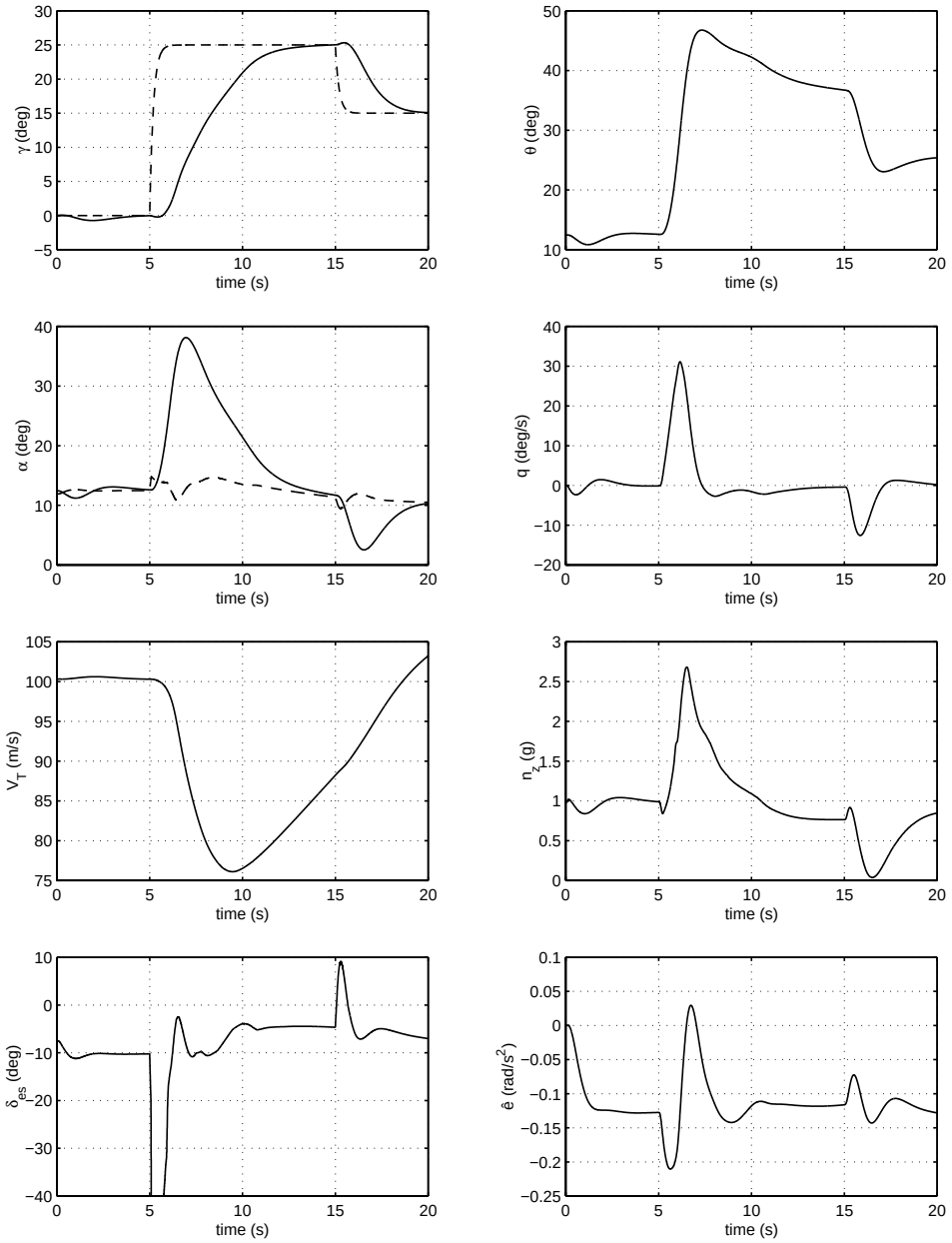


Figure 7.5 *Assessment maneuver M4: flight path angle demands,* $\gamma^{ref} = 25^\circ$ *followed by* $\gamma^{ref} = 15^\circ$. *The dashed line in the* α *plot represents* α_0.

