SWEDISH DEFENCE
RESEARCH AGENCY

FOI-R--1586--SE
January 2005
1650-1942

Scientific report

John W.C. Robinson
Ulrik Nilsson

Trimming & Nonlinear Autopilot for the GSACM Model

Systems Technology Division
SE-172 90 STOCKHOLM
Sweden

Swedish Defence Research Agency
Systems Technology Division
SE-172 90 STOCKHOLM
Sweden

FOI-R--1586--SE
January 2005
1650-1942

Scientific report

John W.C. Robinson
Ulrik Nilsson

Trimming & Nonlinear Autopilot for the GSACM Model

Issuing organization	Report number, ISRN	Report type
Swedish Defence Research Agency Systems Technology Division SE-172 90 STOCKHOLM Sweden	FOI-R--1586--SE	Scientific report
	Research area code Strike and Protection	
	Month year January 2005	Project no. E6956
	Subcategory Weapons and Protection	
	Subcategory 2	
Author/s (editor/s) John W.C. Robinson Ulrik Nilsson	Project manager Mats Fredriksson	
	Approved by Monica Dahlén	
	Sponsoring agency Swedish Defence Material Administration	
	Scientifically and technically responsible Markus Högberg	

Report title

Trimming & Nonlinear Autopilot for the GSACM Model

Abstract

This report describes design and implementation of a trimming routine and a nonlinear controller for simultaneous attitude and velocity control of a small single engine fighter aircraft. The controller is very general and should be applicable to other thrust and moment controlled vehicles (rigid bodies) moving in a gas or fluid as well, such as various forms of unmanned aircraft and underwater vehicles. It is based on multi-input backstepping theory and utilizes a combination of two motions to archive the control. One is a geodesic movement on the sphere of unit norm quaternions for oriention control and the other is a rotation of the body velocity vector into the right direction combined with thrust control of the engine to set its magnitude. The controller is capable of controlling the orientation and velocity to trimmed values for straight path flight or constant angular velocity turn for large region of deviations from these values. A proof of the stabilizing properties of the controller is given. The aircraft, trim routine and controller are implemented in the Modelica language and simulated in the Dymola environment. The model for the aircraft is referred to as the GSACM model and is based on the Admire model. Simulations are presented to illustrate the performance of the controller under various flight conditions.

Keywords

Control Theory, Modelica, Dymola, Backstepping, Nonlinear, Quaternions

Further bibliographic information	Language English
ISSN 1650-1942	Pages 43
Distribution By sendlist	Price Acc. to pricelist Security classification Unclassified

Utgivare	Rapportnummer, ISRN	Klassificering
Totalförsvarets forskningsinstitut Avdelningen för Systemteknik SE-172 90 STOCKHOLM Sweden	FOI-R--1586--SE	Vetenskaplig rapport
	Forskningsområde: Bekämpning och skydd	
	Månad, år: Januari 2005	Projektnummer: E6956
	Delområde: VVS med styrda vapen	
	Delområde 2	
Författare / redaktör	Projektledare: Mats Fredriksson	
John W.C. Robinson Ulrik Nilsson	Godkänd av: Monica Dahlén	
	Uppdragsgivare / kundbeteckning: FMV	
	Tekniskt och / eller vetenskapligt ansvarig: Markus Högberg	

Rapportens titel

Trimning och ickelinjär autopilot för GSACM modellen

Sammanfattning

Denna rapport beskriver utformning och implementering av en trimningsrutin och ickelinjär regulator för samtidig reglering av attityd (orientering) och hastighet för ett litet enkelmotorigt stridsflygplan. Regulatorn är dock generell och torde vara användbar för andra dragkrafts- och momentstyrda farkoster som rör sig i en gas eller fluid, såsom olika former av obemannade flyg och undervattensfarkoster. Den baseras på multi-input backsteppingteori och använder en kombination av två rörelser för att åstakomma regleringen. Den ena är en geodetisk rörelse på sfären av kvaternioner med enhetsnorm för reglering av orienteringen och den andra är en rotation av kroppen så att hastighetsvektorn får rätt riktning i kombination med dragkraftsstyrning så att den får rätt belopp. Regulatorn är kapabel att reglera orienteringen och hastigheten till trimmade värden för rak flygbana eller för sväng med konstant vinkelhastighet, för ett stort område av avvikelser från dessa värden. Ett bevis för regulatorns stabiliserande egenskaper ges. Flygplanet, trimrutinen och regulatorn är implementerade i språket Modelica och simulerade i Dymolamiljö. Flygplansmodellen benämns GSACM och baseras på Admiremodellen. Simuleringar visas som illustrerar prestanda för regulatorn under varierande flygförhållanden.

Nyckelord

Reglerteori, Modelica, Dymola, Backstepping, Ickelinjär, Kvaternioner

Övriga bibliografiska uppgifter	Språk
	Engelska
ISSN: 1650-1942	Antal sidor: 43
Distribution: Enligt missiv	Pris: Enligt prislista Sekretess: Öppen

Contents

1. Introduction

The present document describes two computer code modules that implement the functions of trimming and nonlinear attitude-velocity autopilot for a version of the Admire aircraft model. These code modules were developed as a part of an evaluation study of the Modelica programming language at FOI department of autonomous systems.

The Admire model represents a small generic single engine fighter aircraft with delta-canard configuration and was originally implemented in Simulink. As of this writing, the latest version of this model is v4.0 and it is freely available on the web via `http://www.foi.se/admire`. Recently, the Admire model has been ported (as FOI sponsored contract work) to the Modelica language, which is a modern object oriented language for modeling and simulation targeted mainly on dynamical systems described by differential-algebraic equations (DAE). [1] The Modelica port [2] of the Admire model is in this report referred to as the GSACM model (which, e.g., could stand for Generic Small Aircraft Model).

After the initial port to Modelica, the GSACM model did not contain any provisions for trimming nor any controller. It was therefore decided that the evaluation study of Modelica should include adding such functionality. The first part of the evaluation project consisted in porting the flight control system (FCS) from the Simulink version (v3.4h) of the Admire model. This FCS is based on classical design techniques and gives the aircraft desireable handling qualities, as seen from a pilot's view. However, two central research topics at the department are nonlinear control and unmanned aerial vehicles (UAVs) and it was therefore later decided that a suitable initial design task for the GSACM model was to develop a nonlinear autopilot for attitude and velocity control. This task furthermore required the development of a suitable standalone trimming routine, so that the controller could be tested on flight paths containing segments of ascent, descent, and smooth turns, as well as abrupt turns changes between these types of segments.

1.1 Outline

In the following chapter we describe the structure of the GSACM model and its implementation in Modelica, and in the next chapter we describe the underlying mathematical model. In Chapter 4 we describe the trim routine. After this follows the chapter where we derive the controller and give a proof of its stability. Finally, in Chapter 6 we present some simulations illustrating the performance of the controller when applied to the GSACM model.

1.2 Notation

Vectors are generally marked using bold face, e.g $\mathbf{V}$, and scalars using ordinary font. Hence, the components of a vector $\mathbf{V} \in \mathbb{R}^3$ are V_1,V_2,V_3. Transposition of a vector $\mathbf{V}$ is marked as $\mathbf{V}^T$.

[1] A starting point for information about Modelica is `http://www.modelica.org`.

[2] The GSACM model is in some aspects slightly simplified, for example the mass distribution is constant.

2. The GSACM model

2.1 The Modelica language

Modelica is an object oriented language for modeling and simulation of dynamical systems. [1] It focuses on systems described by differential algebraic equations (DAEs), but can handle discrete (in time and state) and hybrid systems as well. A key feature of the language is that it permits *acausal* modeling, i.e. models can be expressed in their natural equation form without the need to explicitly structure the calculations sequentially in the form of assignment statements. Moreover, it permits *splitting* of equations across the boundaries of objects so that an equation is completed only when two objects are joined together using a special object class called *connector*. Another feature of the language is that it permits documentation (in the form of HTML code) and graphical information (as vector graphics) to be *integrated* directly into the code. This makes documentation and visualization of the model very straightforward. The object oriented nature of the language inspires modular design and reuse of classes. Consequently, a number of *standard classes* for Modelica have been developed, aimed at modeling tasks in electrical, mechanical and many other engineering disciplines. (An explicit design objective of Modelica has been to facilitate *multi domain modeling*.) The GSACM model employs all of these features of the Modelica language (cf. e.g. Sec. 5.4).

Currently, there are only a few translators for Modelica available and the two most important are Dymola of Dynasim AB and MathModelica of MathCore Engineering AB. They both contain a graphical editor for creating Modelica code (e.g. by drag-and-drop of components from the libraries of standard classes) and they have the Dymola kernel as its central component. MathModelica is implemented as a notebook for Mathematica which makes it easy to do supporting symbolic calculations and use the Modelica model from within Mathematica. Dymola, on the other hand, is a standalone application and it has facilities for producing 3D animations of the system simulated (using components from one of the standard “additional” libraries of classes). Both Dymola and Mathematica are available on the Microsoft Windows as well as GNU/Linux (x86) platforms. For the present work, the Dymola environment on GNU/Linux was used.

2.2 Structure of the model

The GSACM model, see Fig. 2.1, is divided into a number of submodels, or objects, such as `Body`, `Aerodynamics`, `Atmosphere`, `Earth`, `Reference point` and `Engine`. The `Body` object represents the dynamics of the aircraft, the `Atmosphere` object provides the Mach number and other environmental parameters and the `Earth` object provides the value of the gravitation constant at the altitude in question, etc. The `Reference point` object is simply a “summing point” for all the quantities that are calculated as summed contribitions from the various submodels, like *total external force, total external torque* (around the center of gravity (CoG)), *total mass, total moment of inertia* (around CoG) and so on. Summing operations of this type are

[1] A comprehensive overview of the Modelica language and is given in [1]. For a shorter (and perhaps more easily accessible) introduction developed around a number of examples see [2].

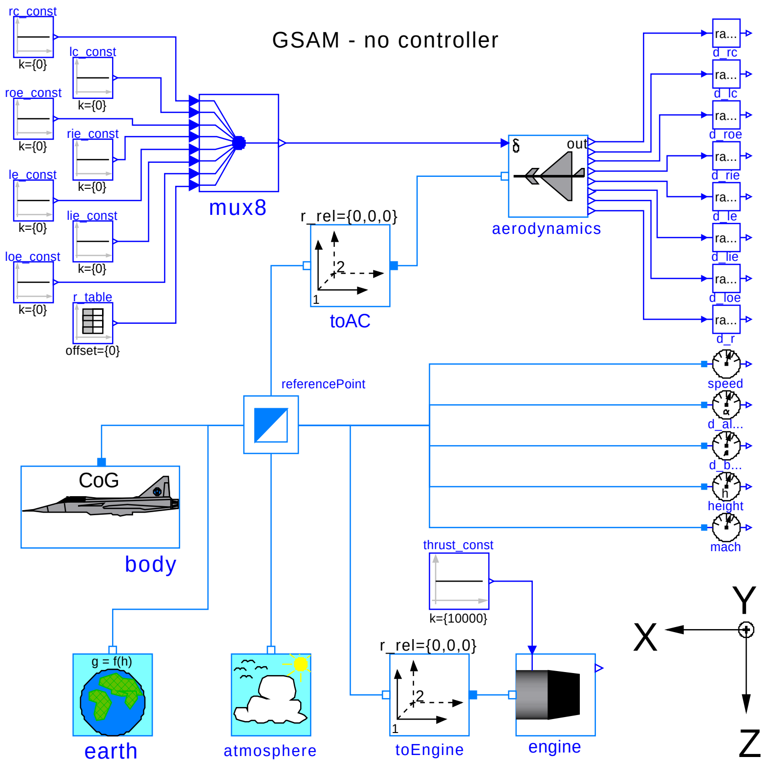


Figure 2.1: Graphical representation of the GSACM model rendered from the graphical annotations in the Modelica code. In this setup, only the “bare” aircraft is modeled, no controller is present. (Control surface deflections are entered as constants in the upper left part of the display, the engine thrust is entered as a constant in the lower right corner and the other parameters, such as initial values, are embedded into the code representing each object.) The (blue) lines connecting objects represent *connect* statements in the Modelica code that connect the `connector` objects. Together they represent equalities in equations, which may or may not be “directed,” i.e. be in the form of assignments, depending on the type of the `connector`.

conveniently implemented in Modelica using so called *flow variables*, which is another feature of the language. A flow variable is a quantity that sums to zero at a connection point and occurs frequently in physical modeling (not only in the mechanical context here but also, of course, in electrical and hydraulical engineering).

In the setup depicted in Fig. 2.1 there is no controller present; the aircraft is free-flying with fixed engine thrust and control surface deflections. The first thing to accomplish when modeling an aircraft is usually to achieve sustained flight. Unless the control surface deflections and engine thrust (as well as initial values of the state in the dynamics) entered in the model correspond to an equilibrium for the differential equations representing the dynamics, this condition will not be met. [2] Therefore, the first thing to add to the setup in Fig. 2.1 to make it more useful is a trimming routine to find such equilibria. In the next chapter we describe the equations of motion in the GSACM model (in the form depicted in Fig. 2.1) before we turn to the details of the trimming routine and autopilot in the following chapters.

[2] If the equilibrium in question is unstable, the time for sustained flight will be very short, but finding such equilibria is still in general the first task to accomplish.

3. Equations of motion

The equations describing the motion of the aircraft are contained in the `Body` object and consist of kinematical as well as dynamical relations expressed in two different (cartesian) coordinate systems. The first, denoted F_e, is *earth fixed* and is considered as an inertial system, and the other, denoted f_b, is *body fixed* with its origin in CoG of the aircraft. [1] All of what is presented here can be found in any standard textbook on rigid body mechanics or aircraft mechanics, such as [3].

3.1 Kinematics

3.1.1 Rotational motion. If f_b is translated to that its origin coincides with that of F_e, the orientation of f_b expressed in terms of the coordinates of F_e can be represented by a rotation matrix $\mathbf{R}$ such that

$$\mathbf{U} = \mathbf{R}\mathbf{u},$$

where $\mathbf{U}$ is an arbitrary vector expressed in F_e and $\mathbf{u}$ is the corresponding vector in f_b. If the system f_b rotates smoothly in F_e (with its origin fixed) with the angular velocity vector $\mathbf{\Omega}$ (expressed in F_e), i.e. if $\mathbf{R}$ varies smoothly, then the angular velocity $\boldsymbol{\omega}$ of this rotation expressed in f_b is given by

$$\boldsymbol{\omega} = \mathbf{R}^{-1}\mathbf{\Omega}. \tag{3.1}$$

Alternatively, the rotation represented by the matrix $\mathbf{R}$ can be expressed by a unit norm quaternion $\mathbf{Q}$. For the time derivative $\dot{\mathbf{Q}}$ of this quaternion we have

$$\dot{\mathbf{Q}} = \frac{1}{2}\tilde{\mathbf{\Omega}} \circ \mathbf{Q}, \tag{3.2}$$

where $\tilde{\mathbf{\Omega}} = (0, \mathbf{\Omega})$ is the pure quaternion (real part zero) formed from the three-vector $\mathbf{\Omega}$ and $\circ$ denotes the quaternion product. By using (3.1) and writing out and rearranging the components on the right hand side of (3.2) this relation can be put on the form

$$\dot{\mathbf{Q}} = \frac{1}{2}\mathbf{Q} \circ \tilde{\boldsymbol{\omega}} = \frac{1}{2}\mathbf{A}(\boldsymbol{\omega})\mathbf{Q}, \tag{3.3}$$

where $\tilde{\boldsymbol{\omega}} = (0, \boldsymbol{\omega})$ is the pure quaternion formed from $\boldsymbol{\omega} = [\omega_1, \omega_2, \omega_3]^T$, the matrix $\mathbf{A}(\boldsymbol{\omega})$ is given by

$$\mathbf{A}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix} \tag{3.4}$$

and the product on the right hand side of (3.3) is ordinary matrix vector product (i.e. the quaternion $\mathbf{Q}$ is considered merely as an ordinary four-vector there). For future use we note that the matrix function $\mathbf{A}$ is linear in its argument and skew symmetric.

[1] The CoG is considered fixed in body coordinates in this version of GSACM.

3.1.2 Translational motion. Turning to translational motion, we have

$$\mathbf{V} = \mathbf{R}\mathbf{v}, \tag{3.5}$$

where $\mathbf{V}$ and $\mathbf{v}$ are the velocity of the aircraft CoG expressed in F_e and f_b, respectively. For the acceleration we obtain from the theory of relative motion

$$\dot{\mathbf{V}} = \frac{d}{dt}\big(\mathbf{R}\mathbf{v}\big) = \mathbf{R}\dot{\mathbf{v}} + \boldsymbol{\Omega} \times \mathbf{R}\mathbf{v} = \mathbf{R}(\dot{\mathbf{v}} + \boldsymbol{\omega} \times \mathbf{v}). \tag{3.6}$$

3.2 Dynamics

3.2.1 Force equation. The linear motion for the CoG of the body is given by Newton's second law in the inertial frame F_e as

$$m\dot{\mathbf{V}} = \mathbf{F}, \tag{3.7}$$

where $\mathbf{F}$ is the sum total of all the external forces acting on the body and m is its mass. [2] The total force $\mathbf{f}$ expressed in f_b is given by

$$\mathbf{f} = \mathbf{R}^{-1}\mathbf{F},$$

and if we combine this with (3.6) we can express the force equation (3.7) in the body coordinate system f_b as

$$m(\dot{\mathbf{v}} + \boldsymbol{\omega} \times \mathbf{v}) = \mathbf{f}^{(a)} + \mathbf{t} + m\mathbf{g}, \tag{3.8}$$

where $\mathbf{f}^{(a)}$ and $\mathbf{t}$ are, respectively, the sum of all the aerodynamical and thrust forces acting on the aircraft expressed in f_b, and $\mathbf{g}$ is the vector of gravitational constants in f_b (i.e. $\mathbf{g} = \mathbf{R}^{-1}\mathbf{G}$ where $\mathbf{G} = [0, 0, g]^T$ is the gravitation vector in F_e and g is the gravitational constant). In this version of the GSACM model the engine is assumed to be mounted on the x-axis in f_b, the aircraft body frame, so the thrust vector $\mathbf{t}$ is of the form $\mathbf{t} = [\tau, 0, 0]^T$, for some τ.

3.2.2 Moment equation. The rotational motion around the origin in F_e is given by Euler's equation. If we denote the moment of inertia tensor around CoG in F_e by $\mathbf{J}$ we thus have

$$\mathbf{J}\dot{\boldsymbol{\Omega}} = \mathbf{M}, \tag{3.9}$$

where $\mathbf{M}$ is the total torque around CoG in F_e. If we further introduce the moment of inertia tensor $\mathbf{j}$ around the origin in f_b as

$$\mathbf{j} = \mathbf{R}^{-1}\mathbf{J}\mathbf{R}$$

we see that we can rewrite (3.9) using the theory of relative motion as

$$\mathbf{M} = \frac{d}{dt}\big(\mathbf{R}\mathbf{j}\mathbf{R}^{-1}\boldsymbol{\Omega}\big) = \frac{d}{dt}\big(\mathbf{R}\mathbf{j}\boldsymbol{\omega}\big) = \mathbf{R}\mathbf{j}\dot{\boldsymbol{\omega}} + \boldsymbol{\Omega} \times \mathbf{R}\mathbf{j}\boldsymbol{\omega} = \mathbf{R}(\mathbf{j}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}). \tag{3.10}$$

Since the torque $\mathbf{m}$ around the origin in f_b is given by

$$\mathbf{m} = \mathbf{R}\mathbf{M}$$

we see that (3.10) gives us the moment equation around the origin in f_b as

$$\mathbf{m} = \mathbf{j}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}, \tag{3.11}$$

where $\mathbf{m}$ is the sum of all aerodynamical and thrust torques [3] in f_b around the origin (i.e. the CoG in F_e).

[2] It is assumed in this version of GSACM that the mass distribution, and hence the total mass, of the aircraft is constant.

[3] Since the engine is assumed to be mounted on the aircraft x-axis, and produces thrust along the same axis, it doesn't give any contribution to the torque.

3.3 Total GSACM motion

Summing up, the equations of motion in the `Body` object of the GSACM model are the four equations (3.5),(3.3), (3.8), and (3.11) which we collect here for easy reference

$$\mathbf{V} = \mathbf{R}\mathbf{v}, \tag{3.12}$$

$$\dot{\mathbf{Q}} = \frac{1}{2}\mathbf{A}(\boldsymbol{\omega})\mathbf{Q}, \tag{3.13}$$

$$\mathbf{f}^{(a)} + \mathbf{t} = m(\dot{\mathbf{v}} + \boldsymbol{\omega} \times \mathbf{v} - \mathbf{g}), \tag{3.14}$$

$$\mathbf{m} = \mathbf{j}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}. \tag{3.15}$$

To get the complete *location–orientation* description of the aircraft one has to solve (3.13)–(3.15) and also integrate (3.12) to get the location $\mathbf{O}$ of the origin of f_b expressed in F_e. We note in passing that there are no inherent theoretical or numerical difficulties with the solution of (3.13)–(3.15) since for bounded $\mathbf{f}^{(a)}, \mathbf{t}, \mathbf{m}$ the system is uniformly Lipschitz continuous in $\mathbf{Q}, \mathbf{v}, \boldsymbol{\omega}$.

3.4 Aerodynamics

The aerodynamic forces and moments are defined in the body fixed coordinate system f_b as illustrated in Fig. 3.1. The x-axis is the *roll axis,* the y-axis is the *pitch axis*

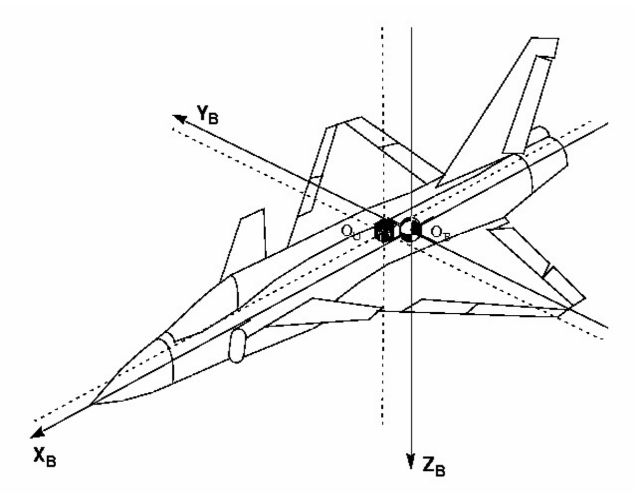


Figure 3.1: The definitions of the the coordinate axes in the body fixed coordinate system f_b used in the GSACM model. (In this figure a second body fixed coordinate system is indicated by dashed lines, the *aerodynamic frame,* but in the version of the GSACM model discussed in this report it coincides with the CoG centered frame f_b.)

and the z-axis is the *yaw axis.* The aerodata used in the GSACM model is the Generic Aerodata Model (GAM) data from SAAB that is also part of Admire and is given in parametric form in terms of coefficients $C_{(\,\cdot\,)}$ for force and moment in the standard fashion. Thus, if $\mathbf{f}^{(a)} = [f_1^{(a)}, f_2^{(a)}, f_3^{(a)}]^T$ and $\mathbf{m} = [m_1, m_2, m_3]^T$ we have

$$f_1^{(a)} = qS_{ref}C_x, \quad (3.16)$$
$$f_2^{(a)} = qS_{ref}C_y, \quad (3.17)$$
$$f_3^{(a)} = qS_{ref}C_z, \quad (3.18)$$
$$m_1 = qS_{ref}b_{ref}C_l, \quad (3.19)$$
$$m_2 = qS_{ref}c_{ref}C_m, \quad (3.20)$$
$$m_3 = qS_{ref}b_{ref}C_n, \quad (3.21)$$

where the dynamic pressure q is given by

$$q = \frac{1}{2}\rho V^2 \quad (3.22)$$

and ρ is the air density, and $V = \|\mathbf{v}\| = \|\mathbf{V}\|$ is the absolute air speed or *aircraft velocity*. The constants S_{ref},b_{ref} and c_{ref} are reference numbers with dimension, chosen such that the coefficients $C_{(\cdot)}$ are dimensionless. [4] The coefficients $C_{(\cdot)}$ have different parametric dependence on the state and environmental parameters (see Chap. 4). [5] All the coefficients $C_{(\cdot)}$ also depend on the *control surface deflections* δ_{lc},δ_{rc},δ_{loe},δ_{roe}, δ_{lie},δ_{rie},$\delta_{\ell e}$,δ_r where δ_{lc},δ_{rc} are the left and right canards, δ_{loe},δ_{roe} are the left and right outer elevons, δ_{lie},δ_{rie} are the left and right inner elevons, δ_{le} is the leading edge flaps and δ_r, finally, is the tail rudder. The control surface configuration is illustrated in Fig. 3.2.

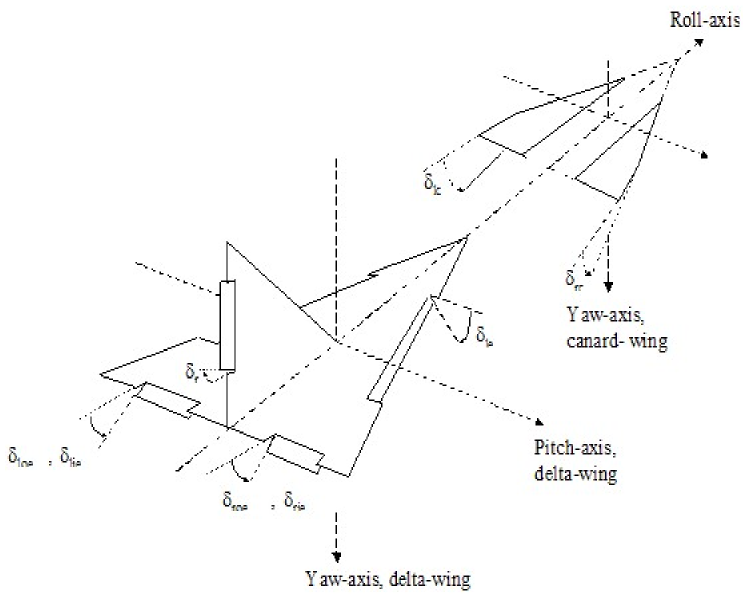


Figure 3.2: Control surface configuration on the GAM aerodata model employed by the GSACM aircraft model.

3.5 Engine

The engine is modeled as a simple thrust source with no parameteric dependence on the environment but with a simple first order dynamics representing the engine

[4] All the details about the aerodynamic data can be found in the GAM documentation that accompanies the Admire model.

[5] There is also a (weak) dependence on the state derivatives.

inertia;

$$\dot{\tau} = b(\tau - u_\tau), \tag{3.23}$$

where τ is the value of the engine thrust (directed along the x-axis in f_b) and u_τ is the control input (desired thrust) to the engine. The time constant b is set to 0.5.

We now turn to the problem of trimming the GSACM model, i.e. finding equilibria for the differential equations (3.13)–(3.15).

4. Trimming

The trimming routine computes equilibrium states in the two aircraft dynamical equations (3.14), (3.15) and the engine thrust equation (3.23) for trimmed wings level flight or for steady turns, for given values of altitude h, Mach number M, flight path angle γ and (total) angular velocity ω. Here, the flight path angle γ is defined in terms of the aircraft velocity $\mathbf{V} = [V_1, V_2, V_3]^T$ in F_e as $\gamma = \arcsin(V_3/V_{xy})$, where V_{xy} is the projection of $\mathbf{V}$ on the xy-plane in F_e, and the angular velocity ω is simply defined as $\omega = \|\mathbf{\Omega}\| = \|\boldsymbol{\omega}\|$. Since the GAM model has redundant control surfaces, i.e. two elevons on each wing and canards, the control commands for a specific set of trimmed states are not unique. For this reason both elevons on each wing are always set to have the same deflection, and the canard deflection can be specified with an additional input, preferably zero or the angle of attack α with switched sign. This effectively reduces the aircraft to one with a simple control surface configuration in the form of aileron (roll), elevator (pitch) and rudder (yaw), with deflections given by the vector $\bar{\delta} = [\delta_a, \delta_e, \delta_r]^T$.

The course of action is to first obtain force equilibrium in body (i.e. f_b) z-direction, assuming that only the angle of attack α has influence on the aerodynamic force $\mathbf{f}^{(a)}$. It is also assumed that the resulting aerodynamic force $f_1^{(a)}$ in body x-direction can be balanced by thrust τ. Forces in body y-direction is not wanted, thus the force balance here is trivial.

Next, the pitch moment $m_{pitch} = m_2$ is considered. The elevator deflection yielding zero pitch moment is computed. Finally, the aileron and rudder deflections are determined for yaw and roll moment balance, i.e. balance for $m_{yaw} = m_3$ and $m_{roll} = m_1$.

The trimming routine provides trimmed values of angle of attack $\alpha^{(0)}$, elevator, aileron and rudder control surface deflections $\bar{\delta}^{(0)} = [\delta_a^{(0)}, \delta_e^{(0)}, \delta_r^{(0)}]^T$, desired thrust $\tau^{(0)}$, body frame velocity $\mathbf{v}^{(0)}$ and angular velocity vector $\boldsymbol{\omega}^{(0)}$, load factor $n_z^{(0)}$ and roll angle $\varphi^{(0)}$.

4.1 Trim Algorithm

For a given Mach number M and altitude h, compute the aircraft velocity $V = \|\mathbf{V}\| = \|\mathbf{v}\|$ using standard atmosphere properties

$$V = V_s(h)M$$

where $V_s(h)$ is the altitude dependent sonic speed. When a non-zero angular velocity ω is provided, the required centripetal acceleration a_{cen} is the the cross product of the angular velocity vector and the velocity vector, or equivalently, with the available scalars

$$a_{cen} = \omega V \cos(\gamma),$$

where γ is the flight path angle. For a trimmed turn we desire no forces in body y-direction, only in the xz-plane of the aircraft, see Figure 4.1. With this the roll angle φ is determined by

$$\varphi = \arctan\left(\frac{a_{cen}}{g}\right),$$

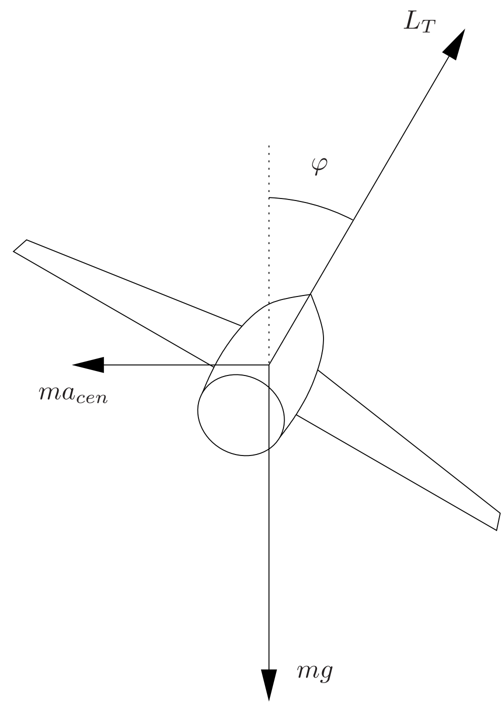


Figure 4.1: Bank angle for no acceleration in body y-direction.

where g is the acceleration of gravity, and further the load factor n_z is given by

$$n_z = \frac{\cos(\gamma)}{\cos(\varphi)}.$$

In Figure 4.1, $L_T = m\sqrt{a_{cen}^2 + g^2}$ is composed of aerodynamic forces and thrust, and is to balance the weight and the centrifugal force.

4.1.1 Force Equilibrium. Obtaining force equilibrium amounts to finding equilibrium points in the state equation (3.14) for body velocity using the relations (3.16)–(3.18) and (3.23). The force C_x, C_y, C_z and moment coefficients C_l, C_m, C_n are functions of a variety of variables, such as Mach number M, angle of attack α, sideslip angle β, deflections of the control surfaces δ_{lc},δ_{rc},δ_{loe},δ_{roe}, δ_{lie},δ_{rie},$\delta_{\ell e}$,δ_r, angular velocity vector $\boldsymbol{\omega}$, etc. For simplicity it is assumed that the force coefficients is mostly affected by the angle of attack α, defined as

$$\alpha = \arccos\left(\frac{v_1}{V}\right) \tag{4.1}$$

(the angle of side slip is set to zero), and that the control surface deflections $\bar{\delta} = [\delta_a, \delta_e, \delta_r]^T$ (aileron, elevator and rudder, respectively) have small influence. Figure 4.2 shows the forces acting on the aircraft. The dotted line in the Figure represent the plane perpendicular to the vector of total acceleration, i.e., the sum of the acceleration of gravity and the centrifugal acceleration. (For the wings level case, the plane is the horizontal plane.) With this assumption we can pose the body z force equilibrium equation as

$$q\, S_{ref}\, C_z(M, \alpha, \omega_y, \bar{\delta}) = m\sqrt{a_{cen}^2 + g^2}\cos(\gamma + \alpha), \tag{4.2}$$

where $C_z(M, \alpha, \bar{\delta})$ is the non-dimensional body z aerodynamic force coefficient, q is the dynamic pressure in (3.22), S_{ref} is the wing reference area and m is the mass of

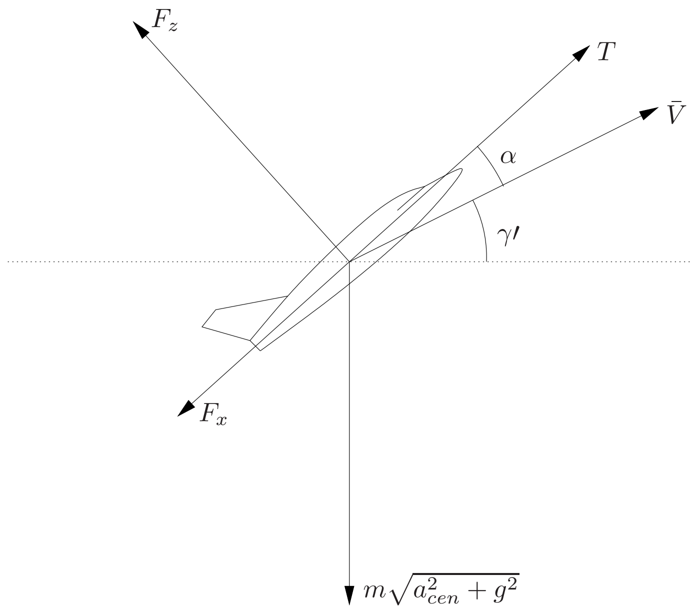


Figure 4.2: Forces acting on the aircraft.

the aircraft. The $C_z(\alpha)$ coefficient is typically obtained from aerodata tables, and thus an explicit solution for α can not be obtained by (4.2). The angle of attack is instead obtained numerically using e.g. the secant method. Knowing the angle of attack α, the body frame equilibrium velocity vector $\mathbf{v}^{(0)}$ is given by

$$\mathbf{v}^{(0)} = [V \cos \alpha, 0, V \sin \alpha]^T, \tag{4.3}$$

and since the thrust force acts along the body x-axis through the center of mass, the required equilibrium thrust $\tau^{(0)}$ is

$$\tau^{(0)} = qS_{ref}C_x(M, \alpha, \bar{\delta}) + m\sqrt{a_{cen}^2 + g^2}\sin(\gamma + \alpha), \tag{4.4}$$

where $C_x(M, \alpha, \bar{\delta})$ is the non-dimensional body x aerodynamic force coefficient.

4.1.2 Moment Equilibrium. The state equation for angular velocity in body frame is the moment equation (3.15) viz.,

$$\mathbf{m} = \mathbf{j}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}, \tag{4.5}$$

where $\mathbf{m} = [m_{roll}, m_{pitch}, m_{yaw}]^T$ is the moment around the origin in f_b produced by the aerodynamics. For trimmed flight the total moment $\mathbf{m}_a$ is zero, and for the trivial case of no turn all terms on the right hand side of (4.5) are zero. However, for a steady state turn, only the first term on the right hand side vanishes. In an arbitrary turn the aircraft does not generally rotate around a principal inertia axis. Hence, the inertia coupling produces a moment that has to be balanced by an opposite aerodynamic moment for a sustained steady turn. In other words, we must have

$$\mathbf{m} = \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}, \tag{4.6}$$

or equivalently, using (3.19)–(3.21),

$$\begin{bmatrix} m_{roll} \\ m_{pitch} \\ m_{yaw} \end{bmatrix} = \begin{bmatrix} qS_{ref}b_{ref}C_l \\ qS_{ref}c_{ref}C_m \\ qS_{ref}b_{ref}C_n \end{bmatrix} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix}, \tag{4.7}$$

where the vector $[m_x, m_y, m_z]^T$ defined as

$$\begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \boldsymbol{\omega} \times \mathbf{j}\boldsymbol{\omega}$$

is introduced for convenience. Computation of the elevator deflection δ_e by e.g. the secant method is straight-forward, assuming that this control surface only affects the pitch moment

$$C_m(M, \alpha, \omega_y, \delta_e) q S_{ref} c_{ref} - m_y = 0, \tag{4.8}$$

where c_{ref} is the reference mean wing chord. Obtaining the aileron and rudder deflections are a bit bore complicated since both of those affect both the roll and yaw moments. One would like to solve the minimization problem on the form

$$\min_{\delta_a, \delta_r} \quad (m_{roll}(M, \alpha, \omega_x, \omega_z, \delta_a, \delta_r) - m_x)^2 + (m_{yaw}(M, \alpha, \omega_x, \omega_z, \delta_a, \delta_r) - m_z)^2. \tag{4.9}$$

In the absence of optimization routines (4.9) is solved by alternating between two sub problems. Since the aileron has a larger influence on the roll moment than on the yaw moment (and vice versa for the rudder), δ_r is held fixed and δ_a is computed from

$$C_l(M, \alpha, \omega_x, \omega_z, \delta_a, \delta_r) q S_{ref} b_{ref} - m_x = 0, \tag{4.10}$$

where b_{ref} is the reference wing span. Now δ_a is held fixed and δ_r is solved out of

$$C_n(M, \alpha, \omega_x, \omega_z, \delta_a, \delta_r) q S_{ref} b_{ref} - m_z = 0. \tag{4.11}$$

When alternating between (4.10) and (4.11), updating δ_a and δ_r, the objective of (4.9) is obtained with satisfactory precision after just a few iterations.