Appendix

7.A Aircraft Data

Aircraft model properties for the GAM [5] and for the HIRM [56].

Entity			GAM	HIRM
mass	m	kg	9100.0	15296.0
moment of inertia	I_x	$\mathrm{kg\,m^2}$	21000.0	
	I_y	$\mathrm{kg\,m^2}$	81000.0	163280.0
	I_z	$\mathrm{kg\,m^2}$	101000.0	
	I_{xz}	$\mathrm{kg\,m^2}$	2500.0	
wing planform area	S	m	45.0	37.16
wing span	b	m	10.0	11.4
mean aerodynamic chord	$\bar{c}$	m	5.20	3.511

8

Conclusions

In this thesis, we have investigated backstepping as a possible framework for aircraft flight control design. The main result is that by recognizing the naturally stabilizing parts of the aerodynamic lift and side forces, globally stabilizing control laws can be designed which rely on less information of these forces than corresponding NDI designs, based on feedback linearization.

To conclude, let us evaluate the proposed control laws with respect to five major issues, relevant to all control designs, namely *stability*, *tuning*, *robustness*, *input saturation*, and *disturbance attenuation*.

Stability Backstepping deals explicitly with stability, through the construction of a Lyapunov function for the closed loop system along with the construction of the control law itself. The derived state feedback flight control laws yield asymptotic stability within the entire flight envelope of practical interest, including high angles of attack.

Tuning Since our backstepping designs focus on utilizing useful nonlinearities rather than cancelling them, the resulting closed loop systems are not linear. This means that the aircraft response to the pilot inputs will not be independent of, e.g., the angle of attack. However, we have shown that the nominal performance, valid around an operating point of interest, can be tuned according to one's requirements.

Robustness Robustness profits greatly from our backstepping approach. Since cancellation of the nonlinear aerodynamic lift and side forces, $L(\alpha)$ and $Y(\beta)$, is avoided, complete knowledge of these forces is not required. We have also proposed adaptation techniques for handling model errors in the aerodynamic moments.

Input saturation Our backstepping control laws all possess a certain gain margin. Thus, stability is retained for a certain amount of control surface saturation leading to a reduction of the moments produced.

Disturbance attenuation A main weakness of nonlinear control design in general, including backstepping, is the lack of tools to quantitatively analyze the effects of disturbances, e.g., wind gusts and sensor noise in our aircraft control case. Fortunately, as reported by Enns et al. [15], a flight controller giving a properly selected bandwidth generally does a good job of suppressing such disturbances.

Bibliography

[1] Richard J. Adams, James M. Buffington, and Siva S. Banda. Design of nonlinear control laws for high-angle-of-attack flight. *Journal of Guidance, Control, and Dynamics*, 17(4):737–746, 1994.

[2] Muthana T. Alrifai, Joe H. Chow, and David A. Torrey. A backstepping nonlinear control approach to switched reluctance motors. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 4652–4657, December 1998.

[3] David Anderson and Scott Eberhardt. How airplanes fly: A physical description of lift. *Sport Aviation*, February 1999.
`http://www.allstar.fiu.edu/aero/airflylvL3.htm`.

[4] Zvi Artstein. Stabilization with relaxed controls. *Nonlinear Analysis, Theory, Methods & Applications*, 7(11):1163–1173, 1983.

[5] Hans Backström. Report on the usage of the Generic Aerodata Model. Technical report, Saab Aircraft AB, May 1997.

[6] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 1995.

[7] Jean-Luc Boiffier. *The Dynamics of Flight: The Equations*. John Wiley & Sons, 1998.

[8] R. P. Brent. *Algorithms for Minimization Without Derivatives*. Prentice-Hall, 1973.

[9] M. M. Bridges, D. M. Dawson, and C. T. Abdallah. Control of rigid-link, flexible-joint robots: A survey of backstepping approaches. *Journal of Robotic Systems*, 12(3):199–216, 1995.

[10] James J. Carroll, Andrew J. Geoghan, Darren M. Dawson, and Praveen Vedagarbha. A backstepping based computed torque controller for switched reluctance motors driving inertial loads. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 779–786, September 1995.

[11] James J. Carroll, Manfred Schneider, and Darren M. Dawson. Integrator backstepping techniques for the tracking control of permanent magnet brush DC motors. In *Conference Record of the 1993 IEEE Industry Applications Society Annual Meeting*, pages 663–671, October 1993.

[12] Carin Cronander. Control applications in aeronautics. Licentiate thesis 99-23, Department of Aeronautics, Royal Institute of Technology, 1999.

[13] J. E. Dennis, Jr. and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. SIAM, 1996.

[14] Wayne C. Durham, Frederick H. Lutze, and William Mason. Kinematics and aerodynamics of the velocity vector roll. *Journal of Guidance, Control, and Dynamics*, 17(6):1228–1233, November–December 1994.

[15] Dale Enns, Dan Bugajski, Russ Hendrick, and Gunter Stein. Dynamic inversion: an evolving methodology for flight control design. *International Journal of Control*, 59(1):71–91, January 1994.

[16] Bernard Etkin and Lloyd Duff Reid. *Dynamics of Flight: Stability and Control*. John Wiley & Sons, third edition, 1996.

[17] Kenan Ezal, Zigang Pan, and Petar V. Kokotović. Locally optimal and robust backstepping design. *IEEE Transactions on Automatic Control*, 45(2):260–270, February 2000.

[18] Dan Fontaine and Petar Kokotović. Approaches to global stabilization of a nonlinear system not affine in control. In *Proceedings of the 1998 American Control Conference*, pages 1645–1647, June 1998.

[19] G. E. Forsythe, M. A. Malcolm, and C. B. Moler. *Computer Methods for Mathematical Computations*. Prentice-Hall, 1976.

[20] Jonas Fredriksson. Nonlinear control of turbocharged diesel engines. Licentiate thesis 329L, Control Engineering Laboratory, Department of Signals and Systems, Chalmers University of Technology, 1999.

[21] Randy A. Freeman and Petar V. Kokotović. *Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*. Birkhäuser, 1996.

[22] Randy A. Freeman and James A. Primbs. Control Lyapunov functions: New ideas from an old source. In *Proceedings of the 35th Conference on Decision and Control*, pages 3926–3931, December 1996.

[23] William L. Garrard, Dale F. Enns, and S. Anthony Snell. Nonlinear feedback control of highly manoeuvrable aircraft. *International Journal of Control*, 56 (4):799–812, 1992.

[24] S. Torkel Glad. Robustness of nonlinear state feedback - a survey. *Automatica*, 23(4):425–435, 1987.

[25] Åslaug Grøvlen and Thor I. Fossen. Nonlinear control of dynamic positioned ships using only position feedback: An observer backstepping approach. In *Proceedings of the 35th Conference on Decision and Control*, pages 3388–3393, December 1996.

[26] James K. Hall and Meir Pachter. Formation maneuvers in three dimensions. In *Proceedings of the 39th Conference on Decision and Control*, December 2000.

[27] Ola Härkegård and S. Torkel Glad. A backstepping design for flight path angle control. In *Proceedings of the 39th Conference on Decision and Control*, pages 3570–3575, Sydney, Australia, December 2000.

[28] Ola Härkegård and S. Torkel Glad. Control of systems with input nonlinearities and uncertainties: an adaptive approach. Technical Report LiTH-ISY-R-2302, Department of Electrical Engineering, Linköpings universitet, SE-581 83 Linköping, Sweden, October 2000.

[29] Ola Härkegård and S. Torkel Glad. Flight control design using backstepping. Technical Report LiTH-ISY-R-2323, Department of Electrical Engineering, Linköpings universitet, SE-581 83 Linköping, Sweden, December 2000.

[30] HARV. The NASA High Angle-of-Attack Research Vehicle homepage. `http://www.dfrc.nasa.gov/Projects/HARV`.

[31] John Hauser and Ali Jadbabaie. Aggressive maneuvering of a thrust vectored flying wing: A receding horizon approach. In *Proceedings of the 39th Conference on Decision and Control*, December 2000.