Now we have a trimmed condition that is fairly good. To increase accuracy one can preferably repeat the procedure from the force equilibrium computation, using the control surface deflections obtained from the moment equilibrium condition. Iterating the whole trimming algorithm three or four times result in a very accurate trimmed condition.

5. Autopilot

5.1 The attitude-velocity control problem

The three-dimensional attitude-velocity control problem can be cast as the problem of controlling all three components of the aircraft velocity vector $\mathbf{V}$ in the frame F_e. If we only consider a fixed relation between the components of the vector $\mathbf{V}$, i.e. only consider the *direction* of $\mathbf{V}$, we obtain a natural three-dimensional generalization of the standard two-dimensional definition of flight path angle γ used in Chap. 4. The three dimensional version of the problem considered here thus has one added degree of difficulty, namely the control of the *magnitude* of $\mathbf{V}$. Controlling the attitude and velocity in three dimensions can be considered as the most basic task for an autopilot to accomplish.

If we recall the relation (3.12) the velocity in the frames F_e and f_b respectively as

$$\mathbf{V} = \mathbf{R}\mathbf{v},$$

we see that attitude-velocity control problem can (neglecting wind) be split into two separate problems; (i) the problem of controlling the body components (i.e. in f_b) of the airspeed vector $\mathbf{v}$ and (ii) simultaneously controlling the orientation $\mathbf{R}$ of the aircraft (in F_e). Since $\mathbf{R}$ can be viewed as a member of $SO(3)$, the group of rotations in three-dimensional space $\mathbb{R}^3$, which is a three-dimensional manifold, the attitude-velocity control problem is six-dimensional. The generalized control surface deflections represented by $\bar{\delta} = [\delta_a, \delta_e, \delta_r]^T$ together with the thrust control input u_τ constitute our primary control commands, and these clearly correspond to a four-dimensional manifold. Hence, the control problem is *underactuated.*

5.2 Backstepping

Backstepping is a methodology to systematically design stabilizing controllers for nonlinear systems using a recursive procedure. It requires the system to be of a certain diagonal type but allows the designer a large degree of freedom in the design process. In particular, it allows the designer to keep nonlinear terms that act stabilizing and to focus on destabilizing terms, thus reducing the need for control action. Backstepping is described in many textbooks, see e.g. [4, Sec. 14.3] or the standard reference [5]. Most of what is presented below about backstepping is taken more or less directly from [5].

5.2.1 Lyapunov stability theory. The central component in backstepping is the concept of a Lyapunov function, and particularly control Lyapunov function (CLF). In this section we are going to review some of the basic results concerning such functions that are used in backstepping. The treatment will be brief. For proofs and more detailed developments on the definitions the reader is referred to the references mentioned above.

To begin with we introduce the generic nonlinear system model that we are going to use. The autonomous version of this model is simply

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)), \quad t \in \mathbb{R}, \mathbf{x}(0) = \mathbf{x}_0, \tag{5.1}$$

where the state vector $\mathbf{x}$ belongs to $\mathbb{R}^n$ and f is a Lipschitz continuous function $\mathbb{R}^n \to \mathbb{R}^n$. By standard results for ordinary differential equations we know that for each $\mathbf{x}_0 \in \mathbb{R}^n$ the initial value problem in (5.1) has a unique solution defined over some time interval containing the origin. Moreover, we have the following fundamental result (a proof can be found in [5]).

Theorem 5.2.1. *Let $\mathbf{x} = 0$ be an equilibrium point of (5.1) and let $\mathcal{V} : \mathbb{R}^n \to [0, \infty)$ be a continuously differentiable positive definite and radially unbounded function such that along the solutions $\mathbf{x}(t)$ to (5.1) we have*

$$\frac{d}{dt}\mathcal{V}(\mathbf{x}(t)) = \frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x}(t))\dot{\mathbf{x}}(t) = \frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x}(t))f(\mathbf{x}(t)) \leq -W(\mathbf{x}(t)) \leq 0, \quad \forall t \in \mathbb{R}, \tag{5.2}$$

where $W : \mathbb{R}^n \to \mathbb{R}$ is a continuous function. Then, all solutions $\mathbf{x}(t)$ of (5.1) are globally uniformly bounded and satisfy

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

In addition, if W is positive definite, then the equilibrium at $\mathbf{x} = 0$ is globally uniformly asymptotically stable.

Remark 5.2.2. Implicit in this theorem is the assumption of global existence of solutions to (5.1), which is not guaranteed by the local Lipschitz continuity of f. However, the local Lipschitz continuity together with existence of a compact positively invariant set (see below) is sufficient to guarantee global existence and uniqueness for solutions starting in this set, cf. e.g. [4, Thm. 3.3].

The function $\mathcal{V}$ in the theorem is called a *Lyapunov* function. It frequently happens that one cannot guarantee (5.2) to hold for all starting points $\mathbf{x}(0)$ in $\mathbb{R}^n$ but only a subset thereof, and for this case a more general version of Theorem 5.2.1 is needed. We shall present one such generalization, based on the concept of invariant sets, known as *LaSalle's Theorem*. A set $\mathcal{M} \subseteq \mathbb{R}^n$ is called an *invariant set* for the solutions $\mathbf{x}(t)$ to (5.1) if

$$\mathbf{x}(0) \in \mathcal{M} \Rightarrow \forall t \in \mathbb{R} : \mathbf{x}(t) \in \mathcal{M}.$$

In other words, the solution can never cross the border and is confined to either $\mathcal{M}$ or $\mathcal{M}^c$ (the complement of $\mathcal{M}$) at all times. Similarly, a set $\mathcal{M} \subseteq \mathbb{R}^n$ is called a *positively invariant set* for the solutions $\mathbf{x}(t)$ to (5.1) if

$$\mathbf{x}(0) \in \mathcal{M} \Rightarrow \forall t \geq 0 : \mathbf{x}(t) \in \mathcal{M}.$$

In this case the solution can enter $\mathcal{M}$ but never escape (in forward time). A typical invariant set is an equilibrium point or limit cycle for the system (5.1) and a typical positively invariant set is the set

$$\{\mathbf{x}(0) \in \mathbb{R}^n : \mathcal{V}(\mathbf{x}(0)) \leq c_0 \text{ and } \frac{d}{dt}\mathcal{V}(\mathbf{x}(t)) \leq 0,\, t \geq 0\},$$

where c_0 is some constant.

Theorem 5.2.3 (LaSalle). *Let $\mathcal{M}$ be a compact positively invariant set of (5.1) and let $\mathcal{V} : \mathbb{R}^n \to [0, \infty)$ be a continuously differentiable function such that*

$$\frac{d}{dt}\mathcal{V}(\mathbf{x}(t)) \leq 0, \quad \forall \mathbf{x}(0) \in \mathcal{M},\, t \geq 0.$$

Let $\mathcal{M}_0$ be the set

$$\mathcal{M}_0 = \{\mathbf{x}(0) \in \mathcal{M} : \frac{d}{dt}\mathcal{V}(\mathbf{x}(t)) = 0,\, t \geq 0\}$$

and let $\bar{\mathcal{M}}$ be the largest invariant set contained in $\mathcal{M}_0$. The, every solution $\mathbf{x}(t)$ to (5.1) with $\mathbf{x}(0) \in \mathcal{M}$ converges to $\bar{\mathcal{M}}$ as $t \to \infty$.

A proof of Theorem 5.2.3 can be found in [4].

We now turn to controlled systems and we consider systems on the generic affine form

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + g(\mathbf{x}(t))\mathbf{u}(t), \tag{5.3}$$

where f is as in (5.1), $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ is continuous and $\mathbf{u}(t)$ is a vector in $\mathbb{R}^m$ of continuous (control) functions. A smooth positive definite and radially unbounded function $\mathcal{V} : \mathbb{R}^n \to [0,\infty)$ is called a *control Lyapunov function* for the system (5.3) if it holds that

$$\inf_{\mathbf{u}\in\mathbb{R}^m} \frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x})\big(f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}\big) < 0, \quad \forall \mathbf{x} \neq 0.$$

Intuitively, the existence of a CLF for the system (5.3) suggests that by proper choice of control signal $\mathbf{u}$ the system can be stabilized. To see this more clearly, assume that a smooth feedback law $\ell : \mathbb{R}^n \to \mathbb{R}^m$ has been chosen such that

$$\mathbf{u} = \ell(\mathbf{x}), \tag{5.4}$$

which renders the system (5.3) on the form

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) + g(\mathbf{x}(t))\ell(\mathbf{x}(t)). \tag{5.5}$$

The feedback connected system (5.5) is of the form (5.1). Thus, if there exist a Lyapunov function $\mathcal{V}$ for (5.5) and "rate margin function" W as in Theorem 5.2.1 the feedback connected system (5.5) is stable. Indeed, for the important case of a scalar control signal $\mathbf{u}$ there is a standard procedure (known as *Sontag's formula*) by which such an ℓ and W can be constructed (for a large class of f, g), given a CLF. [1]

5.2.2 Integrator backstepping. In backstepping one doesn't start with the concept of CLF and derive a stabilizing control law from it, but instead a stabilizing control law together with a CLF are constructed simultaneously, with a recursive procedure evolving state by state. The main condition imposed on the systems to be controlled is one about *stabilizability*:

Assumption 5.2.4. Consider the system in (5.3) and assume that there exist a smooth feedback law ℓ as in (5.4) and a smooth, positive definite and radially unbounded function $\mathcal{V} : \mathbb{R}^n \to [0,\infty)$ such that

$$\frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x})\big(f(\mathbf{x}) + g(\mathbf{x})\ell(\mathbf{x})\big) \le -W(\mathbf{x}) \le 0, \quad \forall \mathbf{x} \in \mathbb{R}^n, \tag{5.6}$$

for some continuous $W : \mathbb{R}^n \to \mathbb{R}$ which is positive definite.

Under assumption 5.2.4, there exists a standard procedure for recursively constructing a Lyapunov function and feedback control law that renders the feedback connected system stable. [2] We are later going to apply this procedure to a slightly augmented version of the system (5.3), namely

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f(\mathbf{x}(t)) + g(\mathbf{x}(t))\boldsymbol{\xi}(t), & (5.7)\\ \dot{\boldsymbol{\xi}}(t) &= h(\boldsymbol{\xi}(t)) + k\mathbf{u}(t), & (5.8) \end{aligned}$$

where $\boldsymbol{\xi}$ takes values in $\mathbb{R}^m$, the function $h : \mathbb{R}^m \to \mathbb{R}^m$ is continuous and k is an invertible matrix in $\mathbb{R}^{m\times m}$. Since k in (5.8) is nonsingular, and we can choose the

[1] More generally, the existence of a CLF has been shown to be both necessary and sufficient for global asymptotic stabilizability as in Thm. 5.2.1.

[2] Note that Assumption 5.2.4 merely is a statement about *existence* of a Lyapunov function and accompanying stabilizing control law and no uniqueness is implied.

control $\mathbf{u}$ freely, the system in (5.7), (5.8) is equivalent to the apparently simpler system

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f(\mathbf{x}(t)) + g(\mathbf{x}(t))\boldsymbol{\xi}(t), && (5.9) \\ \dot{\boldsymbol{\xi}}(t) &= \mathbf{u}(t). && (5.10) \end{aligned}$$

The system in (5.9), (5.10) differs structurally from (5.5) only by the extra integrator states in (5.10), and for this reason the control design method applied to (5.9), (5.10) is called integrator backstepping. The main result summarizing the method is the following.

Theorem 5.2.5. *Consider the system (5.9), (5.10) and suppose that (5.9) satisfies Assumption 5.2.4 with $\boldsymbol{\xi}$ replaced by the control $\mathbf{u}$ in (5.4). If the function W is positive definite, then*

$$\mathcal{V}_a(\mathbf{x}, \boldsymbol{\xi}) = \mathcal{V}(\mathbf{x}) + \frac{1}{2}\|\boldsymbol{\xi} - \ell(\mathbf{x})\|^2 \tag{5.11}$$

is a CLF for the full system (5.9), (5.10) (i.e. $\boldsymbol{\xi}$ plays the role of a control in (5.11)) and there exists a feedback law $\mathbf{u} = \ell_a(\mathbf{x}, \boldsymbol{\xi})$ that makes the full system (5.9), (5.10) globally asymptotically stable around $\mathbf{x} = 0, \boldsymbol{\xi} = 0$. One such control law is

$$\mathbf{u} = -c(\boldsymbol{\xi} - \ell(\mathbf{x})) + \frac{\partial \ell(\mathbf{x})}{\partial \mathbf{x}}\big(f(\mathbf{x}) + g(\mathbf{x})\boldsymbol{\xi}\big) - \big(\frac{\partial \mathcal{V}}{\partial \mathbf{x}}(\mathbf{x})g(\mathbf{x})\big)^T, \quad c > 0. \tag{5.12}$$

If the function W is only positive semidefinite, then there exists a feedback law ℓ_a such that

$$\frac{d}{dt}\mathcal{V}_a(\mathbf{x}(t), \boldsymbol{\xi}(t)) \leq -W_a(\mathbf{x}(t), \boldsymbol{\xi}(t)) \leq 0, \tag{5.13}$$

where $W_a(\mathbf{x}, \boldsymbol{\xi}) > 0$ whenever $W(\mathbf{x}) > 0$ or $\boldsymbol{\xi} \neq \ell_a(\mathbf{x}, \boldsymbol{\xi})$. This guarantees global boundedness and and convergence of the vector

$$\begin{bmatrix} \mathbf{x}(t) \\ \boldsymbol{\xi}(t) \end{bmatrix}$$

to the largest invariant $\bar{\mathcal{M}}_a$ set in

$$\mathcal{M}_a = \left\{ \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\xi} \end{bmatrix} \in \mathbb{R}^{n+m} : W(\mathbf{x}) = 0, \boldsymbol{\xi} = \ell_a(\mathbf{x}, \boldsymbol{\xi}) \right\}.$$

A proof of Theorem 5.2.5 can be found in [5] for the case $m = 1$ and in [4] for the more general multi-input case treated here. We point out that the control law suggested in the Theorem is often not the best choice in applications since it may mean cancellation of useful nonlinearities. The requirement to make the left hand side of (5.13) negative (or nonpositive) can in general be met in a number of ways, yielding several useful control laws in applications. Furthermore, the two most important design choices available when applying the above theorem is the selection of $\mathcal{V}$ and the accompanying feedback law ℓ. Indeed, once these two components are chosen, the remaining task to stabilize the full system (5.9), (5.10) using $\mathcal{V}_a$ is to select a feedback law such that $\boldsymbol{\xi}$ tracks ℓ well, i.e. such that the error dynamics $\boldsymbol{\xi} - \ell$ are stable.

5.3 Backstepping the GSACM model

We shall now show how to bring the GSACM model in (3.12)–(3.15) on the standard form (5.9),(5.10) used for integrator backstepping.

The complete aircraft kinematics and dynamics are described by the system of equations formed by (3.12)–(3.15) and (3.23). For the attitude-velocity control problem outlined in Sect. 5.1 we only need (3.13)–(3.15) and (3.23). As the (full) *state vector* we therefore take

$$\begin{bmatrix} \mathbf{v} \\ \mathbf{Q} \\ \boldsymbol{\omega} \\ \tau \end{bmatrix},$$

and as *control variables* we take $\mathbf{m}$ in (3.15) and u_τ in (3.23). (It would perhaps be more natural to view the (generalized) control surface deflections $\bar{\delta}^{(0)} = [\delta_a^{(0)}, \delta_e^{(0)}, \delta_r^{(0)}]^T$ as control variables, but we shall tacitly assume that we only consider the set of control deflections for which there is an invertible relation between $\bar{\delta}^{(0)}$ and $\mathbf{m}$ (which, e.g., excludes the stall region). Hence, we may equivalently take the components of $\mathbf{m}$ as control variables.)

We shall also have reason to consider the reduced system described by only the body relations (3.14), (3.15) and (3.23), for which the state vector is

$$\begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \\ \tau \end{bmatrix}. \tag{5.14}$$

(The control variables are then the same as for the full system.)

5.3.1 Equilibrium points.

Let

$$\begin{bmatrix} \mathbf{v}_0 \\ \mathbf{Q}_0 \\ \boldsymbol{\omega}_0 \\ \tau_0 \end{bmatrix} \tag{5.15}$$

be an equilibrium point for the (full) state dynamics in (3.13)–(3.15) and (3.23). A glance at (3.13) reveals that $\boldsymbol{\omega}_0$ must be 0 at such an equilibrium. For the reduced system (3.14), (3.15) and (3.23), however, we may have equilibria where $\boldsymbol{\omega}_0$ is nonzero. Obtaining such equilibria for the reduced system, using procedures analogous to those described in Chap. 4, is what is usually referred to as "trimming" in the literature. In Appendix A it is shown that for constant nonzero $\boldsymbol{\omega}_0$ the derivative $\dot{\mathbf{Q}}$ is constant and nonzero so for an equilibrium point of the reduced state vector (5.14) such that $\boldsymbol{\omega}_0 \neq 0$ the orientation $\mathbf{Q}$ is time varying. For the special case of equilibrium for the reduced state it is therefore convenient to reserve the symbol $\mathbf{Q}_0$ for the time-varying reference value of the orientation that the equation (3.13) produces. Moreover, since we assume that such a time-varying reference point is a solution trajectory to the full system (3.13)–(3.15) and (3.23) we have

$$\dot{\mathbf{Q}}_0 = \frac{1}{2}\mathbf{A}(\boldsymbol{\omega}_0)\mathbf{Q}_0.$$

Now, a change of variables

$$\begin{bmatrix} \mathbf{v} \\ \mathbf{Q} \\ \boldsymbol{\omega} \\ \tau \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{v}+\mathbf{v}_0 \\ \mathbf{Q}+\mathbf{Q}_0 \\ \boldsymbol{\omega}+\boldsymbol{\omega}_0 \\ \tau+\tau_0 \end{bmatrix}$$

brings the system (3.13)–(3.15) and (3.23) onto the form

$$\begin{aligned}
\dot{\mathbf{v}} &= \frac{1}{m}\mathbf{f}^{(a)} + \frac{1}{m}\mathbf{t} + \mathbf{g} + (\mathbf{v}+\mathbf{v}_0)\times(\boldsymbol{\omega}+\boldsymbol{\omega}_0), && (5.16)\\
\dot{\mathbf{Q}} + \dot{\mathbf{Q}}_0 &= \frac{1}{2}\mathbf{A}(\boldsymbol{\omega}+\boldsymbol{\omega}_0)(\mathbf{Q}+\mathbf{Q}_0), && (5.17)\\
\dot{\boldsymbol{\omega}} &= \mathbf{j}^{-1}(\mathbf{m} - (\boldsymbol{\omega}+\boldsymbol{\omega}_0)\times\mathbf{j}(\boldsymbol{\omega}+\boldsymbol{\omega}_0)), && (5.18)\\
\dot{\tau} &= b(\tau+\tau_0-u_\tau), && (5.19)
\end{aligned}$$

where we have written out the derivative $\dot{\mathbf{Q}}_0$ since it may or may not be zero depending on if we study an equilibrium for the full or the reduced state vector. In (5.19) (and henceforth), the variables $\mathbf{v}, \mathbf{Q}, \boldsymbol{\omega}, \tau$ thus represent, in the case we consider an equilibrium for the full state vector, deviations from the equilibrium in (5.15). In the case of an equilibrium for the reduced state vector in (5.14) the variables $\mathbf{v}, \boldsymbol{\omega}, \tau$ represent deviations from equilibrium values and $\mathbf{Q}$ represents deviation from the time-varying reference value $\mathbf{Q}_0$.

Further, the aerodynamic forces $\mathbf{f}^{(a)}$ are mainly dependent on $\mathbf{v}+\mathbf{v}_0$, the aerodynamic moments $\mathbf{m}$ are mainly dependent on $\mathbf{v}+\mathbf{v}_0, \boldsymbol{\omega}+\boldsymbol{\omega}_0$, and the thrust $\mathbf{t}$ acts only along the aircraft x-axis, i.e. the x-axis in the body f_b. We can make this dependence explicit by writing

$$\mathbf{f}^{(a)} = \mathbf{f}^{(a)}(\mathbf{v}+\mathbf{v}_0), \quad \mathbf{m} = \mathbf{m}(\mathbf{v}+\mathbf{v}_0, \boldsymbol{\omega}+\boldsymbol{\omega}_0), \quad \mathbf{t} = (\tau+\tau_0)\mathbf{e}_x, \tag{5.20}$$

where $\mathbf{e}_x$ is a unit vector in the f_b body x-direction. Likewise, the gravity vector $\mathbf{g}$ is only dependent on $\mathbf{Q}+\mathbf{Q}_0$ and therefore we can write

$$\mathbf{g} = \mathbf{g}(\mathbf{Q}+\mathbf{Q}_0). \tag{5.21}$$

If we now define the vector functions $\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0), \tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0)$, and $\tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0)$ by

$$\begin{aligned}
\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) &= \mathbf{f}^{(a)}(\mathbf{v}+\mathbf{v}_0) - \mathbf{f}^{(a)}(\mathbf{v}_0), && (5.22)\\
\tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0) &= \mathbf{m}(\mathbf{v}+\mathbf{v}_0,\boldsymbol{\omega}+\boldsymbol{\omega}_0) - \mathbf{m}(\mathbf{v}_0,\boldsymbol{\omega}_0), && (5.23)\\
\tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) &= \mathbf{g}(\mathbf{Q}+\mathbf{Q}_0) - \mathbf{g}(\mathbf{Q}_0), && (5.24)
\end{aligned}$$

and use (5.20) we can rewrite the dynamics in (5.16)–(5.19) in terms of the dynamics around the equilibrium point (5.15) as

$$\begin{aligned}
\dot{\mathbf{v}} &= \frac{1}{m}\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) + \frac{1}{m}\tau\mathbf{e}_x + \tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) + (\mathbf{v}+\mathbf{v}_0)\times\boldsymbol{\omega} + \mathbf{v}\times\boldsymbol{\omega}_0, && (5.25)\\
\dot{\mathbf{Q}} &= \frac{1}{2}\mathbf{A}(\boldsymbol{\omega})(\mathbf{Q}+\mathbf{Q}_0) + \frac{1}{2}\mathbf{A}(\boldsymbol{\omega}_0)\mathbf{Q}, && (5.26)\\
\dot{\boldsymbol{\omega}} &= \mathbf{j}^{-1}(\tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0) - (\boldsymbol{\omega}+\boldsymbol{\omega}_0)\times\mathbf{j}\boldsymbol{\omega} - \boldsymbol{\omega}\times\mathbf{j}\boldsymbol{\omega}_0), && (5.27)\\
\dot{\tau} &= b(\tau-\tilde{u}_\tau), && (5.28)
\end{aligned}$$

where we have used the fact that many terms cancel or vanish per definition at the equilibrium and also introduced $\tilde{u}_\tau$ as

$$\tilde{u}_\tau = u_\tau - \tau_0.$$

(Note that there is no linearization involved in (5.25)–(5.28), it is still the full nonlinear equations albeit rewritten such that terms that sum to zero are omitted.) On the form (5.25)–(5.28) the state dynamics are really (almost) on the standard form for integrator backstepping, with $\tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0)$ and $\tilde{u}$ as the control variables.

5.3.2 Standard form. Before we can explicitly write the state dynamics (5.25)–(5.28) around the equilibrium (5.15) on standard form we need to introduce two more quantities. The first is the matrix function $\mathbf{B}$ taking values in $\mathbb{R}^{4\times 3}$ obtained via the identity

$$\mathbf{A}(\boldsymbol{\omega})\mathbf{Q} = \mathbf{B}(\mathbf{Q})\boldsymbol{\omega}, \tag{5.29}$$

i.e. by simply rearranging the terms on the matrix-quaternion product on the left hand side. The second quantity needed is the matrix function $\mathbf{C}$ taking values in $\mathbb{R}^{3\times 3}$ representing the cross product, so that e.g.

$$\mathbf{C}(\mathbf{v})\boldsymbol{\omega} = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \mathbf{v} \times \boldsymbol{\omega}. \tag{5.30}$$

By the properties of the cross product the matrix $\mathbf{C}$ is skew-symmetric, i.e. $\mathbf{C}^T = -\mathbf{C}$. Using these quantities we can now rewrite the state equations (5.25)–(5.28) one final time as

$$\begin{aligned}
\dot{\mathbf{v}} &= \frac{1}{m}\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) + \tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) + \mathbf{v}\times\boldsymbol{\omega}_0 + \mathbf{C}(\mathbf{v}+\mathbf{v}_0)\boldsymbol{\omega} + \frac{1}{m}\tau\mathbf{e}_x, && (5.31)\\
\dot{\mathbf{Q}} &= \frac{1}{2}\mathbf{A}(\boldsymbol{\omega}_0)\mathbf{Q} + \frac{1}{2}\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}, && (5.32)\\
\dot{\boldsymbol{\omega}} &= \mathbf{j}^{-1}(\mathbf{j}\boldsymbol{\omega}\times(\boldsymbol{\omega}+\boldsymbol{\omega}_0) + \mathbf{j}\boldsymbol{\omega}_0\times\boldsymbol{\omega}) + \mathbf{j}^{-1}\tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0), && (5.33)\\
\dot{\tau} &= b\tau - b\tilde{u}_\tau. && (5.34)
\end{aligned}$$

The state dynamics in (5.31)–(5.34) are now indeed on the standard form (5.7), (5.8), which can be seen by making the identifications (here $\sim$ means "corresponds to")

$$\mathbf{x} \sim \begin{bmatrix} \mathbf{v} \\ \mathbf{Q} \end{bmatrix}, \quad \boldsymbol{\xi} \sim \begin{bmatrix} \boldsymbol{\omega} \\ \tau \end{bmatrix}, \quad \mathbf{u} \sim \begin{bmatrix} \tilde{\mathbf{m}}(\mathbf{v},\mathbf{v}_0,\boldsymbol{\omega},\boldsymbol{\omega}_0) \\ \tilde{u}_\tau \end{bmatrix} \tag{5.35}$$

and

$$f \sim \begin{bmatrix} \frac{1}{m}\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) + \tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) + \mathbf{v}\times\boldsymbol{\omega}_0 \\ \frac{1}{2}\mathbf{A}(\boldsymbol{\omega}_0)\mathbf{Q} \end{bmatrix}, \tag{5.36}$$

$$g \sim \begin{bmatrix} \mathbf{C}(\mathbf{v}+\mathbf{v}_0) & \frac{1}{m}\mathbf{e}_x \\ \frac{1}{2}\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0) & \mathbf{0}_{4\times 1} \end{bmatrix}, \quad h \sim \begin{bmatrix} \mathbf{j}^{-1}(\mathbf{j}\boldsymbol{\omega}\times(\boldsymbol{\omega}+\boldsymbol{\omega}_0) + \mathbf{j}\boldsymbol{\omega}_0\times\boldsymbol{\omega}) \\ b\tau \end{bmatrix}, \tag{5.37}$$

$$k \sim \begin{bmatrix} \mathbf{j}^{-1} & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times 3} & -b \end{bmatrix}. \tag{5.38}$$

For an equilibrium point (5.15) to the full state dynamics (5.31)–(5.34) the expressions above simplify since we know that $\boldsymbol{\omega}_0$ in this case must be zero, so the terms in f, h above involving $\boldsymbol{\omega}_0$ vanish.

5.3.3 Desired dynamics. The first task when developing a controller for the system on standard backstepping form is to find a suitable Lyapunov function V as in Thm. 5.2.5 and an accompanying feedback law ℓ such that the feedback connected first part of the system, corresponding to (5.7), is stable with suitable dynamics. When determining what is "suitable" dynamics for the system (5.31), (5.32) we must take into account at least two obvious requirements; (i) the need to aerodynamically stabilize the aircraft and (ii) the desire to solve the attitude and velocity control problem outlined in Sec. 5.1. It is intuitively clear that these two requirements can not be dealt with independently since rotating the aircraft body so that the body velocity vector $\mathbf{v}$ takes a desired value does not necessarily mean that the aircraft has a desired orientation $\mathbf{Q}$. (An additional complication is that we are dealing with an underactuated system, as mentioned at the end of Sec. 5.1.) We are going to solve this problem via a two-tier approach where the controller basically acts as a stabilizing controller as long as the error in the velocity vector $\mathbf{v}$ is large and only when this error has become small will the controller engage the attitude-orientation control, using a smooth transition.

5.3.4 A two-tier controller. We are now going to mathematically develop the control strategy outlined in the section above. The result will be a feedback law like ℓ in Thm. 5.2.5 which, when inserted instead of $\boldsymbol{\xi}$ in the system equation (5.7) (i.e. in (5.31), (5.32)), will give the system represented by (5.7) the "desired dynamics." [3] Such a feedback law is also called a "virtual control" because it acts as a control on the upper half of the system represented by (5.7). We are going to develop the control strategy in two parts, just as indicated above, and after we have presented the parts give a proof of stabilizing properties of the total controller in the following section.

Recently, the related problem of controlling the velocity and angular velocity of a rigid body model of an aircraft has been treated by Glad and Härkegård [6]. They assume, however, that the thrust force of the engine is always aligned along the velocity vector and this is clearly not the case in general. We present a solution to the simultaneous attitude and velocity control problem that employs a velocity control similar to that of Glad and Härkegård but extends it to the case of nonaligned thrust and combines it with a quaternion based control for attitude control. The attitude control utilizes spherical linear interpolation (slerp) on the sphere $\mathcal{S}^3 \subseteq \mathbb{R}^4$ to compute a geodesic representing the minimal rotation of the body needed to control the attitude to the desired value. The velocity control employs rotation of the airspeed vector and thrust control to stabilize angle of attack, the sideslip angle and the absolute velocity to trimmed values.

Stabilizing the velocity vector. The problem of aerodynamically stabilizing the aircraft, without regard to its orientation, is not hard once the system has been brought onto the form (5.31)–(5.34). For instance, one can control $\mathbf{m}(\mathbf{v}, \mathbf{v}_0, \boldsymbol{\omega}, \boldsymbol{\omega}_0)$ such that the velocity vector $\mathbf{v} + \mathbf{v}_0$ is rotated into a position aligned with $\mathbf{v}_0$, while simultaneously controlling the thrust setting $\tilde{u}_\tau$ so that the magnitude becomes right.

A simple way of achieving a rotation of the velocity vector $\mathbf{v} + \mathbf{v}_0$ in the right direction is to use a (virtual) control of the form

$$\boldsymbol{\omega}_{\mathbf{v}}^{des}(\mathbf{v}, \mathbf{v}_0) = -\frac{c_{\mathbf{v}}}{\|\mathbf{v}_0\|^2} \mathbf{v} \times \mathbf{v}_0, \tag{5.39}$$

where $c_{\mathbf{v}}$ is some positive constant. To give some motivation at this point for the choice (5.39) of a (virtual) control $\boldsymbol{\omega}_{\mathbf{v}}^{des}(\mathbf{v}, \mathbf{v}_0)$ one can note that

$$\begin{aligned} -(\mathbf{v} + \mathbf{v}_0) \times \frac{1}{\|\mathbf{v}_0\|^2}(\mathbf{v} \times \mathbf{v}_0) &= -\mathbf{v} \times \frac{1}{\|\mathbf{v}_0\|^2}(\mathbf{v} \times \mathbf{v}_0) - \mathbf{v}_0 \times \frac{1}{\|\mathbf{v}_0\|^2}(\mathbf{v} \times \mathbf{v}_0) \\ &= -\mathbf{v} \times \frac{1}{\|\mathbf{v}_0\|^2}(\mathbf{v} \times \mathbf{v}_0) - P_{[\mathbf{v}_0]^\perp}(\mathbf{v}), \end{aligned} \tag{5.40}$$

where the first term on the right is perpendicular to $\mathbf{v}$ and the second is the negative projection of $\mathbf{v}$ onto the subspace of vectors in $\mathbb{R}^3$ that are orthogonal to $\mathbf{v}_0$. The first term is in general much smaller in magnitude than the second and therefore, when inserted instead of $\boldsymbol{\omega}$ in (5.31), the (virtual) control $\boldsymbol{\omega}_{\mathbf{v}}^{des}(\mathbf{v}, \mathbf{v}_0)$ in (5.39) can act to reduce the error $\mathbf{v}$. However, since the main reduction of the velocity error $\mathbf{v}$ is in the component of it that is orthogonal to $\mathbf{v}_0$ there is a need to complement the control with some action also in the direction of $\mathbf{v}_0$.

[3] For this reason, the feeback $\ell(\mathbf{x})$ is often in the literature denoted $\boldsymbol{\xi}^{des}(\mathbf{x})$, and we shall also use this notation here.

The direction of $\mathbf{v}_0$ is normally almost the same as the direction in which the thrust acts (here, in the body x-direction) and therefore it is natural to try to achieve control action in the $\mathbf{v}_0$ direction by (virtual) thrust control. To see how this can be done it is instructive to study in more detail the first component of the system equation (5.31) which, when written out, reads

$$\begin{aligned} \dot{v}_1 &= \frac{1}{m}\tilde{f}_1^{(a)} + \tilde{g}_1 + ((\mathbf{v}+\mathbf{v}_0)\times\boldsymbol{\omega})_1 + \frac{1}{m}\tau \\ &= \frac{\rho}{2m}\|\mathbf{v}+\mathbf{v}_0\|^2 S_{ref}\, C_x(\|\mathbf{v}+\mathbf{v}_0\|,\alpha) + \tilde{g}_1 \\ &\quad +((\mathbf{v}+\mathbf{v}_0)\times\boldsymbol{\omega})_1 + \frac{1}{m}\tau, \end{aligned} \tag{5.41}$$

where the angle of attack α is defined in (4.1), we have inserted the expression (3.22) for the dynamic pressure and neglected the influence of the control surface deflections $\bar{\delta}$ in the expression for C_x. The object of a virtual control for the thrust in (5.41) would be to give the right hand side a more suitable appearance (when the variable $\boldsymbol{\omega}$ is substituted for whatever total virtual control $\boldsymbol{\omega}^{des}$ for the angular velocity is finally selected).

The first term on the right hand side of (5.41) could be a target for the virtual thrust control to effect, but it acts stabilizing (for most aircraft configurations / flight conditions) and will therefore be omitted from consideration here. The second term, however, would be a suitable target for compensation via virtual control action. Still, to make things simple we choose here not to compenstate for any of these terms but introduce instead only a simple negative velocity feedback to achieve some control along the direction of $\mathbf{v}_0$, giving the virtual thrust control $\tau^{des}(\mathbf{v},\mathbf{v}_0,\mathbf{Q},\mathbf{Q}_0)$ as

$$\tau^{des}(\mathbf{v},\mathbf{v}_0,\mathbf{Q},\mathbf{Q}_0) = -\frac{c_\tau m v_1}{c_\ell + v_1^2}\Big(\frac{\mathbf{v}^T\mathbf{v}_0}{\|\mathbf{v}_0\|}\Big)^2 = -\frac{c_\tau m v_1}{c_\ell + v_1^2}\big\|P_{[\mathbf{v}_0]}(\mathbf{v})\big\|^2, \tag{5.42}$$

where c_τ, c_ℓ are positive constants and $P_{[\mathbf{v}_0]}(\mathbf{v})$ is the projection of $\mathbf{v}$ onto the one dimensional subspace spanned by $\mathbf{v}_0$. This type of virtual thrust control would, if $\mathbf{v}$ is large and mostly aligned with $\mathbf{v}_0$, approximately give a stable linear first order contribution to the dynamics in (5.41). (For small $\mathbf{v}$ this control would under the same conditions do essentially nothing.) Therefore, when c_τ is close to $c_\mathbf{v}$ it is clear from (5.39) and (5.40) that the combined effect of the virtual controls $\boldsymbol{\omega}_\mathbf{v}^{des}(\mathbf{v},\mathbf{v}_0)$ and $\tau^{des}(\mathbf{v},\mathbf{v}_0,\mathbf{Q},\mathbf{Q}_0)$ for large $\mathbf{v}$ is to give the overall system roughly first order stable (virtual) dynamics for the error $\mathbf{v}$.