[32] W. B. Herbst. Future flight technologies. *Journal of Aircraft*, 17(8):561–566, August 1980.

[33] J. Hu, D. M. Dawson, and K. Anderson. Position control of a brushless DC motor without velocity measurements. *IEE Proceedings of Electric Power Applications*, 142(2):113–122, June 1995.

[34] Jun Hu, Darren M. Dawson, and Yi Qian. Position tracking control for robot manipulators driven by induction motors without flux measurements. *IEEE Transactions on Robotics and Automation*, 12(3):419–438, June 1996.

[35] Chien H. Huang and Gareth J. Knowles. Application of nonlinear control strategies to aircraft at high angles of attack. In *Proceedings of the 29th Conference on Decision and Control*, pages 188–193, December 1990.

[36] Alberto Isidori. *Nonlinear Control Systems*. Springer, third edition, 1995.

[37] Mrdjan Jankovic, Miroslava Jankovic, and Ilya Kolmanovsky. Constructive Lyapunov control design for turbocharged diesel engines. *IEEE Transactions on Control Systems Technology*, 8(2):288–299, March 2000.

[38] Zhong-Ping Jiang and Henk Nijmeijer. Tracking control of mobile robots: A case study in backstepping. *Automatica*, 33(7):1393–1399, 1997.

[39] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse. A toolkit for nonlinear feedback design. *Systems & Control Letters*, 18(2):83–92, February 1992.

[40] Hassan K. Khalil. *Nonlinear Systems*. Prentice-Hall, second edition, 1996.

[41] Petar Kokotović. Constructive nonlinear control: Progress in the 90's. In *IFAC 1999 Proceedings*, pages 49–77, 1999.

[42] Petar Kokotović, Hassan K. Khalil, and John O'Reilly. *Singular Perturbation Methods in Control: Analysis and Design*. Academic Press, 1986.

[43] Petar V. Kokotović. The joy of feedback: Nonlinear and adaptive. *IEEE Control Systems Magazine*, 12(3):7–17, June 1992.

[44] Arthur J. Krener and Alberto Isidori. Linearization by output injection and nonlinear observers. *Systems & Control Letters*, 3:47–52, June 1983.

[45] Miroslav Krstić, Dan Fontaine, Petar V. Kokotović, and James D. Paduano. Useful nonlinearities and global stabilization of bifurcations in a model of jet engine surge and stall. *IEEE Transactions on Automatic Control*, 43(12): 1739–1745, December 1998.

[46] Miroslav Krstić, Ioannis Kanellakopoulos, and Petar Kokotović. *Nonlinear and Adaptive Control Design*. John Wiley & Sons, 1995.

[47] Miroslav Krstić and Petar V. Kokotović. Lean backstepping design for a jet engine compressor model. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 1047–1052, 1995.

[48] Stephen H. Lane and Robert F. Stengel. Flight control design using non-linear inverse dynamics. *Automatica*, 24(4):471–483, 1988.

[49] Tomas Larsson. Linear quadratic design and μ-analysis of a fighter stability augmentation system. Master's thesis, Linköpings universitet, 1997.

[50] J. Lévine. Are there new industrial perspectives in the control of mechanical systems? In Paul M. Frank, editor, *Advances of Control: Highlights of ECC'99*, chapter 7, pages 197–226. Springer, 1999.

[51] Johan Löfberg. Backstepping with local LQ performance and global approximation of quadratic performance. In *Proceedings of the American Control Conference*, pages 3898–3902, June 2000.

[52] A. M. Lyapunov. *The General Problem of the Stability of Motion*. Taylor & Francis, 1992. English translation of the original publication in Russian from 1892.

[53] Jean-François Magni, Samir Bennani, and Jan Terlouw, editors. *Robust Flight Control: A Design Challenge*. Springer, 1997.

[54] Donald McLean. *Automatic Flight Control Systems*. Prentice Hall, 1990.

[55] S. K. Mudge and R. J. Patton. Variable structure control laws for aircraft manoeuvres. In *International Conference on Control, 1988*, pages 564–568, 1988.