Getting the orientation right. We now turn to the problem of controlling the attitude of the aircraft. When conceiving a solution to this problem we will, in analogy with the approach above, neglect the other part of the control problem, viz. the problem of aerodynamically stabilizing the aircraft. The attitude control is achieved by rotating the body along the shortest path from the current orientation to the desired orientation on the set of unit norm quaternions, which we here identify with (one "half" of) S^3, the unit sphere in ordinary four dimensional space. Such a shortest path is the same as a *geodesic* (in the ordinary Euclidean metric on the tangent space of S^3). A simple parameterization for this type of geodesic called Spherical Linear Interpolation (cf. Appendix A), or slerp, was introduced by Shoemake in 1985 [7] and is widely used in computer graphics.

The slerp that describes the path from unit norm quaternion $\mathcal{Q}_1$ to unit norm quaternion $\mathcal{Q}_0$ is given by

$$\mathcal{Q}(t) = \frac{\sin((1-t)\theta)\mathcal{Q}_0}{\sin(\theta)} + \frac{\sin(t\theta)\mathcal{Q}_1}{\sin(\theta)}, \quad t\in[0,1], \tag{5.43}$$

where θ is given by

$$\cos(\theta) = \mathcal{Q}_0^T \mathcal{Q}_1 \tag{5.44}$$

and the inner product on the right hand side is calculated as for ordinary vectors in four-space. The time-derivative of the slerp is easily calculated as

$$\dot{\mathcal{Q}}(t) = \frac{\theta}{\sin(\theta)}(\cos(t\theta)\mathcal{Q}_1 - \cos((1-t)\theta)\mathcal{Q}_0), \quad t \in [0,1], \tag{5.45}$$

and this shows that the motion along the path takes place at constant speed i.e. $\|\dot{\mathcal{Q}}(t)\| \equiv const$. In our application, where we want to design a feedback law based on the slerp formula (5.43), the start quaternion $\mathcal{Q}_1$ will be constantly changing and so we really only use the expression (5.45) for the slerp velocity vector, and evaluate it at the (changing) starting point. Indeed, at least in the case that $\mathbf{Q}_0$ is constant (i.e. $\boldsymbol{\omega}_0 = 0$) it is clear that what we want to achieve with the (virtual) control is

$$\dot{\mathbf{Q}} = c_{\mathbf{Q}} \dot{\mathcal{Q}}(0), \tag{5.46}$$

where $\mathbf{Q}$ is the state quaternion in (5.32) and $\mathbf{Q} + \mathbf{Q}_0, \mathbf{Q}_0$ in (5.32) are used instead of $\mathcal{Q}_0, \mathcal{Q}_1$ in (5.45), for some positive constant $c_{\mathbf{Q}}$. This gives a condition for the sought virtual angular velocity $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ for attitude control as

$$\frac{1}{2}\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0) = c_{\mathbf{Q}} \frac{\theta}{\sin(\theta)}\big(\mathbf{Q}_0 - \cos(\theta)(\mathbf{Q}+\mathbf{Q}_0)\big), \tag{5.47}$$

where now θ is given by

$$\cos(\theta) = (\mathbf{Q}+\mathbf{Q}_0)^T \mathbf{Q}_0. \tag{5.48}$$

From (5.47) it might appear impossible to solve (uniquely) for $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ since the matrix $\mathbf{B}$ is not square, but if we remember that the left hand side of (5.47) is really just another way of writing the product of two quaternions (cf. the discussion on how (5.16)–(5.19) is transformed into (5.31)–(5.34)), one unit norm and one pure, we can determine $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ explicitly. Working through the algebra we get (cf. Appendix A)

$$\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0) = c_{\mathbf{Q}} \frac{2\theta}{\sin(\theta)} \Im(\mathbf{Q}^c \circ \mathbf{Q}_0) \tag{5.49}$$

(with θ as in (5.48)) where $\Im(\,\cdot\,)$ denotes quaternion imaginary part, $(\,\cdot\,)^c$ denotes quaternion conjugation (sign change on the imaginary part) and $\circ$ denotes quaternion product. In case $\boldsymbol{\omega}_0$ is constant but nonzero it is clear, after a moments contemplation, that the same principle for selecting $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ ought to apply, and that the resulting dynamics in this “moving” scenario on $\mathcal{S}^3$ then becomes the same as in (5.32), if we replace $\boldsymbol{\omega}$ there by $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$. We then have

$$\frac{d}{dt}(\mathbf{Q}+\mathbf{Q}_0) = \dot{\mathbf{Q}} + \dot{\mathbf{Q}}_0 = \frac{1}{2}\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)(\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0) + \boldsymbol{\omega}_0), \tag{5.50}$$

with $\dot{\mathbf{Q}}$ as in (5.46) and $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ as in (5.49).

Putting together the complete controller. The complete (virtual) controller corresponding to ℓ in Thm. 5.2.5 is now given by the vector

$$\begin{pmatrix} \boldsymbol{\omega}^{des}(\mathbf{v}, \mathbf{v}_0, \mathbf{Q}, \mathbf{Q}_0) \\ \tau^{des}(\mathbf{v}, \mathbf{v}_0, \mathbf{Q}, \mathbf{Q}_0) \end{pmatrix} = \begin{pmatrix} \boldsymbol{\omega}_{\mathbf{v}}^{des}(\mathbf{v}, \mathbf{v}_0) + \boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0) \\ \tau^{des}(\mathbf{v}, \mathbf{v}_0, \mathbf{Q}, \mathbf{Q}_0), \end{pmatrix}, \tag{5.51}$$

with the components on the right hand side given by (5.39),(5.49) and (5.42). It should be pointed out, however, that many other controller solutions are possible. Once the model has been put on the standard form as in (5.31)–(5.34) there are many ways of constructing stabilizing controllers using backstepping. The slerp is, as remarked before, frequently appearing in texts on computer graphics but to the best of the authors' knowledge the application to aircraft (or indeed rigid body) control is new.

5.3.5 Stability. We shall now give a proof of stability for the proposed controller. First we introduce the Lyapunov function that we are going to base our development on and then we give some mathematical details that will be used in the analysis. Then, we give a proof of stability. The calculations use some standard properties of quaternions that can be found in standard texts on flight mechanics, such as [3].

Lyapunov function. As a candidate for the "inner" Lyapunov function $\mathcal{V}$ as in Thm.5.2.5 we shall take

$$\mathcal{V}(\mathbf{v},\mathbf{Q}) = \frac{\gamma_{\mathbf{v}}}{2}\|\mathbf{v}\|^2 + \frac{\gamma_{\mathbf{Q}}}{2}\|\mathbf{Q}\|^2, \tag{5.52}$$

where $\gamma_{\mathbf{v}}, \gamma_{\mathbf{Q}}$ are two positive constants [4] (to be determined later) and the norms are ordinary 2-norms in $\mathbb{R}^3$ and $\mathbb{R}^4$, respectively. With this choice, the time derivative of $\mathcal{V}$ along the solutions of (5.31),(5.32), and with $\boldsymbol{\omega}, \tau$ replaced by $\boldsymbol{\omega}^{des}, \tau^{des}$ as in (5.51), is given by

$$\begin{aligned}
\frac{d}{dt}\mathcal{V}(\mathbf{v},\mathbf{Q}) &= \gamma_{\mathbf{v}}\mathbf{v}^T\dot{\mathbf{v}} + \gamma_{\mathbf{Q}}\mathbf{Q}^T\dot{\mathbf{Q}} \\
&= \gamma_{\mathbf{v}}\mathbf{v}^T\big(\frac{1}{m}\tilde{\mathbf{f}}(\mathbf{v},\mathbf{v}_0) + \tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) + (\mathbf{v}+\mathbf{v}_0)\times\boldsymbol{\omega}^{des} + \frac{1}{m}\tau^{des}\mathbf{e}_x\big) \\
&\quad \frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{A}(\boldsymbol{\omega}_0)\mathbf{Q} + \gamma_{\mathbf{Q}}\mathbf{Q}^T\big(\frac{1}{2}\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}^{des}\big) \\
&= \frac{\gamma_{\mathbf{v}}}{m}\mathbf{v}^T\tilde{\mathbf{f}}(\mathbf{v},\mathbf{v}_0) + \gamma_{\mathbf{v}}\mathbf{v}^T\tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) \\
&\quad +\gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0\times\boldsymbol{\omega}_{\mathbf{v}}^{des}) + \gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0\times\boldsymbol{\omega}_{\mathbf{Q}}^{des}) + \gamma_{\mathbf{v}}\frac{v_1}{m}\tau^{des} \\
&\quad +\frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{v}}^{des} + \frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{Q}}^{des},
\end{aligned} \tag{5.53}$$

where we have used the fact that $\mathbf{A}(\boldsymbol{\omega}_0)$ is skew symmetric. We are now going to study the terms on the right hand side of (5.53) in more detail.

Preliminaries for the Lyapunov function terms. To begin with we shall collect some results that will be used several times below. The first result is the standard matrix-vector representation of a quaternion product, of which we have seen a special case in (3.4) before.

Let $\mathcal{Q}_1 = (a_1, \mathbf{b}_1)$, $\mathcal{Q}_2 = (a_2, \mathbf{b}_2)$ be two quaternions with real parts $a_1, a_2 \in \mathbb{R}$ and imaginary parts $\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^3$, respectively. Then the quaternion product $\mathcal{Q}_1 \circ \mathcal{Q}_2$ can be written in terms of an ordinary matrix-vector product as

$$\mathcal{Q}_1 \circ \mathcal{Q}_2 = \mathbf{T}(\mathcal{Q}_1)\mathcal{Q}_2,$$

where the matrix $\mathbf{T}(\mathcal{Q}_1)$ is given by

$$\mathbf{T}(\mathcal{Q}_1) = \begin{bmatrix} a_1 & -\mathbf{b}_1^T \\ \mathbf{b}_1 & \mathbf{C}(\mathbf{b}_1) + a_1\mathbf{I}_{3\times 3} \end{bmatrix}, \tag{5.54}$$

and $\mathbf{C}$ is the skew-symmetric matrix in (5.30) giving the vector product. The quaternion norm $N(\,\cdot\,)$ is defined as the square of the ordinary 2-norm for vectors in $\mathbb{R}^4$ and a fundamental property of the quaternion norm is

$$N(\mathcal{Q}_1 \circ \mathcal{Q}_2) = N(\mathcal{Q}_1)N(\mathcal{Q}_2). \tag{5.55}$$

[4]The reason we introduce two independent weighting constants here is that we want to be able to control the relative magnitude of all three terms of the resulting total control law in (5.12).

It therefore follows that if $\mathcal{Q}_1$ is a unit norm quaternion we have

$$\|\mathbf{T}(\mathcal{Q}_1)\| = \sup_{\|\mathcal{Q}_2\|=1} \|\mathbf{T}(\mathcal{Q}_1)\mathcal{Q}_2\| = \sup_{\|\mathcal{Q}_2\|=1} \|\mathcal{Q}_1 \circ \mathcal{Q}_2\| = \sup_{\|\mathcal{Q}_2\|=1} \big(N(\mathcal{Q}_1)N(\mathcal{Q}_2)\big)^{1/2} = 1, \quad (5.56)$$

where $\|\cdot\|$ denotes both the ordinary 2-norm on $\mathbb{R}^4$ and its induced counterpart on $\mathbb{R}^{4\times4}$, respectively.

If $\mathbf{u}, \mathbf{U} \in \mathbb{R}^3$ are two vectors and $\mathbf{R} \in \mathbb{R}^{3\times3}$ is a rotation matrix such that $\mathbf{U} = \mathbf{R}\mathbf{u}$ we have

$$(0, \mathbf{u}) = \mathcal{Q}^c \circ (0, \mathbf{U}) \circ \mathcal{Q}, \quad (5.57)$$

where $\mathcal{Q}$ is the unit norm quaternion corresponding to $\mathbf{R}$ and $(\cdot)^c$ denotes quaternion conjugation (sign change on the imaginary part). (In fact, we can define $\mathcal{Q}$ by this relation.) In connection with expressions of the form (5.57) we shall have reason to consider also expressions of the form

$$\Im(\mathcal{Q}_1 \circ (0, \mathbf{U}) \circ \mathcal{Q}_2), \quad (5.58)$$

where $\mathcal{Q}_1$,$\mathcal{Q}_2$ are two arbitrary quaternions. The expression (5.58) will occur in connection with matrix-vector products of the form

$$\begin{aligned}
\begin{bmatrix} 0 \\ \Im(\mathcal{Q}_1 \circ (0, \mathbf{U}) \circ \mathcal{Q}_2) \end{bmatrix} &= \begin{bmatrix} 0 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times1} & \mathbf{I}_{3\times3} \end{bmatrix} \mathcal{Q}_1 \circ (0, \mathbf{U}) \circ \mathcal{Q}_2 \\
&= \begin{bmatrix} 0 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times1} & \mathbf{I}_{3\times3} \end{bmatrix} \mathbf{T}(\mathcal{Q}_1 \circ (0, \mathbf{U}))\mathcal{Q}_2 \\
&= \|\mathcal{Q}_1\|\|\mathbf{U}\| \begin{bmatrix} 0 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times1} & \mathbf{I}_{3\times3} \end{bmatrix} \mathbf{T}(\frac{\mathcal{Q}_1}{\|\mathcal{Q}_1\|} \circ (0, \frac{\mathbf{U}}{\|\mathbf{U}\|}))\mathcal{Q}_2 \\
&= \|\mathcal{Q}_1\|\|\mathbf{U}\|\, \mathbf{S}(\mathcal{Q}_1, \mathbf{U})\mathcal{Q}_2,
\end{aligned}$$

where the matrix $\mathbf{S}(\mathcal{Q}_1, \mathbf{U})$ is given by

$$\mathbf{S}(\mathcal{Q}_1, \mathbf{U}) = \begin{bmatrix} 0 & \mathbf{0}_{1\times3} \\ \mathbf{0}_{3\times1} & \mathbf{I}_{3\times3} \end{bmatrix} \mathbf{T}(\frac{\mathcal{Q}_1}{\|\mathcal{Q}_1\|} \circ (0, \frac{\mathbf{U}}{\|\mathbf{U}\|}))$$

From (5.55) and (5.56) it follows that we have a bound for the matrix $\mathbf{S}(\mathcal{Q}_1, \mathbf{U})$ as

$$\|\mathbf{S}(\mathcal{Q}_1, \mathbf{U})\| \le 1.$$

Finally we shall make an elementary observation about maximization of bilinear forms. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^4$ be two arbitrary vectors. By the Cauchy-Schwarz inequality we then have

$$\sup_{\mathbf{D}\in\mathbb{R}^{4\times4}, \|\mathbf{D}\|\le1} \mathbf{y}^T\mathbf{D}\mathbf{x} \le \|\mathbf{x}\|\|\mathbf{y}\|.$$

It is easy to see that the supremum indeed is a maximum and is attained for the rank-1 matrix $\hat{\mathbf{D}}_1$ given by

$$\hat{\mathbf{D}}_1 = \frac{1}{\|\mathbf{x}\|\|\mathbf{y}\|}\mathbf{y}\mathbf{x}^T.$$

Thus, if we collect three vectors $\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3 \in \mathbb{R}^4$ that are orthonormal and orthogonal to $\mathbf{x}$ and likewise collect three more vectors $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \in \mathbb{R}^4$ that are orthonormal and orthogonal to $\mathbf{y}$, and form the matrix $\hat{\mathbf{D}}(\mathbf{x}, \mathbf{y})$ given by

$$\hat{\mathbf{D}}(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} \frac{\mathbf{y}}{\|\mathbf{y}\|} & \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{bmatrix} \begin{bmatrix} \frac{\mathbf{x}^T}{\|\mathbf{x}\|} \\ \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \mathbf{u}_3^T \end{bmatrix} \quad (5.59)$$

we have

$$\sup_{\mathbf{D}\in\mathbb{R}^{4\times 4}, \|\mathbf{D}\|\leq 1} \mathbf{y}^T\mathbf{D}\mathbf{x} = \mathbf{y}^T\hat{\mathbf{D}}(\mathbf{x},\mathbf{y})\mathbf{x} = \|\mathbf{x}\|\|\mathbf{y}\|$$

where the maximizer $\hat{\mathbf{D}}(\mathbf{x},\mathbf{y})$ is an orthogonal matrix.

Properties of Lyapunov function terms. *Gravity term.* If we define (in $\mathbb{R}^4$) the embedded velocity error $\bar{\mathbf{v}}$ as

$$\bar{\mathbf{v}} = \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix},$$

where $\mathbf{v}$, as before, is the velocity error in (5.31), we can, using (5.57) write the second term on the right in (5.53) as

$$\begin{aligned}
\gamma_{\mathbf{v}}\mathbf{v}^T\tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) &= \gamma_{\mathbf{v}}\bar{\mathbf{v}}^T\big((\mathbf{Q}+\mathbf{Q}_0)^c \circ (0,\mathbf{G}) \circ (\mathbf{Q}+\mathbf{Q}_0) - \mathbf{Q}_0^c \circ (0,\mathbf{G}) \circ \mathbf{Q}_0\big) \\
&= \gamma_{\mathbf{v}}\bar{\mathbf{v}}^T\big(\mathbf{Q}^c \circ (0,\mathbf{G}) \circ \mathbf{Q} + \mathbf{Q}_0^c \circ (0,\mathbf{G}) \circ \mathbf{Q} - (\mathbf{Q}_0^c \circ (0,\mathbf{G}) \circ \mathbf{Q})^c\big) \\
&= \gamma_{\mathbf{v}}\bar{\mathbf{v}}^T\big(\mathbf{Q}^c \circ (0,\mathbf{G}) \circ \mathbf{Q} + 2(0,\Im(\mathbf{Q}_0^c \circ (0,\mathbf{G}) \circ \mathbf{Q}))\big) \\
&= \gamma_{\mathbf{v}}\|\mathbf{Q}\|\|\mathbf{G}\|\bar{\mathbf{v}}^T\mathbf{T}\big(\frac{\mathbf{Q}^c}{\|\mathbf{Q}\|} \circ (0,\frac{\mathbf{G}}{\|\mathbf{G}\|})\big)\mathbf{Q} + 2\gamma_{\mathbf{v}}\|\mathbf{G}\|\bar{\mathbf{v}}^T\mathbf{S}\big(\mathbf{Q}_0^c,\frac{\mathbf{G}}{\|\mathbf{G}\|}\big)\mathbf{Q} \\
&\leq 20\,\gamma_{\mathbf{v}}|\bar{\mathbf{v}}^T\mathbf{T}\big(\frac{\mathbf{Q}^c}{\|\mathbf{Q}\|} \circ (0,\frac{\mathbf{G}}{\|\mathbf{G}\|})\big)\mathbf{Q}| + 20\,\gamma_{\mathbf{v}}|\bar{\mathbf{v}}^T\mathbf{S}\big(\mathbf{Q}_0^c,\frac{\mathbf{G}}{\|\mathbf{G}\|}\big)\mathbf{Q}| \\
&\leq 40\,\gamma_{\mathbf{v}} \sup_{\mathbf{D}\in\mathbb{R}^{4\times 4}, \|\mathbf{D}\|=1} \bar{\mathbf{v}}^T\mathbf{D}\mathbf{Q} \\
&= 40\,\gamma_{\mathbf{v}}\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q},
\end{aligned} \tag{5.60}$$

where $\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})$ is the matrix function in (5.59) and we have used the facts that $\|\mathbf{Q}_0\| = 1$, $\|\mathbf{Q}\| \leq 2$ and that for the inertial gravity vector $\mathbf{G}$ (defined in connection with (3.8)) we have $\|\mathbf{G}\| \leq 10$.