[56] E. A. M. Muir et al. Robust flight control design challenge problem formulation and manual: The high incidence research model (HIRM). Technical Report TP-088-4, Group for Aeronautical Research and Technology in EURope GARTEUR-FM(AG08), 1997.

[57] Robert C. Nelson. *Flight Stability and Automatic Control*. McGraw-Hill, second edition, 1998.

[58] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*. Cambridge University Press, second edition, 1992.

[59] Marc O. Rauw. *FDC 1.2 - A SIMULINK Environment for Flight Dynamics and Control Analysis*. Zeist, The Netherlands, February 1998. `http://www.dutchroll.com`.

[60] Jacob Reiner, Gary J. Balas, and William L. Garrard. Robust dynamic inversion for control of highly maneuverable aircraft. *Journal of Guidance, Control, and Dynamics*, 18(1):18–24, January–February 1995.

[61] Jacob Reiner, Gary J. Balas, and William L. Garrard. Flight control design using robust dynamic inversion and time-scale separation. *Automatica*, 32(11): 1493–1504, 1996.

[62] Wilson J. Rugh. *Linear System Theory*. Prentice Hall, second edition, 1996.

[63] Saab–BAE Gripen AB. JAS 39 Gripen homepage. `http://www.gripen.se`.

[64] Saab Aircraft AB. Generic Aerodata Model (GAM). `http://www.flyg.kth.se/edu-dir/downloads-dir`.

[65] R. Sepulchre, M. Janković, and P. V. Kokotović. *Constructive Nonlinear Control*. Springer, 1997.

[66] R. Sepulchre, M. Janković, and P. V. Kokotović. Interlaced systems and recursive designs for global stabilization. In *Proceedings of the 1997 European Control Conference*, 1997.

[67] Rodolphe Sepulchre, Mrdjan Jankovic, and Petar V. Kokotovic. Integrator forwarding: A new recursive nonlinear robust design. *Automatica*, 33(5):979–984, May 1997.

[68] Sahjendra N. Singh, Phil Chandler, Corey Schumacher, Siva Banda, and Meir Pachter. Nonlinear adaptive close formation control of unmanned aerial vehicles. *Dynamics and Control*, 10(2):179–194, 2000.

[69] Sahjendra N. Singh and Marc Steinberg. Adaptive control of feedback linearizable nonlinear systems with application to flight control. *Journal of Guidance, Control, and Dynamics*, 19(4):871–877, July–August 1996.

[70] Jean-Jaques E. Slotine and Weiping Li. *Applied Nonlinear Contol*. Prentice Hall, 1991.

[71] S. Anthony Snell, Dale F. Enns, and William L. Garrard Jr. Nonlinear inversion flight control for a supermaneuverable aircraft. *Journal of Guidance, Control, and Dynamics*, 15(4):976–984, July–August 1992.

[72] Eduardo D. Sontag. A 'universal' construction of Artstein's theorem on nonlinear stabilization. *Systems & Control Letters*, 13:117–123, 1989.

[73] Marc L. Steinberg and Anthony B. Page. Nonlinear adaptive flight control with genetic algorithm design optimization. *International Journal of Robust and Nonlinear Control*, 9(14):1097–1115, 1999.

[74] Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation*. John Wiley & Sons, 1992.

[75] Jann Peter Strand, Kenan Ezal, Thor I. Fossen, and Petar V. Kokotović. Nonlinear control of ships: A locally optimal design. In *Proceedings of the IFAC NOLCOS'98*, pages 732–737, July 1998.

[76] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice Hall, second edition, 1993.

[77] K. H. Well, B. Faber, and E. Berger. Optimization of tactical aircraft maneuvers utilizing high angles of attack. *Journal of Guidance, Control, and Dynamics*, 5(2):131–137, March–April 1982.

[78] Bing-Yu Zhang and Blaise Morton. Robustness analysis of dynamic inversion control laws applied to nonlinear aircraft pitch-axis models. *Nonlinear Analysis, Theory, Methods & Applications*, 32(4):501–532, 1998.

[79] Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control*. Prentice-Hall, 1996.