Desired velocity control term. For the term involving $\mathbf{v}$ and the virtual control $\boldsymbol{\omega}_{\mathbf{v}}^{des}$ we have similarly

$$\begin{aligned}
\gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0 \times \boldsymbol{\omega}_{\mathbf{v}}^{des}) &= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\mathbf{v}^T\big(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|} \times (\mathbf{v} \times \frac{\mathbf{v}_0}{\|\mathbf{v}_0\|})\big) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\mathbf{v}^T\Big(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|} \times \big((P_{[\mathbf{v}_0]}(\mathbf{v}) + P_{[\mathbf{v}_0]^\perp}(\mathbf{v})\big) \times \frac{\mathbf{v}_0}{\|\mathbf{v}_0\|}\big)\Big) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\mathbf{v}^T\big(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|} \times P_{[\mathbf{v}_0]^\perp}(\mathbf{v}) \times \frac{\mathbf{v}_0}{\|\mathbf{v}_0\|}\big)\big) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\mathbf{v}^T P_{[\mathbf{v}_0]^\perp}(\mathbf{v}) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\big(P_{[\mathbf{v}_0]}(\mathbf{v}) + P_{[\mathbf{v}_0]^\perp}(\mathbf{v})\big)^T P_{[\mathbf{v}_0]^\perp}(\mathbf{v}) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{v}}\|P_{[\mathbf{v}_0]^\perp}(\mathbf{v})\|^2,
\end{aligned} \tag{5.61}$$

where the fourth equality follows from standard properties of the vector product.

Undesired velocity control term. In order to obtain a bound for the term in (5.53) involving both $\mathbf{v}$ and $\boldsymbol{\omega}_{\mathbf{Q}}^{des}$ we shall assume that $|\theta| \leq \pi/2$. We can then write

$$\begin{aligned}
\gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0 \times \boldsymbol{\omega}_{\mathbf{Q}}^{des}) &= \gamma_{\mathbf{v}}\mathbf{v}^T\big(\mathbf{v}_0 \times \frac{2c_{\mathbf{Q}}\theta}{\sin(\theta)}\Im(\mathbf{Q}^c \circ \mathbf{Q}_0)\big) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{Q}}\frac{2\theta}{\sin(\theta)}\mathbf{v}^T\mathbf{C}(\mathbf{v}_0)\Im(\mathbf{Q}_0^c \circ \mathbf{Q})\big) \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{Q}}\frac{2\theta}{\sin(\theta)}\bar{\mathbf{v}}^T\begin{bmatrix} 0 & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{3\times 1} & \mathbf{C}(\mathbf{v}_0)\end{bmatrix}\mathbf{T}(\mathbf{Q}_0^c)\mathbf{Q} \\
&= -\gamma_{\mathbf{v}}c_{\mathbf{Q}}\frac{2\theta}{\sin(\theta)}\|\mathbf{v}_0\|\bar{\mathbf{v}}^T\begin{bmatrix} 0 & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{3\times 1} & \mathbf{C}(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|})\end{bmatrix}\mathbf{T}(\mathbf{Q}_0^c)\mathbf{Q} \\
&\leq \gamma_{\mathbf{v}}c_{\mathbf{Q}}\pi\|\mathbf{v}_0\| \sup_{\mathbf{D}\in\mathbb{R}^{4\times 4},\|\mathbf{D}\|=1}\bar{\mathbf{v}}^T\mathbf{D}\mathbf{Q} \\
&= \gamma_{\mathbf{v}}c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q}, \qquad (5.62)
\end{aligned}$$

where we have used the easily verified fact that $\|\mathbf{C}(\mathbf{v}_0)\| = \|\mathbf{v}_0\|$, the bound (5.56) for $\mathbf{T}(\mathbf{Q}_0^c)$ and the bound $1 \leq \theta/\sin(\theta) \leq \pi/2$ for $|\theta| \leq \pi/2$.

Virtual thrust term. The virtual thrust control term in (5.53) is simply

$$\gamma_{\mathbf{v}}\frac{v_1}{m}\tau^{des} = -\gamma_{\mathbf{v}}c_{\tau}\frac{v_1^2}{c_\ell + v_1^2}\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2. \qquad (5.63)$$

Undesired attitude control term. For the term involving $\mathbf{Q}$ and $\boldsymbol{\omega}_{\mathbf{v}}^{des}$ we start by recalling that by the definition (5.29) of $\mathbf{B}$ we have

$$\begin{aligned}
\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)(\mathbf{v}\times\mathbf{v}_0) &= \mathbf{A}(\mathbf{v}\times\mathbf{v}_0)(\mathbf{Q}+\mathbf{Q}_0) \\
&= -\mathbf{A}(\mathbf{C}(\mathbf{v}_0)\mathbf{v})(\mathbf{Q}+\mathbf{Q}_0) \\
&= -(\mathbf{Q}+\mathbf{Q}_0)\circ(0,\mathbf{C}(\mathbf{v}_0)\mathbf{v}) \\
&= -\|\mathbf{v}_0\|\mathbf{T}(\mathbf{Q}+\mathbf{Q}_0)\begin{bmatrix} 0 & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{3\times 1} & \mathbf{C}(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|})\end{bmatrix}\bar{\mathbf{v}}.
\end{aligned}$$

With this we obtain the following estimate

$$\begin{aligned}
\frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{v}}^{des} &= -\frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|^2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)(\mathbf{v}\times\mathbf{v}_0) \\
&= \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\mathbf{Q}^T\mathbf{T}(\mathbf{Q}+\mathbf{Q}_0)\begin{bmatrix} 0 & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{3\times 1} & \mathbf{C}(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|})\end{bmatrix}\bar{\mathbf{v}} \\
&= -\frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\bar{\mathbf{v}}^T\begin{bmatrix} 0 & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{3\times 1} & \mathbf{C}(\frac{\mathbf{v}_0}{\|\mathbf{v}_0\|})\end{bmatrix}\mathbf{T}^T(\mathbf{Q}+\mathbf{Q}_0)\mathbf{Q} \\
&\leq \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\sup_{\mathbf{D}\in\mathbb{R}^{4\times 4},\|\mathbf{D}\|=1}\bar{\mathbf{v}}^T\mathbf{D}\mathbf{Q} \\
&= \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q}. \qquad (5.64)
\end{aligned}$$

Desired attitude control term. From the defining relation (5.47) for $\boldsymbol{\omega}_{\mathbf{Q}}^{des}$ we have

$$\begin{aligned}
\frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{Q}}^{des} &= \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}\mathbf{Q}^T\big(\mathbf{Q}_0 - \cos(\theta)(\mathbf{Q}+\mathbf{Q}_0)\big) \\
&= -\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 + \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}(1-\cos(\theta))\mathbf{Q}^T\mathbf{Q}_0 \\
&= -\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}(\mathbf{Q}^T\mathbf{Q}_0)^2, \qquad (5.65)
\end{aligned}$$

where we have used the fact that (5.48) implies $1-\cos(\theta) = -\mathbf{Q}^T\mathbf{Q}_0$ for the last equality.

Before we are ready for our first stability result we must introduce the following assumption about the local behavior of the aerodynamic forces near an equilibrium.

Assumption 5.3.1. Assume that the function $\mathbf{v} \mapsto \mathbf{v}^T\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0)$ in (5.22) is locally negative definite, i.e. for some open set $\mathcal{U} \subseteq \mathbb{R}^3$ containing the origin the following expansion holds

$$\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) = \mathbf{F}(\mathbf{v}_0)\mathbf{v} + \mathcal{O}(\|\mathbf{v}\|^2), \quad \mathbf{v} \in \mathcal{U},$$

where the symmetric part $\frac{1}{2}(\mathbf{F}(\mathbf{v}_0) + \mathbf{F}^T(\mathbf{v}_0))$ of $\mathbf{F}(\mathbf{v}_0)$ is a negative definite matrix (of "stability derivatives"), and that the constants c_ℓ and $c_\mathbf{v}$ can be selected such that

$$\frac{1}{m}\mathbf{v}^T\tilde{\mathbf{f}}(\mathbf{v},\mathbf{v}_0) - c_\mathbf{v}\|\mathbf{v}\|^2 + c_\mathbf{v}\big(1 - \frac{v_1^2}{c_\ell + v_1^2}\big)\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 \le -(\sigma c_\mathbf{v} + c_c)\|\mathbf{v}\|^2, \quad \mathbf{v} \in \mathcal{U}, \tag{5.66}$$

for some positive constants σ, c_c such that $\sigma > \pi/2$.

The condition of negative definiteness of the symmetric part of $\mathbf{F}(\mathbf{v}_0)$ is for most aircraft configurations satisfied under normal flying conditions. To see that also the condition (5.66) can be reasonable to assume we let $\lambda_3 \le \lambda_2 \le \lambda_1 < 0$ be the eigenvalues of the symmetric part $\frac{1}{2}(\mathbf{F}(\mathbf{v}_0)+\mathbf{F}^T(\mathbf{v}_0))$ of $\mathbf{F}(\mathbf{v}_0)$. Then, locally around $\mathbf{v} = 0$ we have

$$\begin{aligned}
&\frac{1}{m}\mathbf{v}^T\tilde{\mathbf{f}}(\mathbf{v},\mathbf{v}_0) - c_\mathbf{v}\|\mathbf{v}\|^2 + c_\mathbf{v}\big(1 - \frac{v_1^2}{c_\ell + v_1^2}\big)\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 = \\
&\frac{1}{m}\mathbf{v}^T\mathbf{F}(\mathbf{v}_0)\mathbf{v} - c_\mathbf{v}\|\mathbf{v}\|^2 + c_\mathbf{v}\big(1 - \frac{v_1^2}{c_\ell + v_1^2}\big)\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 + \mathcal{O}(\|\mathbf{v}\|^3) \le \\
&\frac{\lambda_1}{m}\|\mathbf{v}\|^2 - c_\mathbf{v}\|\mathbf{v}\|^2 + c_\mathbf{v}\big(1 - \frac{v_1^2}{c_\ell + v_1^2}\big)\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 + \mathcal{O}(\|\mathbf{v}\|^3) = \\
&\frac{\lambda_1}{m}\|\mathbf{v}\|^2 - c_\mathbf{v}(\|\mathbf{v}\|^2 - \|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2) + \mathcal{O}(\|\mathbf{v}\|^3) \le \\
&\frac{\lambda_1}{m}\|\mathbf{v}\|^2 + \mathcal{O}(\|\mathbf{v}\|^3),
\end{aligned} \tag{5.67}$$

where we have used the fact that

$$\|\mathbf{v}\|^2 = \|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 + \|P_{[\mathbf{v}_0]^\perp}(\mathbf{v})\|^2. \tag{5.68}$$

The first term on the right hand side in (5.67) is negative definite and if moreover

$$\frac{\lambda_1}{m} < -\sigma c_\mathbf{v} \tag{5.69}$$

then there exists a $c_c > 0$ such that (5.66) is satisfied in some $\mathcal{U} \subseteq \mathbb{R}^3$ containing the origin. For many aircraft configurations the set $\mathcal{U}$ can be quite large and the condition (5.66) can be fulfilled not only locally. Indeed, when $\mathbf{v}$ becomes large the left hand side of (5.66) is determined by the first two terms only. We note also that (5.69) gives a bound for the possible velocity feedback gain values $c_\mathbf{v}$ as

$$c_\mathbf{v} < -\frac{\lambda_1}{m\sigma}. \tag{5.70}$$

We have now everything in place for our first stability result.

Lemma 5.3.2. *Suppose that Assumption 5.3.1 holds, for some set $\mathcal{U}$ and constants $\sigma > \pi/2, c_c > 0$, let $c_\tau = c_\mathbf{v}$ and assume that there exists a positive constant $\theta_0 < \pi/2$ such that*

$$\gamma_\mathbf{v}(40 + c_\mathbf{Q}\pi\|\mathbf{v}_0\|) + \frac{\gamma_\mathbf{Q}c_\mathbf{v}}{2\|\mathbf{v}_0\|} \le 2\sqrt{\gamma_\mathbf{v}c_\mathbf{v}\gamma_\mathbf{Q}c_\mathbf{Q}\sigma\theta_0\cot(\theta_0)}. \tag{5.71}$$

Then the Lyapunov function candidate in (5.52) is indeed an "inner" Lyapunov function for the integrator backstepping problem for (5.31)–(5.34) over the domain of $(\mathbf{v}, \mathbf{Q})$ *such that* $\mathbf{v} \in \mathcal{U}$ *and* θ *in (5.48) satisfies* $|\theta| < \theta_0$*, i.e. the Lyapunov function candidate satisfies condition (5.6) in Assumption 5.2.4 over this domain.*

Proof. For the time derivative of the Lyapunov function candidate (5.52) along the solutions to the system (5.31), (5.32), with the virtual controls (5.51) inserted, we get using (5.53) and (5.60)–(5.65) the following estimate

$$\begin{aligned} \frac{d}{dt}\mathcal{V}(\mathbf{v},\mathbf{Q}) &= \frac{\gamma_{\mathbf{v}}}{m}\mathbf{v}^T\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) + \gamma_{\mathbf{v}}\mathbf{v}^T\tilde{\mathbf{g}}(\mathbf{Q},\mathbf{Q}_0) + \gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0\times\boldsymbol{\omega}_{\mathbf{v}}^{des}) \\ &\quad +\gamma_{\mathbf{v}}\mathbf{v}^T(\mathbf{v}_0\times\boldsymbol{\omega}_{\mathbf{Q}}^{des}) + \gamma_{\mathbf{v}}\frac{v_1}{m}\tau^{des} \\ &\quad +\frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{v}}^{des} + \frac{\gamma_{\mathbf{Q}}}{2}\mathbf{Q}^T\mathbf{B}(\mathbf{Q}+\mathbf{Q}_0)\boldsymbol{\omega}_{\mathbf{Q}}^{des} \\ &\le \frac{\gamma_{\mathbf{v}}}{m}\mathbf{v}^T\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) + 40\,\gamma_{\mathbf{v}}\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} - \gamma_{\mathbf{v}}c_{\mathbf{v}}\|P_{[\mathbf{v}_0]^\perp}(\mathbf{v})\|^2 \\ &\quad +\gamma_{\mathbf{v}}c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} - \gamma_{\mathbf{v}}c_\tau\frac{v_1^2}{c_\ell+v_1^2}\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 \\ &\quad +\frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}(\mathbf{Q}^T\mathbf{Q}_0)^2 \\ &= \frac{\gamma_{\mathbf{v}}}{m}\mathbf{v}^T\tilde{\mathbf{f}}^{(a)}(\mathbf{v},\mathbf{v}_0) - \gamma_{\mathbf{v}}c_{\mathbf{v}}\|\bar{\mathbf{v}}\|^2 + \gamma_{\mathbf{v}}c_{\mathbf{v}}\big(1-\frac{v_1^2}{c_\ell+v_1^2}\big)\|P_{[\mathbf{v}_0]}(\mathbf{v})\|^2 \\ &\quad -\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}(\mathbf{Q}^T\mathbf{Q}_0)^2 \\ &\quad +\big(\gamma_{\mathbf{v}}(40+c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|) + \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\big)\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q}, \end{aligned} \tag{5.72}$$

where we in the last equality have used the facts (5.68) and $\|\bar{\mathbf{v}}\| = \|\mathbf{v}\|$ as well as the assumption $c_\tau = c_{\mathbf{v}}$. If we apply condition (5.66) we can proceed one step further with (5.72) to obtain

$$\begin{aligned} \frac{d}{dt}\mathcal{V}(\mathbf{v},\mathbf{Q}) &\le -\gamma_{\mathbf{v}}c_{\mathbf{v}}\sigma\|\bar{\mathbf{v}}\|^2 - \gamma_{\mathbf{v}}c_c\|\bar{\mathbf{v}}\|^2 \\ &\quad -\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\frac{\theta}{\sin(\theta)}(\mathbf{Q}^T\mathbf{Q}_0)^2 \\ &\quad +\big(\gamma_{\mathbf{v}}(40+c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|) + \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\big)\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} \end{aligned} \tag{5.73}$$

From condition (5.71) we get for $|\theta| < \theta_0$ that

$$\begin{aligned} &-\gamma_{\mathbf{v}}c_{\mathbf{v}}\sigma\|\bar{\mathbf{v}}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta\cot(\theta)\|\mathbf{Q}\|^2 \\ &\qquad +\big(\gamma_{\mathbf{v}}(40+c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|) + \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\big)\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} \le \\ &-\gamma_{\mathbf{v}}c_{\mathbf{v}}\sigma\|\bar{\mathbf{v}}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta_0\cot(\theta_0)\|\mathbf{Q}\|^2 \\ &\qquad +\big(\gamma_{\mathbf{v}}(40+c_{\mathbf{Q}}\pi\|\mathbf{v}_0\|) + \frac{\gamma_{\mathbf{Q}}c_{\mathbf{v}}}{2\|\mathbf{v}_0\|}\big)\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} \le \\ &-\gamma_{\mathbf{v}}c_{\mathbf{v}}\sigma\|\bar{\mathbf{v}}\|^2 - \gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta_0\cot(\theta_0)\|\mathbf{Q}\|^2 \\ &+2\sqrt{\gamma_{\mathbf{v}}c_{\mathbf{v}}\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\sigma\theta_0\cot(\theta_0)}\,\bar{\mathbf{v}}^T\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q} = \\ &-\|\sqrt{\gamma_{\mathbf{v}}c_{\mathbf{v}}\sigma}\,\bar{\mathbf{v}} - \sqrt{\gamma_{\mathbf{Q}}c_{\mathbf{Q}}\theta_0\cot(\theta_0)}\,\hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q}\|^2, \end{aligned} \tag{5.74}$$

and if we combine (5.73) with (5.74) we therefore have

$$\begin{aligned} \frac{d}{dt}\mathcal{V}(\mathbf{v},\mathbf{Q}) &\leq -\gamma_{\mathbf{v}} c_c \|\bar{\mathbf{v}}\|^2 - \gamma_{\mathbf{Q}} c_{\mathbf{Q}} \frac{\theta}{\sin(\theta)} (\mathbf{Q}^T \mathbf{Q}_0)^2 \\ &\quad - \|\sqrt{\gamma_{\mathbf{v}} c_{\mathbf{v}} \sigma} \bar{\mathbf{v}} - \sqrt{\gamma_{\mathbf{Q}} c_{\mathbf{Q}} \theta_0 \cot(\theta_0)} \hat{\mathbf{D}}(\bar{\mathbf{v}},\mathbf{Q})\mathbf{Q}\|^2 \\ &\leq -\gamma_{\mathbf{v}} c_c \|\mathbf{v}\|^2 - \gamma_{\mathbf{Q}} c_{\mathbf{Q}} \frac{\theta}{\sin(\theta)} (\mathbf{Q}^T \mathbf{Q}_0)^2, \end{aligned} \tag{5.75}$$

for $\mathbf{v} \in \mathcal{U}$, $|\theta| < \theta_0$. Thus, if we take

$$-W(\mathbf{v},\mathbf{Q}) = -\gamma_{\mathbf{v}} c_c \|\mathbf{v}\|^2 - \gamma_{\mathbf{Q}} c_{\mathbf{Q}} \frac{\theta}{\sin(\theta)} (\mathbf{Q}^T \mathbf{Q}_0)^2, \tag{5.76}$$

and make the identifications (5.35), (5.36) we see that we satisfy condition (5.6) in Assumption 5.2.4 for $\mathbf{v} \in \mathcal{U}, |\theta| < \theta_0$. ☐

Remark 5.3.3. Even though the Assumption 5.2.4 is global in nature it is clear that it can be applied in a local version, as is done here. It may not be immediately clear that the second term in (5.76) is negative definite in $\mathbf{Q}$ but if we recall that for $\theta \neq 0$ we have $1 > \cos(\theta) = (\mathbf{Q} + \mathbf{Q}_0)^T \mathbf{Q}_0 = \mathbf{Q}^T \mathbf{Q}_0 + 1$ and thus $\mathbf{Q}^T \mathbf{Q}_0 < 0$, and at the same time $\theta / \sin(\theta) > 0$ for $|\theta| < \pi/2$ this should be clear.

Having established the above lemma we can now present our main result.

Theorem 5.3.4. *Assume that the conditions in (5.3.2) are fulfilled. Then the integrator backstepping problem for the attitude-velocity control problem for (5.31)–(5.34) is solvable using the standard method in Theorem 5.2.5 (using the identifications in (5.35)–(5.38)) and one stabilizing control law is given by (5.12).*

Proof. The result follows immediately from Lemma 5.3.2 above. ☐

It is instructive to take a closer look at the condition (5.71), which equivalently can be written

$$\frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}}(40 + c_{\mathbf{Q}} \pi \|\mathbf{v}_0\|) + \frac{c_{\mathbf{v}}}{2\|\mathbf{v}_0\|} \leq 2\sqrt{\frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} c_{\mathbf{v}} c_{\mathbf{Q}} \sigma \theta_0 \cot(\theta_0)}. \tag{5.77}$$

Solutions to the condition (5.77) for the Lyapunov gain ratio $\gamma_{\mathbf{v}}/\gamma_{\mathbf{Q}}$ and orientation feedback gain $c_{\mathbf{Q}}$ can be expressed in terms of the velocity feedback gain constant $c_{\mathbf{v}}$ as

$$0 < \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} < \frac{2 c_{\mathbf{v}} \sigma \theta_0 \cot(\theta_0)}{\pi (c_{\mathbf{v}} + 80\|\mathbf{v}_0\|)},$$

$$\frac{-c_{\mathbf{v}} \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} - 80\|\mathbf{v}_0\| \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} + 4 c_{\mathbf{v}} \sigma \theta_0 \cot(\theta_0)}{2\|\mathbf{v}_0\|^2 \pi^2 \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}}} - \frac{\sqrt{2}\sqrt{-\frac{c_{\mathbf{v}}^2 \theta_0 \cot(\theta_0) \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} \sigma + 80 c_{\mathbf{v}} \theta_0 \cot(\theta_0) \|\mathbf{v}_0\| \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} \sigma - 2 c_{\mathbf{v}}^2 (\theta_0 \cot(\theta_0))^2 \sigma^2}{\|\mathbf{v}_0\|^4 (\frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}})^2}}}{\pi^2} \leq c_{\mathbf{Q}} \leq$$

$$\frac{-c_{\mathbf{v}} \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} - 80\|\mathbf{v}_0\| \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} + 4 c_{\mathbf{v}} \sigma \theta_0 \cot(\theta_0)}{2\|\mathbf{v}_0\|^2 \pi^2 \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}}} + \frac{\sqrt{2}\sqrt{-\frac{c_{\mathbf{v}}^2 \theta_0 \cot(\theta_0) \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} \sigma + 80 c_{\mathbf{v}} \theta_0 \cot(\theta_0) \|\mathbf{v}_0\| \pi \frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}} \sigma - 2 c_{\mathbf{v}}^2 (\theta_0 \cot(\theta_0))^2 \sigma^2}{\|\mathbf{v}_0\|^4 (\frac{\gamma_{\mathbf{v}}}{\gamma_{\mathbf{Q}}})^2}}}{\pi^2}.$$

Together with the inequality (5.70) this is enough information to select $c_{\mathbf{v}}, c_{\mathbf{Q}}, \gamma_{\mathbf{v}}$ and $\gamma_{\mathbf{Q}}$.

5.4 Implementation details

As mentioned before, the GSACM model employs a number of the features of Modelica language, such as the ability to express equations in an acausal form and solving of both differential and algebraic equations simultaneously.

An example of this is in the moment to control surface inversion object (cf. Fig. 6.1 below). The required generic control surface deflections $\bar{\delta}$ (cf. Chap. 4) are computed (using a fixed control allocation over the physical control surfaces) by simply solving for $\bar{\delta}$ given the desired moments m_x, m_y, m_z, using the aerodynamical functions that calculate m_x, m_y, m_z given $\bar{\delta}$. Another example is the transformation from the form (5.7),(5.8) to the form (5.9),(5.10). (Recall that the result in Thm. 5.2.5 is given for a system on the form (5.9),(5.10) but should be applied to a system on the form (5.7),(5.8).) If the system we were working with would be of the form (5.9),(5.10) we would, by Thm. 5.2.5 and (5.10), for the error $\boldsymbol{\xi} - \ell(\mathbf{x})$ have the following differential equation

$$\frac{d}{dt}\big(\boldsymbol{\xi} - \ell(\mathbf{x})\big) = -c(\boldsymbol{\xi} - \ell(\mathbf{x})) - \big(\frac{\partial \mathcal{V}}{\partial \mathbf{x}} g(\mathbf{x})\big)^T. \tag{5.78}$$

(This can be viewed as an equation for $\boldsymbol{\xi}$ which, when applied to (5.9),(5.10), is sufficient for stabilization of the feedback connected system (5.5).) However, in the variables that we actually have the differential equation for the error reads

$$\frac{d}{dt}\big(\boldsymbol{\xi} - \ell(\mathbf{x})\big) = h(\boldsymbol{\xi}) + k\mathbf{u} - \frac{\partial \ell(\mathbf{x})}{\partial \mathbf{x}}(f(\mathbf{x}) + g(\mathbf{x})\boldsymbol{\xi}). \tag{5.79}$$

(This can be viewed as an equation for $\mathbf{u}$ in (5.8).) Now, by simply equating the right hand sides of (5.78) and (5.79) we formally perform the required transformation, and this is the way it is implemented in the controller for the GSACM model.

6. Simulations

In this chapter we shall give some examples of the behavior of the controller developed in the previous chapter when used in conjunction with the GSACM model. An overview of the model with the controller added is given in Fig. 6.1. In this setup there are no actuator dynamics present and the body dynamics and controller are exactly as described in the text. The aerodynamics (both in the model used by the controller for moment to control surface deflection computation and the aircraft aerodynamics) are the full aerodynamics, as given by the aerodata tables in the Admire model. The simulations have been carried out using the Dymola integrated development environment on a personal computer running GNU/Linux.

6.1 Pull-up maneuver

A flight trajectory lasting for 120 seconds was programmed consisting of a pull-up maneuver followed by a short dive, executed on a straight track, see Fig. 6.2. It starts at wings level flight at an altitude of 3000 meters. About 5 seconds after the start a pull-up maneuver is executed resulting in a maximal flight path angle of about 35°, after which the aircraft levels out at an altitude of about 7000 meters. Finally, a short dive is executed followed by a leveling out at a little over 6000 meters. The various constants occurring in the controller description were chosen to give a stable system with moderate performance.

The programmed maneuvers consist of a series of small (piecewise linear) changes in the set point value (i.e. trim value) for attitude. However, the value $\boldsymbol{\omega}_0$ for the trimmed angular velocity is held at zero at all times in this example. This means that only during those brief periods of time when the angular velocity of the nominal flight path is actually zero can the values for orientation and velocity be expected to converge to their trimmed counterparts. Despite this, the controller shows good tracking abilities also during periods where the attitude set point changes.

As can bee seen from Fig. 6.3, the deviations in Mach number and velocity from trimmed values are moderate over the entire flight path until the dive is executed. During the dive the Mach number and velocity deviation in body x-axis both rise, which is natural since no air brakes are used. The values move back towards trimmed values at the end of the flight when the dive is interrupted.

In Figure 6.4 the angle of attack and Euler angles are displayed. The angle of attack in Fig. 6.4(a) shows clear deviations from the trimmed values, and this is caused by the repeated small changes in trimmed flight path angle. The largest deviation occurs at the start of the pull-up maneuver. The Euler angles in Fig. 6.4(b) show good tracking of the trimmed values and they converge during the periods when the trimmed values are constant.

6.2 Mixed maneuver

In this case a flight trajectory lasting for 200 seconds was programmed which included maneuvering as well as straight path flight. The other parameters were the same as for the pull-up maneuver. The chosen flight trajectory is shown in Figure 6.5 and starts at wings level flight at an altitude of 3000 meters. The following right and

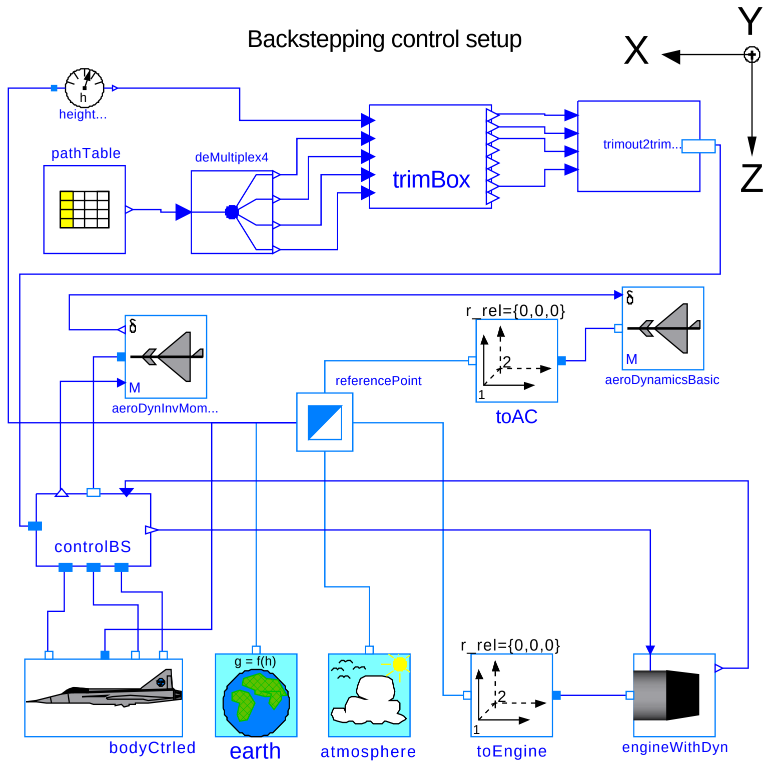


Figure 6.1: Backstepping attitude and velocity controller setup for the GSACM model. The body object in Fig. 2.1 has here been replaced with the bodyCtrled object which has added connectors to the controlBS object, which implements the backstepping controller. The aeroDynInvMoment object (shown just above the controlBS object) performs the inversion form control moments to control surface deflections, which are then in turn forwarded to the aeroDynamicsBasic object. The latter object calculates the aerodynamic forces and moments based on velocity, angular velocity air density and control surface deflections. At the top of the figure are shown the objects pathTable and trimBox which are responsible for creating the nominal flight path and calculating the corresponding trim values for the state variables, respectively.

left turns require bank angles of about 30°. As the left turn is completed, the aircraft begins an 11°-12° climb. During this climb, the Mach number is gradually increased from 0.6 to 0.7 shown in Figure 6.6(a). The trajectory ends at an altitude of about 8000 meters. In this case, there are long periods where the set points (trimmed values) are constant where, if everything else is held constant, the values of attitude and velocity should converge since the trimmed value ω_0 for the angular velocity is set to its correct (in general nonzero) value.

The velocity tracking yields quite small velocity deviations from the trimmed values, as seen in Figure 6.6(b). There seem to be a steady state velocity error in the last part of the trajectory. The reason is that we require a constant Mach number during the climb, but the controller tracks the velocity. Since the velocity of sound decreases with altitude, the velocity of the aircraft will be decreased as well. Hence the velocity deviation.

Figure 6.7(a) shows the actual angle of attack as well as the trimmed angle of attack at each time instant. The discrepancies in these values seen between 75 seconds and 120 seconds can be explained by the pull-up maneuver for the climb and

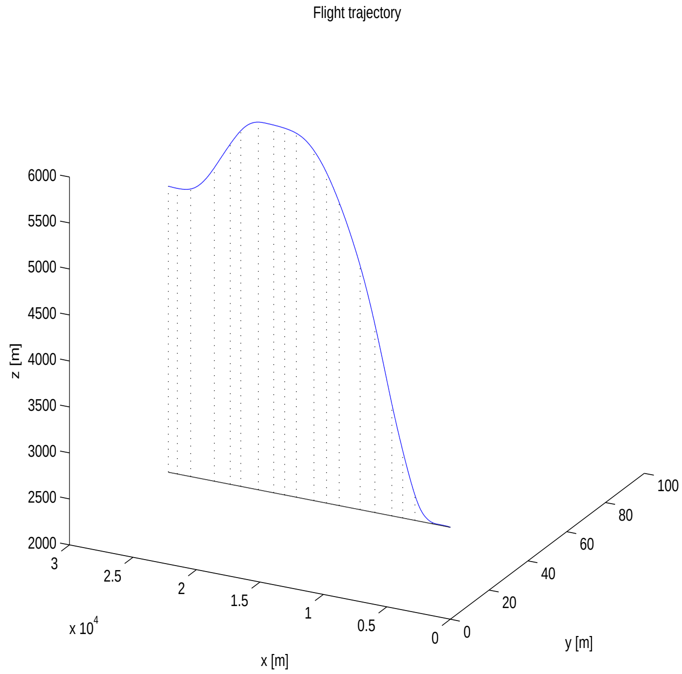
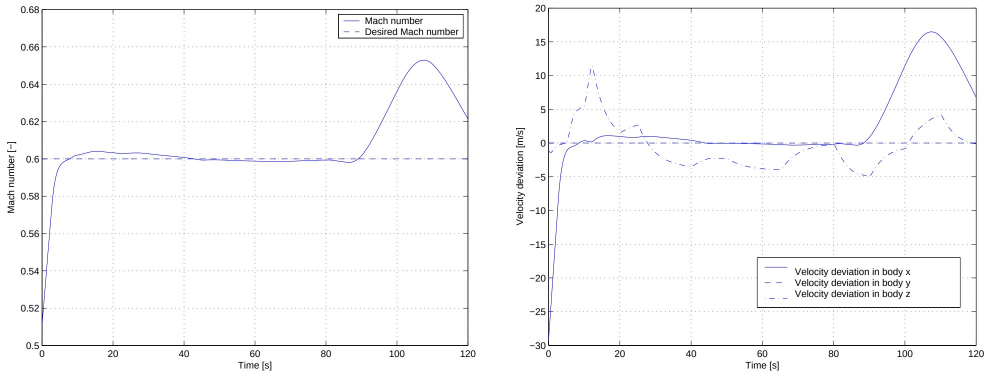


Figure 6.2: Flight trajectory, pull-up maneuver.

by the increase in velocity. The increasing angle of attack after 120 seconds is due to the climb since the air gets thinner. The small mismatch during the climb originates from the velocity deviation discussed above. As desired, the sideslip angle is small during the whole simulation, see Figure 6.7(b).

The orientation of the aircraft stays very close to the trimmed orientation, shown in Figure 6.8. The somewhat larger discrepancy in these values occurring between 75 and 100 seconds comes from the maneuver when the aircraft finishes the turn, by banking to wings level flight, and simultaneously pulls up for the climb. In the last half of the simulation, the slowly increasing angle of attack can be noted in the aircraft orientation.

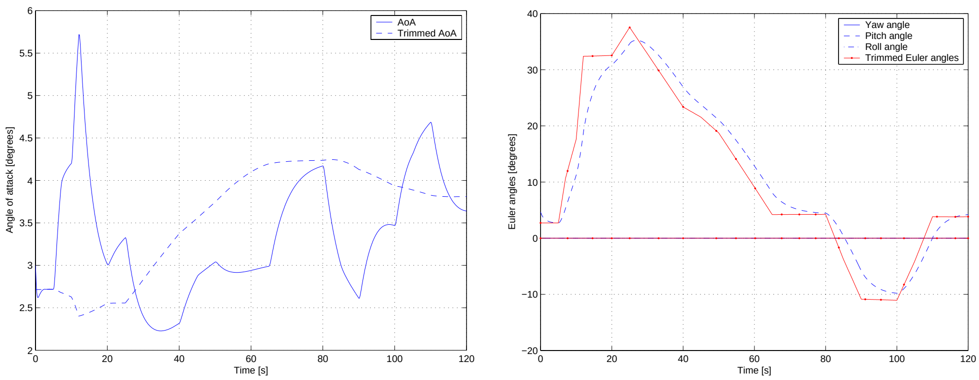
Finally, in Figure 6.9, the value of the time derivative of the "inner" Lyapunov function used (corresponding to $\mathcal{V}$ in Thm. 5.2.5) along the solutions to the feedback connected system (corresponding to (5.5)) is displayed. It can be seen from Fig. 6.9(a) that the overall behavior of the time derivative is as expected with the most negative values occurring shortly after the start when the system moves towards an equilibrium corresponding to straight path flight. The large negative values here can be explained by the fact that the aircraft starts at Mach 0.5 but the trimmed value is 0.6. The time derivative of the Lyapunov function becomes positive on a few occasions when the set point values change between trimmed values. These periods of time correspond to the most dramatic changes in trimmed angular velocity ω_0 (as can be inferred from e.g. the trimmed roll angle, cf. Fig. 6.8(b)) and during these periods the Lyapunov function rate of change need not be negative.



(a) Mach number

(b) Velocity deviation

Figure 6.3: Mach number and velocity deviation, pull-up maneuver.



(a) Angle of attack

(b) Euler angles

Figure 6.4: Angle of attack and Euler angles, pull-up maneuver.

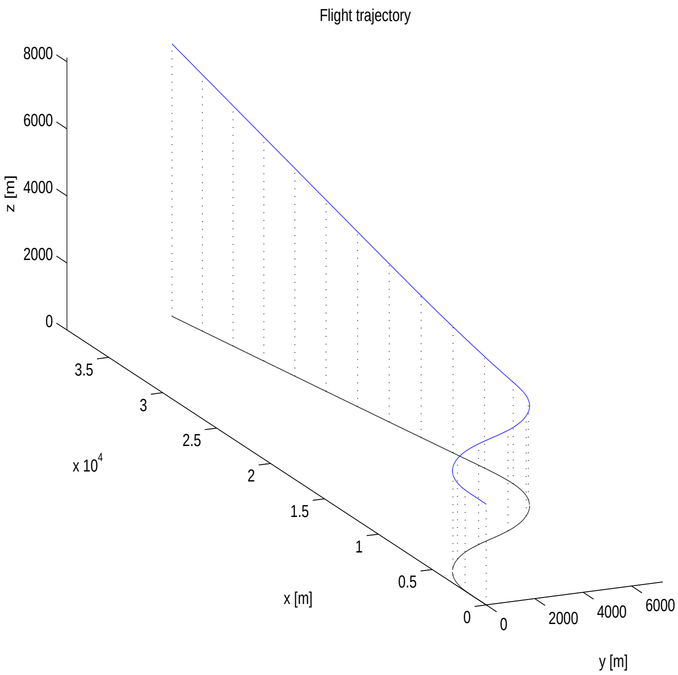
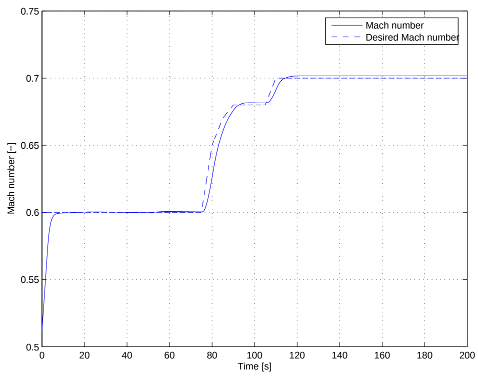
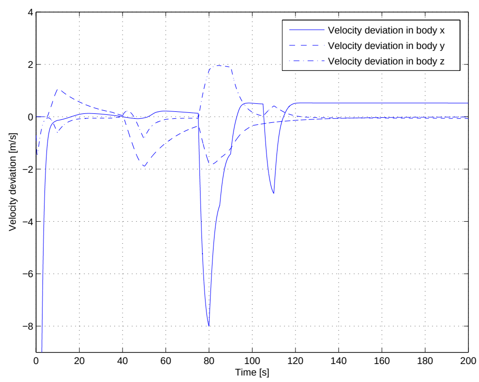


Figure 6.5: Flight trajectory, mixed maneuver.

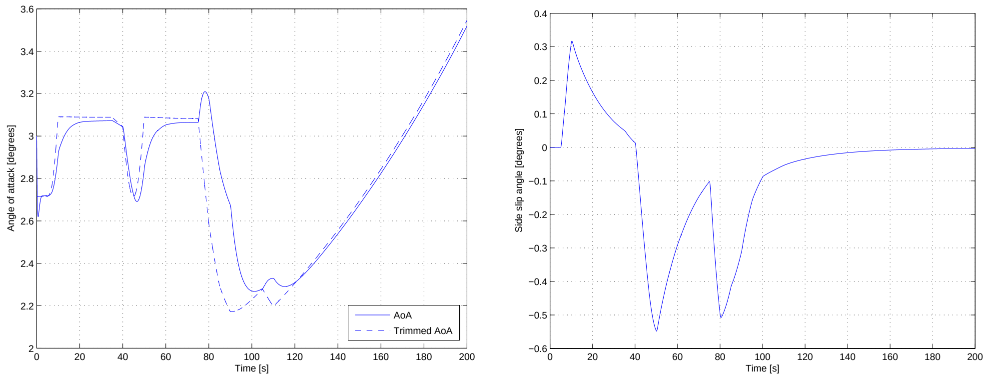


(a) Mach number



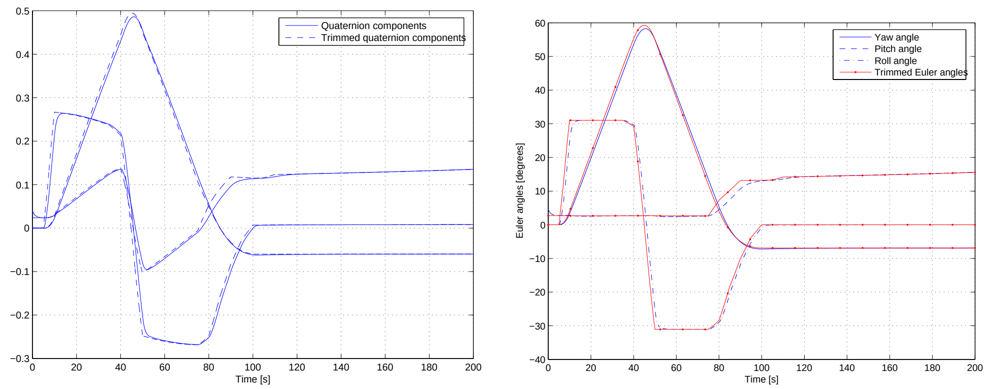
(b) Velocity deviation

Figure 6.6: Mach number and velocity deviation, mixed maneuver.



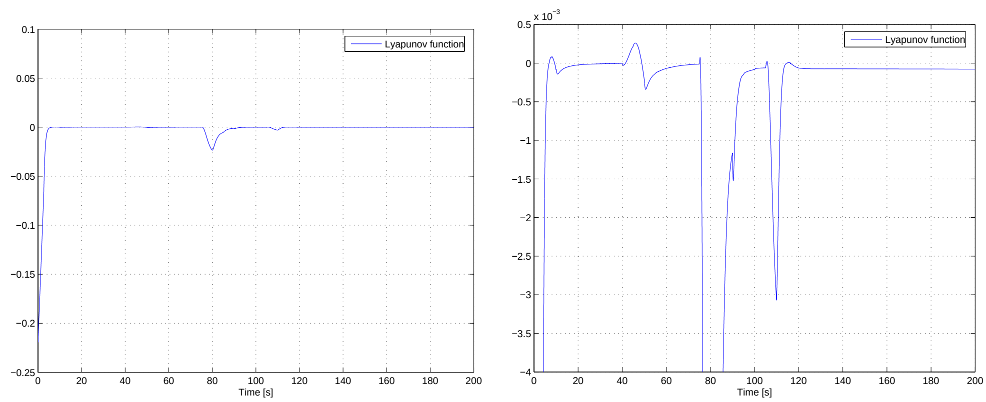
(a) Angle of attack (b) Sideslip angle

Figure 6.7: Angle of attack and sideslip angle, mixed maneuver.



(a) Quaternion components (b) Euler angles

Figure 6.8: Components of the quaternion and corresponding Euler angles, mixed maneuver.



(a) Lyapunov function rate of change (b) Detail of (a)

Figure 6.9: Lyapunov function rate of change, mixed maneuver.

A. Sperical Linear Interpolation, slerp

The slerp [7] was introduced as a canonical way to smoothly interpolate between two orientations described by quaternions. There have been several extensions of the idea, such as interpolations between several orientations, but we shall only make use of the slerp here.

In order to introduce the slerp we need to recall some properties of quaternions. The first and most basic property is that if we, as usual, associate with each unit norm vector $\mathbf{u} \in \mathbb{R}^3$ a pure quaternion (one with real part zero) $\tilde{\mathbf{u}} = (0, \mathbf{u})$ we have

$$\tilde{\mathbf{u}} \circ \tilde{\mathbf{u}} = -1 = (-1, 0) \tag{A.1}$$

(in analogy with the case for complex numbers). This means that for $\phi \in \mathbb{R}$ we can write down the following formal series expansion

$$\exp(\phi\tilde{\mathbf{u}}) = 1 + \phi\tilde{\mathbf{u}} + \frac{(\phi\tilde{\mathbf{u}})^2}{2!} + \frac{(\phi\tilde{\mathbf{u}})^3}{3!} + \ldots$$

(where the powers of course are for quaternion multiplication) and if we employ (A.1) we see that the series in fact converges in the usual topology of $\mathbb{R}^4$. Next we note that any quaternion with norm less than or equal to π can be uniquely expressed as $\phi\tilde{\mathbf{u}}$ with $\phi \in [0, \pi]$ and $\tilde{\mathbf{u}}$ a unit norm quaternion. Hence, if we recall the power series for $\sin$ and $\cos$ we see that we can in fact rigorously define the exponential function injectively for pure quaternions with norm less than or equal to π as

$$\exp(\phi\tilde{\mathbf{u}}) = \cos(\phi) + \sin(\phi)\tilde{\mathbf{u}}$$

(which is the quaternion analogue of Euler's formula). Moreover, by the above any unit norm quaternion $\mathcal{Q}$ is in the range of the exponential function, i.e.

$$\mathcal{Q} = \cos(\phi) + \sin(\phi)\tilde{\mathbf{u}} \tag{A.2}$$

for some unique $\phi, \tilde{\mathbf{u}}$ with $\phi \in [0, \pi]$ and $\tilde{\mathbf{u}}$ unit norm and pure, and therefore we can likewise define the logarithm function as

$$\log(\mathcal{Q}) = \phi\tilde{\mathbf{u}}, \tag{A.3}$$

where the right hand side is a pure quaternion. Having defined the exponential and logarithm functions it is easy to proceed and define the power of of a unit norm quaternion $\mathcal{Q}$ as

$$\mathcal{Q}^t = \exp(\phi\tilde{\mathbf{u}})^t = \cos(t\phi) + \sin(t\phi)\tilde{\mathbf{u}}, \quad t \in \mathbb{R} \tag{A.4}$$

(which is the quaternion analogue of de Moivre's identity). This gives us the following important differential equation

$$\frac{d}{dt}\mathcal{Q}^t = -\phi\sin(t\phi) + \phi\cos(t\phi)\tilde{\mathbf{u}} = (\cos(t\phi) + \sin(t\phi)\tilde{\mathbf{u}}) \circ \phi\tilde{\mathbf{u}} = \mathcal{Q}^t \circ \log(\mathcal{Q}).$$

It implies in particular that if $\mathcal{Q}_0$ is another quaternion we have

$$\frac{d}{dt}(\mathcal{Q}_0 \circ \mathcal{Q}^t) = (\mathcal{Q}_0 \circ \mathcal{Q}^t) \circ \log(\mathcal{Q}). \tag{A.5}$$

Next we shall show that the slerp in (5.43) satisfies an equation of the form (A.5).

To begin with we note that since $\mathcal{Q}_0$,$\mathcal{Q}_1$ both are unit norm quaternions we have

$$\mathcal{Q}_0^{-1} \circ \mathcal{Q}_1 = \mathcal{Q}_0^c \circ \mathcal{Q}_1 = (\cos(\theta), \sin(\theta)\mathbf{u}) = \cos(\theta) + \sin(\theta)\tilde{\mathbf{u}} \tag{A.6}$$

for a certain unit norm $\mathbf{u} \in \mathbb{R}^3$, where θ is the angle defined in (5.44) and $\tilde{\mathbf{u}} = (0, \mathbf{u})$. Therefore, the formula (5.43) for the slerp can be written

$$\begin{aligned}
\mathcal{Q}(t) &= \mathcal{Q}_0 \circ \Big(\frac{\sin((1-t)\theta)}{\sin(\theta)} + \frac{\sin(t\theta)}{\sin(\theta)}\mathcal{Q}_0^{-1} \circ \mathcal{Q}_1\Big) \\
&= \mathcal{Q}_0 \circ \Big(\frac{\sin((1-t)\theta)}{\sin(\theta)} + \frac{\sin(t\theta)\cos(\theta)}{\sin(\theta)} + \sin(t\theta)\tilde{\mathbf{u}}\Big) \\
&= \mathcal{Q}_0 \circ (\cos(t\theta) + \sin(t\theta)\tilde{\mathbf{u}}) \\
&= \mathcal{Q}_0 \circ (\mathcal{Q}_0^{-1} \circ \mathcal{Q}_1)^t, \quad t \in [0,1],
\end{aligned} \tag{A.7}$$

where we have used (A.2),(A.4) as well as some standard trigonometric identities. Thus, if we take $\mathcal{Q} = \mathcal{Q}_0^{-1} \circ \mathcal{Q}_1$ in (A.5) we see that the slerp satisfies the differential equation

$$\frac{d}{dt}\mathcal{Q}(t) = \mathcal{Q}(t) \circ \log(\mathcal{Q}_0^{-1} \circ \mathcal{Q}_1) = \mathcal{Q}(t) \circ \theta\tilde{\mathbf{u}} = \frac{1}{2}\mathcal{Q}(t) \circ (0, 2\theta\mathbf{u}), \tag{A.8}$$

where $\tilde{\mathbf{u}}$ is the one occurring in (A.6) and θ is the angle defined in (5.44). Since $\tilde{\mathbf{u}}$ is a pure quaternion, this shows that the slerp satisfies a differential equation of the same form as (3.3) but where the angular velocity is given by the constant vector $2\theta\mathbf{u}$. Moreover, as an initial value problem it is clear that for bounded $\boldsymbol{\omega}$ the differential equation (3.3) has (by the global Lipschitz continuity) a unique solution. It follows that the solution to (3.3) is a slerp *if and only if the angular velocity vector driving the equation is constant in time.* The angular velocity vector in question for the slerp in (5.43),(A.7) is $2\theta\tilde{\mathbf{u}}$ with $\tilde{\mathbf{u}} = (0, \mathbf{u})$ as in (A.6), where $\mathbf{u} \in \mathbb{R}^3$ represents the constant axis of turning for the rotation described by the slerp. The solution to the differential equation (3.3) in this case at time t is thus the quaternion $\mathcal{Q}(t)$ as in (5.43),(A.7) which represents a rotation $2t\theta$ radians along the axis given by the vector $\mathbf{u} \in \mathbb{R}^3$.

The only item remaining in order to fully understand the connection between the slerp in (5.43),(A.7) and the solutions to the differential equation (3.3) is to derive an explicit expression for the vector $\mathbf{u}$ occurring in (A.6). This is quickly done however, since by (A.6) we have

$$\frac{1}{\sin(\theta)}\big(\mathcal{Q}_0^c \circ \mathcal{Q}_1 - \cos(\theta)\big) = \tilde{\mathbf{u}} = (0, \mathbf{u}).$$

This shows that the left hand side in fact is pure and we can therefore write

$$\mathbf{u} = \frac{1}{\sin(\theta)}\Im\big(\mathcal{Q}_0^c \circ \mathcal{Q}_1 - \cos(\theta)\big) = \frac{1}{\sin(\theta)}\Im(\mathcal{Q}_0^c \circ \mathcal{Q}_1).$$

It is now easy to verify that the vector $2\theta\mathbf{u}$ becomes identical with $\boldsymbol{\omega}_{\mathbf{Q}}^{des}(\mathbf{Q}, \mathbf{Q}_0)$ in (5.49) for the case $c_{\mathbf{Q}} = 1$ if we replace $\mathcal{Q}_0$,$\mathcal{Q}_1$ by $\mathbf{Q} + \mathbf{Q}_0$,$\mathbf{Q}_0$, respectively.

B. Translation between document and code notation

In Tables B.1,B.2 we have listed, side by side, the quantities describing the the kinematics and dynamics of the aircraft in their notation as we have presented it here and in the notation used in the Modelica code of the GSACM model.

Document vs. code notation, BF defined quantities		
This document	**GSACM code**	**Annotation**
$\mathbf{v}$	`v`	Velocity vector
$\boldsymbol{\omega}$	`w`	Angular velocity vector
ω_1	`p_w`	Roll rate
ω_2	`q_w`	Pitch rate
ω_3	`r_w`	Yaw rate
$\mathbf{f}^{(a)} + \mathbf{t}$	`F`	Total force† vector
$\mathbf{m}$	`M`	Total moment vector about CoG
$\mathbf{j}$	`I_CoG`	Moment of inertia about CoG

Table B.1: Translation table between notation used in this document and the notation used in the GSACM model Modelica code for body frame f_b (BF) defined quantities. † Note that there is *not* an exact correspondence between the total force in the document and code, respectively, since one part of the total force is expressed separately (explicitly) in the code, namely the "gravity vector" (denoted g in the code).

Document vs. code notation, misc. quantities		
This document	**GSACM code**	**Annotation**
$\mathbf{O}$	`r`	Body CG location in IF
$\mathbf{Q}$	`Q`	Body orientation quaternion in IF
m	`m_tot`	Total mass of the body
$\mathbf{R}$	`Tmat`	Rot. matrix BF→IF (corrsp. to $\mathbf{Q}$)

Table B.2: Translation table between notation used in this document and the notation used in the GSACM model Modelica code for inertial frame F_e (IF) defined quantities and quantities that are defined without reference to a frame.

Bibliography

[1] P. Fritzson, *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*, IEEE Press, Piscataway NJ, 2004.

[2] Michael A. Tiller, *Introduction to Physical Modeling with Modelica*, Kluwer, Boston MA, 2001.

[3] B.L. Stevens and F.L. Lewis, *Aircraft Control and Simulation*, Wiley, Hoboken NJ, 2003.

[4] H.K. Khalil, *Nonlinear Systems*, 3:rd ed., Prentice Hall, NJ, 2002.

[5] M. Krstić, I. Kanellakopoulos and P. Kokotović, *Nonlinear and Adaptive Control Design*, Wiley, New York, 1995.

[6] T. Glad, and O. Härkegård, "Backstepping Control of a Rigid Body," *Proc. CDC '02*, Las Vegas NV, 2002.

[7] K. Shoemake, "Animating Rotation with Quaternion Curves," Proc. SIGGRAPH '85, ACM, San Francisco, CA, 1985, pp. 245–254.